FEDOPD: PRIVATE ON-POLICY DISTILLATION OF HETEROGENEOUS CLIENT KNOWLEDGE INTO FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Transferring decentralized domain expertise to a centralized foundation model is critical for federated foundation models (FedFMs). However, existing methods either expose model states via parameter sharing or risk training data leakage through client-side content generation and outcome-reliant feedback. We propose **FedOPD**, an on-policy distillation framework that renders clients entirely *generation-free* and *outcome-free*. In FedOPD, the server dispatches its own reasoning rollouts to domain clients, whose frozen experts passively evaluate the received tokens via teacher forcing to return dense, token-level log-probabilities. This discrepancy directly optimizes the sequence-level reverse KL divergence with an $\mathcal{O}(1)$ variance bound, bypassing coarse verifiers and memorization leakage. To facilitate practical deployment, we develop **SPBO** to prevent catastrophic forgetting across sequential domains and **TTF** to prune feedback communication by up to 80%. Extensive experiments on mathematical and sequential multi-domain benchmarks show that FedOPD outperforms prior federated baselines and centralized verifier-guided GRPO across 3B and 7B students, while using only 1% of FedDF's communication bandwidth.

1 INTRODUCTION

Federated foundation models (FedFMs) aim to enrich a centralized base model with proprietary domain expertise distributed across private clients (Fan et al., 2025; Kang et al., 2023; Ren et al., 2025). Increasingly, such knowledge resides in compact expert models fine-tuned locally by domain entities (e.g., healthcare and financial institutions). Transferring it would strengthen the server model's specialized capabilities, but privacy mandates and proprietary constraints preclude clients from sharing their training data, ground-truth responses, or model parameters.

Existing client-to-server transfer paradigms fall into two lines, both with privacy vulnerabilities: (1) **Parameter-aggregation methods** iteratively synchronize and average locally updated weights or adapters (McMahan et al., 2017; Zhang et al., 2023; Zhu et al., 2026b); the transmitted parameters are costly to communicate and can be inverted into training samples (Zhu et al., 2019). (2) **Client-generative and outcome-reliant methods** instead rely on clients' local outputs—either transferring synthetic or expert-generated content for centralized distillation (Li et al., 2024; Abacha et al., 2024), or returning scalar rewards derived from private outcome verifiers (Zhu et al., 2026a). Generated content risks regurgitating sensitive training corpora (Carlini et al., 2021), while scalar rewards conditioned on private ground-truth outcomes remain susceptible to inference-based leakage and yield sparse signals across long reasoning trajectories. Neither paradigm simultaneously achieves leakage resistance, communication efficiency, and dense supervision.

We propose **FedOPD**, an on-policy distillation framework that turns client experts from generative sources or outcome arbiters into passive, token-level critics, so that each client is *generation-free* and *outcome-free*. The server samples reasoning trajectories from its current student and dispatches them to domain-matching clients; a frozen client expert evaluates these server-authored tokens via teacher forcing and returns only per-token log-probabilities. The server then uses the teacher–student log-probability gap as a dense token-level advantage in a clipped policy-gradient objective (Schulman et al., 2017), which optimizes the sequence-level reverse KL divergence. The upload is thus

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.

restricted to scalars on server-generated sequences, closing the exposure channels of parameter sharing and generative memorization, while providing fine-grained rather than scalar supervision.

Extending FedOPD to sequential multi-domain settings raises two practical challenges: *catastrophic forgetting* across consecutively visited domains, and the cumulative uplink cost of token-level feedback on long trajectories. Without modifying the OPD objective, **SPBO** (Stability–Plasticity Balanced Optimization) restarts the learning-rate schedule per domain block to restore plasticity and penalizes parameter drift along coordinates critical to earlier domains, and **TTF** (Top-Token Feedback) requests teacher feedback only on tokens where the student is most uncertain.

Theoretically, we prove that FedOPD minimizes the sequence-level reverse KL with an $O(1)$ variance bound, whereas sequence-level credit accumulates $\Omega(T)$ variance; derive stability–plasticity bounds for SPBO that restrict backward forgetting; and show that TTF minimizes the worst-case gradient approximation error while preserving a descent direction. Empirically, with 3B and 7B students guided by frozen 1.5B client experts on mathematical and sequential Math→Code→Science benchmarks, FedOPD surpasses the strongest federated baseline by up to 1.38 points on math and 3.42 points across domains, and even outperforms centralized, verifier-guided GRPO, while using only 1% of FedDF's communication. SPBO improves sequential cross-domain accuracy by 3.98 points, and TTF cuts 80% of the feedback uplink at a cost of only 0.74 points.

Our main contributions are summarized as follows:

- **A Generation-Free Transfer Paradigm**: We identify the fundamental vulnerabilities of parameter-sharing and client-generative paradigms, and propose FedOPD, an outcome-free, generation-free on-policy distillation framework across asymmetric model scales.
- **Continual & Communication-Efficient Designs**: We introduce SPBO to resolve the stability–plasticity dilemma across sequential domains and TTF to prune up to 80% of token feedback communication, supported by theoretical stability and error bounds.
- **Theoretical & Empirical Superiority**: We provide complete theoretical justifications for variance reduction, forgetting bounds, and descent preservation. Extensive experiments show that FedOPD consistently outperforms model-, data-, and feedback-level baselines while utilizing minimal communication.

2 Related Work

Knowledge transfer in federated foundation models. FedFMs transfer proprietary domain expertise from clients to a server-hosted model (Fan et al., 2025; Kang et al., 2023; Ren et al., 2025). *Parameter-aggregation methods* synchronize client updates of a shared model (McMahan et al., 2017), usually through parameter-efficient modules at foundation-model scale (Zhang et al., 2023; Sun et al., 2024; Yi et al., 2023; Fan et al., 2023; Ye et al., 2024); FedPDPO extends this route to preference alignment (Zhu et al., 2026b). They require every client to host the server's architecture, incur heavy uplink traffic, and expose updates to gradient inversion (Zhu et al., 2019; Chen et al., 2024). *Client-generative and outcome-reliant methods* keep weights local and transfer what clients produce. Clients may upload synthetic domain data, optionally under differential privacy (Li et al., 2024; Abacha et al., 2024; Hou et al., 2025), which can reproduce memorized training examples (Carlini et al., 2021). Heterogeneous distillation exchanges client predictions on a public or synthesized transfer set (Li & Wang, 2019; Lin et al., 2020; Zhu et al., 2021); it admits different architectures, but the transfer set is decoupled from the server model's own continuations and each position carries a full next-token distribution. FedGRPO returns one scalar reward per server-sampled solution, computed against private client outcomes (Zhu et al., 2026a), so the signal depends on private references and assigns the same credit to every token. FedOPD instead makes the client a passive critic that only scores server-sampled trajectories token by token.

On-policy distillation. Knowledge distillation (Hinton et al., 2015) on teacher- or human-written sequences trains the student on prefixes it will not visit at inference, an error that compounds along the horizon (Ross et al., 2011; Bengio et al., 2015). On-policy distillation instead has the teacher score student-generated sequences, through token-level divergences (Agarwal et al., 2024), reverse-KL policy gradients (Gu et al., 2024), or the per-token teacher–student log-probability gap used directly as the advantage (Thinking Machines Lab, 2025). On-policy training also forgets less: fine-tuning on off-policy targets erodes prior capabilities (Yang et al., 2024; Ren et al., 2024), and

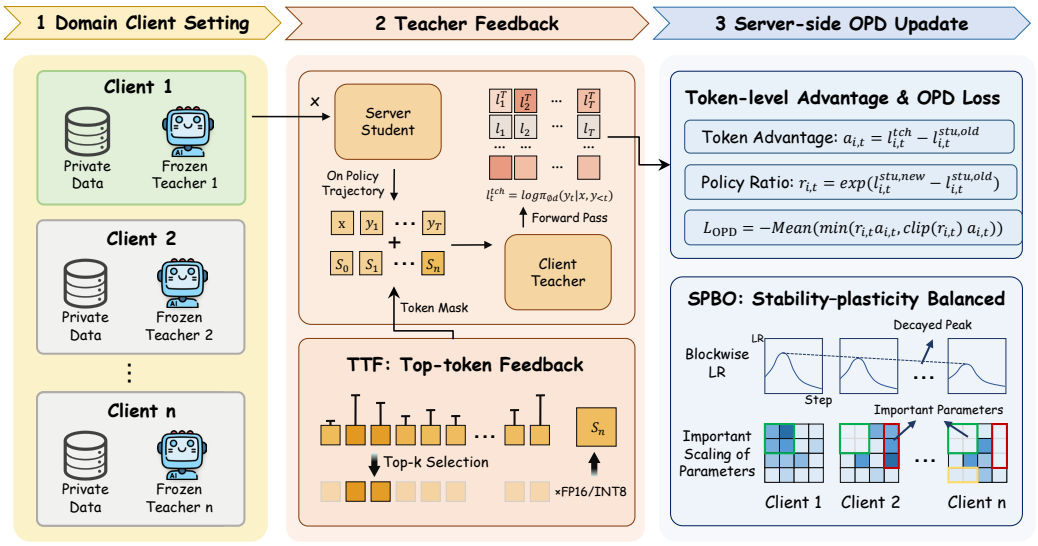


Figure 1: Overview of FedOPD. The server sends its own on-policy trajectory to the domain-matched client, whose frozen expert only scores it by teacher forcing and returns per-token log-probabilities; no parameters, generated tokens, or reference answers leave the client. The server uses the teacher–student gap as a token-level advantage in a clipped policy update. SPBO reshapes the optimizer across sequential domains, and TTF selects which tokens are scored.

reinforcement learning retains more than supervised fine-tuning because on-policy sampling keeps the policy close to the base model (Chu et al., 2025; Shenfeld et al., 2025; Chen et al., 2026). These methods assume a centrally available teacher. FedOPD brings on-policy scoring to compact domain experts held privately by clients, and only token-level log-probabilities leave each client.

3 Method

3.1 Problem Formulation

Setting. A server collaborates with N clients, one per domain; domains may be coarse, such as Math, Code, and Science, or topics within one subject. Client k owns private data $\mathcal{D}_k = \{(x_i^{(k)}, y_i^{(k)})\}$ and a frozen expert π_{ϕ_k} trained on it, which is smaller than the server model but specialized in domain k. The server trains a foundation model π_θ, where θ may be all weights or an adapter, and holds only an unlabeled prompt set $\mathcal{X}_k$ per domain, without reference responses. Every response $y = (y_1, \ldots, y_T)$ used in training is sampled by the server from its own current model; none is taken from a dataset or written by a client. Each prompt is routed to the client of its domain; experts of different scales are never averaged.

Threat model and client feedback. Server and clients are semi-honest: they follow the protocol but may try to infer private information from what they receive. Each client keeps its data $\mathcal{D}_k$ and expert ϕ_k local, since uploaded weights or gradients can be inverted into training samples (Zhu et al., 2019) and content generated by the expert can reproduce them (Carlini et al., 2021). In FedOPD, a client uploads only the following feedback.

Definition 1 (Passive-critic feedback)**.** *Given a prompt x, a response y sampled by the server, and a set S of token positions chosen by the server, client k returns $m_k(x, y; S) = \{\ell_t^{\text{tch}}\}_{t \in S}$, where $\ell_t^{\text{tch}} = \log \pi_{\phi_k}(y_t \mid x, y_{<t})$.*

The client computes m_k with a single teacher-forcing pass of its frozen expert over (x, y). The feedback thus contains no weights, gradients, or activations of ϕ_k (*parameter-free*), no tokens sampled from π_{ϕ_k} (*generation-free*), and no quantity computed from the private answers $y^{(k)}$ or a

verifier (*outcome-free*). Definition 1 specifies what a client uploads, not how much the returned log-probabilities reveal about ϕ_k or $\mathcal{D}_k$.

We operate under the semi-honest threat model where the server legitimately samples and evaluates (x, y) according to the protocol. Although arbitrary black-box log-probability queries could enable membership inference or model extraction under an active, malicious server, providing formal differential privacy against such malicious adversaries is orthogonal to this work and left for future study.

Objective. With $\ell_t^{\mathrm{stu}} = \log \pi_\theta(y_t \mid x, y_{<t})$, the server maximizes the teacher–student agreement on its own rollouts,

$$J(\theta) = \mathbb{E}_{k,\, x\sim\mathcal{X}_k,\, y\sim\pi_\theta(\cdot|x)}\Big[\sum_{t=1}^{T}\big(\ell_t^{\mathrm{tch}} - \ell_t^{\mathrm{stu}}\big)\Big] = -\,\mathbb{E}_{k,\, x\sim\mathcal{X}_k}\,\mathrm{KL}\big(\pi_\theta(\cdot \mid x)\,\big\|\,\pi_{\phi_k}(\cdot \mid x)\big), \quad (1)$$

a sequence-level reverse KL that decomposes over tokens and whose only client-side quantities form the feedback of Definition 1 with S covering all valid tokens.

Deployment constraints. (C1) *Sequential availability*: clients are rarely online together, so the server visits domains in blocks $k = 1, \ldots, N$ without revisiting earlier prompts or feedback, which turns Eq. (1) into a continual-learning problem. (C2) *Uplink budget*: full feedback carries one value per valid token and grows with response length. SPBO and TTF address them without changing Eq. (1).

3.2 FedOPD: Clients as Passive Token-Level Critics

Interaction. The server fixes the behavior policy $\pi_{\theta_{\mathrm{old}}}$, samples $y \sim \pi_{\theta_{\mathrm{old}}}(\cdot \mid x)$ for $x \sim \mathcal{X}_k$, records $\ell_t^{\mathrm{stu,old}}$, and sends (x, y) to client k, which returns ℓ_t^{tch} on every valid completion token (mask M_t; teacher and student share a tokenizer).

Token-level advantage. Each token is scored by the gap between the expert and the student that produced it, normalized across trajectories and teachers:

$$a_t^{\mathrm{raw}} = \ell_t^{\mathrm{tch}} - \ell_t^{\mathrm{stu,old}}, \qquad a_t = \mathrm{clip}\Big(\frac{a_t^{\mathrm{raw}} - \bar{a}^{\mathrm{raw}}}{\sigma_{\mathcal{B}}},\, -c_a,\, c_a\Big), \quad (2)$$

where $\bar{a}^{\mathrm{raw}}$ is the masked trajectory mean and $\sigma_{\mathcal{B}}$ the batch standard deviation of the centered gaps. A positive a_t means the expert assigns y_t more probability than the student, so y_t is reinforced; a negative one suppresses it. The signal is dense and independent of answer correctness; centering removes trajectory-level offsets between experts (Proposition 1), and whitening equalizes their scale.

Policy update. Holding y and a_t fixed, the server minimizes the clipped surrogate of PPO (Schulman et al., 2017) for E epochs per batch,

$$\mathcal{L}_{\mathrm{OPD}}(\theta) = -\frac{\sum_t M_t \min\big(r_t a_t,\ \mathrm{clip}(r_t, 1-\epsilon, 1+\epsilon)\, a_t\big)}{\sum_t M_t}, \qquad r_t = \exp\big(\ell_t^{\mathrm{stu,new}} - \ell_t^{\mathrm{stu,old}}\big), \quad (3)$$

with $\ell_t^{\mathrm{stu,new}} = \log \pi_\theta(y_t \mid x, y_{<t})$. Algorithm 1 gives the full procedure; removing its SPBO and TTF steps recovers plain FedOPD.

3.3 Stability–Plasticity Balanced Optimization

Under (C1), a single learning-rate schedule has decayed before later clients arrive, and unconstrained updates drift along coordinates that earlier domains rely on. SPBO fixes each failure with one optimizer-level change and leaves the client feedback, the advantage, and $\mathcal{L}_{\mathrm{OPD}}$ untouched.

Blockwise learning-rate rejuvenation. Each block of K steps starts a new cosine cycle with a geometrically decaying peak,

$$\eta_k(\tau) = \eta_{\max}\Big[\rho + \big(\beta^{k-1} - \rho\big)\frac{1 + \cos(\pi\tau/K)}{2}\Big], \qquad \tau = 0, \ldots, K-1, \quad (4)$$

Algorithm 1 FedOPD over sequential domain blocks, with the optional SPBO and TTF steps

Input: prompt sets $\{\mathcal{X}_k\}_{k=1}^{N}$; frozen client experts $\{\pi_{\phi_k}\}$; student θ; K, E; [SPBO] $\hat{\Omega} \leftarrow \mathbf{0}$
for $k = 1$ **to** N **do**
  [SPBO] Start a new learning-rate cycle $\eta_k(\cdot)$ (Eq. 4); set $w \leftarrow 1/(1 + \lambda\hat{\Omega})$
  for $\tau = 0$ **to** $K - 1$ **do**
    $\theta_{\text{old}} \leftarrow \theta$; sample $x \sim \mathcal{X}_k$, $y \sim \pi_{\theta_{\text{old}}}(\cdot \mid x)$; record $\ell_t^{\text{stu,old}}$
    $S \leftarrow$ all valid tokens; [TTF] $S \leftarrow S_n$ (Eq. 7)
    Send (x, y, S) to client k; receive $\{\ell_t^{\text{tch}}\}_{t \in S}$ ▷ teacher forcing only
    Compute a_t (Eq. 2); E times: $\theta \leftarrow \theta - \eta_k(\tau)\, w \odot d$, d: AdamW direction of $\nabla\mathcal{L}_{\text{OPD}}$
  end for
  [SPBO] Add the SI importance of block k (Eq. 5) to $\hat{\Omega}$ and normalize
end for
Return θ ▷ without SPBO: one schedule and $w \equiv \mathbf{1}$

so block k peaks at $\eta_{\max}\beta^{k-1}$ and every block decays to the floor $\rho\,\eta_{\max}$: later clients always receive a usable step size, while $\beta < 1$ gradually tempers how far they move the model.

Importance-modulated updates. To locate earlier knowledge, SPBO uses synaptic intelligence (SI) (Zenke et al., 2017): each coordinate i accumulates its contribution to the decrease of $\mathcal{L}_{\text{OPD}}$ during block k,

$$\widetilde{\Omega}_i^{(k)} = \max\Big(0,\ \frac{\sum_\tau -g_{i,\tau}\,\Delta\theta_{i,\tau}}{\big(\theta_i^{(k)} - \theta_i^{(k-1)}\big)^2 + \epsilon_{\text{imp}}}\Big), \tag{5}$$

where $g_{i,\tau}$ and $\Delta\theta_{i,\tau}$ are the gradient and the realized step, and $\theta^{(k)}$ denotes the parameters after block k. Importance is summed over blocks, mean-normalized, and clipped to give $\hat{\Omega}$, which rescales every AdamW update from the second block on:

$$\theta_i \leftarrow \theta_i - \eta_k(\tau)\, w_i\, d_i, \qquad w_i = \frac{1}{1 + \lambda\hat{\Omega}_i}, \tag{6}$$

where $d = \hat{\boldsymbol{\mu}}/(\sqrt{\hat{\boldsymbol{\nu}}} + \epsilon_{\text{A}})$ is the AdamW direction of the unmodulated gradient; rescaling the gradient instead would be cancelled by AdamW's per-coordinate normalization. Equation (6) solves the diagonal proximal step $\min_\Delta \langle d, \Delta\rangle + \frac{1}{2\eta_k(\tau)}\sum_i (1 + \lambda\hat{\Omega}_i)\Delta_i^2$: it penalizes movement along coordinates important to earlier domains, while unused ones ($w_i \approx 1$) remain plastic. Only θ and $\hat{\Omega}$ are carried across blocks; no earlier prompt or feedback is replayed.

3.4 Top-Token Feedback

Under (C2), TTF shrinks the position set S of Definition 1 before the client is contacted and changes nothing else. Using ℓ_t^{stu}, already stored from sampling, the server ranks the L valid tokens of a completion by the uncertainty $u_t = -\ell_t^{\text{stu}}$ and requests feedback on the top fraction n only,

$$S_n = \text{TopK}\big(\{u_t\}_{t=1}^{L}, \lceil nL \rceil\big), \qquad M_t^{\text{TTF}} = M_t\,\mathbf{1}[t \in S_n], \tag{7}$$

and applies Eq. (3) with M^{TTF} in place of M. Retained values may be quantized to INT8 with a shared scale and zero-point, so the value payload scales by n in FP16 and by $n/2$ in INT8. The rule needs neither an extra round trip nor teacher information: confidently predicted tokens have nearly vanishing score functions, so feedback on them barely moves the update (Theorem 2).

3.5 Theoretical Analysis

We analyze the three components in turn; full assumptions and proofs are in Appendix D. Throughout, $s_t = (x, y_{<t})$, $g_t = \ell_t^{\text{tch}} - \ell_t^{\text{stu}}$, $\psi_t = \nabla_\theta \log \pi_\theta(y_t \mid s_t)$, and $p_t = \pi_\theta(y_t \mid s_t)$.

The FedOPD signal. For domain k, let J_k be Eq. (1) restricted to it and $\delta_\theta(s) = \text{KL}\big(\pi_\theta(\cdot \mid s) \,\|\, \pi_{\phi_k}(\cdot \mid s)\big)$ the per-prefix disagreement.

Proposition 1 (Outcome-free token-level signal). *For rollouts of length T drawn from π_θ: (i) $J_k = -\mathbb{E}\sum_t \delta_\theta(s_t)$, a maximum-entropy RL objective with reward $\log \pi_{\phi_k}$ that is maximized exactly at $\pi_\theta = \pi_{\phi_k}$; the FedOPD term $\mathbb{E}\sum_t \psi_t g_t$ is its gradient with the prefix distribution held fixed and, for a per-prefix softmax, vanishes if and only if $\pi_\theta = \pi_{\phi_k}$; (ii) if every step adds conditional variance at least σ^2 to the remaining sum of gaps, weighting token t by the reward-to-go $\sum_{u\geq t} g_u$ or by the sequence-level score $\sum_u g_u$ has variance at least $(T-t)\,\sigma^2$ given $y_{\leq t}$, whereas $|g_t| \leq B$ for log-ratios bounded by B; (iii) prefix-dependent offsets to ℓ_t^{tch} leave the expected update unchanged, and per-trajectory centering removes trajectory-constant offsets exactly.*

Part (i) restates the reverse-KL view of on-policy distillation (Gu et al., 2024; Agarwal et al., 2024): the expert's evaluation of server-authored tokens is a dense reward whose optimum is the expert itself, so neither client generation nor private outcomes are needed. Part (ii) is why a client returns one scalar per token rather than one per completion, and part (iii) is why heterogeneous experts need no calibration once the gaps are centered as in Eq. (2).

SPBO. Let $F_j = -J_j$ be the loss of domain j, $\theta^{(j)}$ the parameters after block j, H_k the learning-rate budget of block k (the sum of its step sizes), and $H_{>j} = \sum_{k>j} H_k$. Assumption 1 (Appendix D) requires $\|d\|_\infty \leq U$ (Lemma 1), $\nabla^2 F_j \preceq \mathrm{diag}(h_j)$ along the later trajectory, and $\hat{\Omega}_i \geq c\sqrt{h_{j,i}}$ for every earlier domain j.

Theorem 1 (Stability–plasticity of SPBO). *(i) For every earlier domain j,*

$$F_j(\theta^{(N)}) - F_j(\theta^{(j)}) \leq \big\langle \nabla F_j(\theta^{(j)}),\, \theta^{(N)} - \theta^{(j)} \big\rangle + \frac{U^2 H_{>j}^2}{2} \sum_i \min\Big\{ h_{j,i},\, \frac{1}{\lambda^2 c^2} \Big\}. \tag{8}$$

(ii) At a step of block k with $\partial_i F_k\, d_i \geq 0$ for all i, and for any $\omega \geq 0$, $\langle \nabla F_k, w \odot d\rangle \geq \frac{1-\kappa_\omega}{1+\lambda\omega}\langle \nabla F_k, d\rangle$, where κ_ω is the share of $\langle \nabla F_k, d\rangle$ contributed by coordinates with $\hat{\Omega}_i > \omega$. (iii) With E epochs per batch, Eq. (4) gives $H_{>j} \leq E\eta_{\max}\big[\frac{(K+1)\beta^j}{2(1-\beta)} + \rho K(N-j)\big]$, whereas a single cosine schedule over all NK steps leaves the last block only $H_N \leq E\eta_{\max}\big[K\big(\rho + \frac{\pi^2}{12N^2}\big) + 1\big]$.

With $\lambda = 0$, the curvature term in Eq. (8) is the full trace $\sum_i h_{j,i}$; SPBO caps every coordinate at $(\lambda c)^{-2}$, so forgetting decreases with λ, while the interference term vanishes at a stationary point of F_j and is negative when later updates also help domain j. By part (ii), plasticity is lost only through the overlap κ_ω between current progress and the coordinates earlier domains rely on. By part (iii), $\beta < 1$ keeps the budget later clients spend on domain j bounded independently of N up to the floor ρ, whereas a single schedule leaves the last client almost none, so what a client can teach would depend on when it arrives.

TTF. For one completion with L valid tokens, let a collect the raw gaps of Eq. (2) and $\Psi = [\psi_1, \ldots, \psi_L]$ the scores at θ_{old}. At the first optimizer step of a batch ($r_t = 1$), full feedback induces the direction Ψa, and TTF, with S fixed before the client is contacted, induces Ψa_S, where a_S is zero outside S. Assume $|a_t| \leq A$ and that the Jacobian of the student logits is bounded by G_J.

Theorem 2 (Teacher-free feedback selection). *(i) For every S, $\|\Psi a - \Psi a_S\| \leq \sqrt{2}\, G_J A \sum_{t\notin S}(1-p_t)$, and the bound is attained; hence, among all sets of size m fixed before the teacher is queried, S_n in Eq. (7) minimizes the worst-case error. (ii) Rounding the retained teacher log-probabilities with step s_q adds at most $\sqrt{2}\, G_J \frac{s_q}{2} \sum_{t\in S}(1-p_t)$. (iii) If the total bound is at most $\varrho\|\Psi a\|$ with $\varrho < 1$, the TTF direction $\hat{g}$ satisfies $\langle \Psi a, \hat{g}\rangle \geq (1-\varrho)\|\Psi a\|^2$, so it remains a descent direction.*

Because a token predicted with $p_t \approx 1$ has a score of norm $O(1-p_t)$ (Lemma 2), the cost of TTF is the uncertainty mass left outside S rather than the retention ratio, and uncertainty ranking is worst-case optimal among rules that must commit before the client is queried.

4 Experiments

We evaluate (Q1) whether FedOPD transfers client expertise better than parameter-aggregation and client-generative methods (Section 4.2), (Q2) whether SPBO controls forgetting under sequential clients (Section 4.3), (Q3) how much uplink TTF saves (Section 4.4), and (Q4) which signal components matter and how robust FedOPD is to the number and order of clients (Section 4.5).

Fine-Grained Mathematical Reasoning

Method	Upload	GSM-H	HARP	MATH500	GSM8K	Int-AL	Int-GE	Int-NT	H-AL	H-GE	H-NT	AVG
Qwen2.5-3B Base												
Student Base	–	39.04	32.91	53.40	79.15	76.66	43.42	54.07	39.59	22.71	27.12	46.81
Central-GT-SFT	Data	37.00	26.72	54.80	73.09	74.73	40.71	52.59	30.82	17.51	21.80	42.98
Central-GRPO	Data	44.66	35.13	61.40	83.47	**81.55**	50.73	61.85	44.23	**25.08**	**29.28**	51.74
FedPDPO	Params	42.08	34.77	60.40	80.36	80.37	46.56	53.70	43.71	24.68	27.62	49.43
DPSDA-FL	Gen.	45.64	35.86	58.80	83.40	81.30	50.10	60.19	44.85	24.29	28.95	51.34
FedDF	Gen.	45.34	35.65	59.60	**84.53**	**81.55**	48.85	58.70	**45.46**	24.53	29.12	51.33
FedGRPO	Reward	39.35	33.77	57.00	78.24	78.18	48.02	53.70	42.27	23.82	27.95	48.23
FedOPD (Ours)	Log-prob	**47.31**	**36.63**	**62.20**	84.00	**81.55**	**51.98**	**63.70**	45.05	24.76	27.95	**52.51**
Qwen2.5-7B Base												
Student Base	–	41.20	42.30	54.60	80.50	72.70	50.52	47.22	48.04	29.34	38.90	50.53
Central-GT-SFT	Data	38.06	43.49	58.20	78.47	76.16	52.40	51.11	49.07	29.50	46.26	52.27
Central-GRPO	Data	**53.80**	50.50	**71.20**	**89.50**	**86.10**	56.80	68.20	53.60	32.80	48.50	61.10
FedPDPO	Params	33.28	47.96	59.40	76.50	77.51	54.07	58.70	53.24	31.52	47.78	54.00
DPSDA-FL	Gen.	44.35	45.82	66.60	82.87	81.72	57.62	65.56	53.40	31.78	47.59	57.73
FedDF	Gen.	51.25	51.90	67.20	88.32	84.67	56.58	68.15	53.09	**34.07**	47.25	60.25
FedGRPO	Reward	28.58	42.13	55.20	69.29	72.03	49.90	48.15	45.77	29.97	46.26	48.73
FedOPD (Ours)	Log-prob	52.69	**53.02**	69.80	89.08	85.76	**58.46**	**70.56**	**54.43**	33.04	**49.42**	**61.63**

Cross-Domain Knowledge Transfer

Method	Upload	MATH500	HARP	Int-MA	HE+	HE	Int-CO	SciBench	ScienceQA	Int-SC	AVG
Qwen2.5-3B Base											
Student Base	–	53.40	32.91	58.00	56.10	63.41	23.40	11.27	73.74	44.80	46.34
Central-GT-SFT	Data	38.20	18.79	41.20	34.15	46.34	**26.80**	8.07	62.41	33.00	34.33
Central-GRPO	Data	55.80	32.93	59.20	57.93	64.02	23.40	12.45	73.59	43.60	46.99
FedPDPO	Params	58.20	34.52	61.40	57.93	63.41	24.20	13.38	71.31	43.00	47.48
DPSDA-FL	Gen.	58.00	34.54	61.80	56.10	63.41	23.80	11.42	73.24	**50.40**	48.08
FedDF	Gen.	60.80	34.17	62.40	58.76	64.24	25.00	12.14	73.09	47.60	48.69
FedGRPO	Reward	53.60	33.52	59.20	56.71	62.80	23.40	12.28	74.24	43.60	46.59
FedOPD (Ours)	Log-prob	**62.40**	**36.13**	**65.00**	**61.59**	**66.46**	24.00	**14.16**	**78.96**	47.80	**50.72**
Qwen2.5-7B Base											
Student Base	–	55.60	41.30	59.80	70.12	**76.83**	27.60	10.26	85.57	49.60	52.96
Central-GT-SFT	Data	42.40	22.20	49.00	59.15	65.85	29.80	8.67	79.23	45.20	44.61
Central-GRPO	Data	60.20	43.16	61.60	69.51	74.39	**31.20**	12.43	88.31	49.80	54.51
FedPDPO	Params	66.80	44.02	69.20	62.20	68.90	28.20	12.86	**89.48**	52.00	54.85
DPSDA-FL	Gen.	**73.20**	45.25	**73.40**	48.17	51.83	27.00	16.04	74.82	43.40	50.35
FedDF	Gen.	67.00	40.29	68.60	64.63	69.51	29.20	15.75	85.21	48.60	54.31
FedGRPO	Reward	57.40	42.01	60.40	70.73	75.61	29.60	10.26	84.94	48.60	53.28
FedOPD (Ours)	Log-prob	72.60	**45.77**	**73.40**	**71.95**	**76.83**	29.20	**18.06**	84.44	**52.20**	**58.27**

Table 1: Main results. *Upload*: what leaves a client. Data: problems with answers or verifiers; Params: model updates; Gen.: outputs generated by the client experts (responses or next-token distributions); Reward: one scalar per response; Log-prob: per-token log-probabilities on server trajectories (Definition 1). Bold: best per student. Int-: in-domain test set; H-: HARP subject split; HE(+): HumanEval(+).

4.1 Experimental Setup

Domains and benchmarks. In the *fine-grained math* setting, three clients hold Algebra, Geometry, and Number Theory, with 2,000 training problems each. We evaluate on held-out in-domain test sets (Int-AL/GE/NT), on the subject splits of HARP (H-AL/GE/NT), which probe out-of-distribution generalization within each sub-field, and on MATH500, HARP, GSM8K, and GSM-Hard. In the *cross-domain* setting, three clients hold Math, Code, and Science (6,000, 9,000, and 10,000 problems), evaluated on in-domain test sets (Int-MA/CO/SC) and on MATH500, HARP, HumanEval(+), ScienceQA, and SciBench. Data sources, deduplication, and splits are given in Appendix B.1.

Models and protocol. The server students are Qwen2.5-3B and Qwen2.5-7B Base. Each client expert is a 1.5B model (Qwen2.5-Math, -Coder, or -Instruct) trained on the client's private data and

Table 2: Sequential distillation (3B). Math-sub.: average of the fine-grained math setting; Cross: per-domain averages and AVG of the cross-domain setting. Full results in Appendix C.1.

Method	Math-sub. AVG	Cross-domain Math	Code	Sci.	AVG
Global-cosine	49.67	48.33	47.97	43.93	46.74
Blockwise-LR	51.75	51.69	48.93	44.57	48.39
Fisher-FedOPD	49.44	50.65	47.64	42.62	46.97
MAS-FedOPD	51.68	52.55	50.14	43.80	48.83
LwF-FedOPD	51.86	51.93	50.41	44.48	48.94
SI-FedOPD	51.97	54.24	**50.68**	44.55	49.83
Mixed FedOPD	52.03	53.11	48.51	41.62	47.75
FedOPD + SPBO	**52.51**	**54.51**	**50.68**	**46.97**	**50.72**

Table 3: Communication over a full cross-domain run (3B, MB). Rel.: total payload relative to FedOPD. Down-link and further TTF variants in Appendix A.3.

Method	Up	Total	Rel.	AVG
FedPDPO	718.61	1437.22	658.5×	47.48
FedDF	217.17	219.43	100.5×	48.69
FedGRPO	0.002	2.51	1.15×	46.59
FedOPD	0.73	2.18	1.00×	**50.72**
+ TTF-80	0.58	2.04	0.93×	50.32
+ TTF-40	0.29	1.75	0.80×	50.02
+ TTF-20	0.15	1.61	0.74×	49.98
+ TTF-40 INT8	0.15	1.61	0.74×	49.92
+ TTF-20 INT8	0.07	1.53	0.70×	48.93

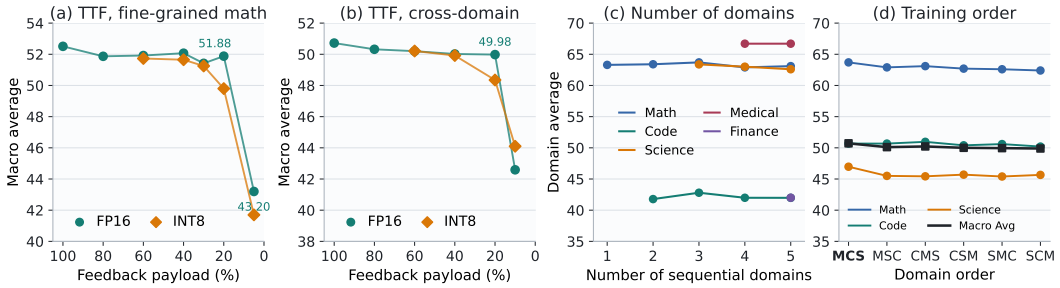


Figure 2: (a, b) Accuracy versus feedback payload for TTF in FP16 and INT8; an INT8 run that scores a fraction n of the tokens sits at $n/2$. (c) Per-domain averages (one public benchmark and the in-domain test set) as sequential clients are added in the order Math, Code, Science, Medical, Finance. (d) The six orders of Math, Code, and Science; M-C-S is the default.

frozen before distillation (Appendix B.3); on the math benchmarks every expert exceeds the 3B student base by more than 15 points, so each client holds knowledge the generalist lacks. Clients are visited in the default orders Algebra→Geometry→Number Theory and Math→Code→Science, with 40 (3B) or 160 (7B) optimizer steps per domain; FedOPD includes SPBO unless stated otherwise. For efficiency we train LoRA adapters; the protocol itself is agnostic to the parameterization (Appendix A.1).

Baselines. All methods share the student initialization, prompts, client experts, domain order, and step budget; they differ in what leaves the client (column *Upload* in Table 1; details in Appendix A.2). *Parameter aggregation*: FedPDPO (Zhu et al., 2026b) optimizes preference pairs on the client and aggregates the resulting updates. *Client-generative and outcome-reliant*: DPSDA-FL (Abacha et al., 2024) fine-tunes the student on responses synthesized by the client experts, FedDF (Lin et al., 2020) distills full expert output distributions, and FedGRPO (Zhu et al., 2026a) returns one scalar reward per response. *Non-private references*: Central-GT-SFT and Central-GRPO pool the answers or verifiers on the server and thus fall outside Definition 1. We report Pass@1 with task verifiers (answer matching, option checking, or test execution) and the unweighted macro average (AVG) over benchmarks; communication is the payload accumulated over a full run (Appendix A.3). Unless stated otherwise, every reported number is the mean over three random seeds.

4.2 Main Results

Comparison with federated transfer. Table 1 shows that FedOPD attains the best average in all four settings while its clients upload only token log-probabilities. The strongest federated baseline changes with the setting (DPSDA-FL or FedDF in math, FedDF or FedPDPO across domains), and FedOPD exceeds it by 1.17 and 1.38 points in math (3B, 7B) and by 2.03 and 3.42 points across domains. The two federated routes fall short for different reasons. FedGRPO, whose clients return

one scalar per response, never gains more than 1.5 points over the student base, as expected from the variance of sequence-level feedback (Proposition 1(ii)). FedDF and FedPDPO become competitive only by moving full distributions or parameter updates, at about $100\times$ and $660\times$ the payload of FedOPD (Table 3).

Evaluation versus imitation. FedOPD also exceeds Central-GRPO, which accesses the verifiers directly, by 0.77 and 0.53 points in math and by 3.73 and 3.76 points across domains. The contrast with DPSDA-FL is most informative: imitating expert-written responses helps within math but harms the 7B student elsewhere (Code $58.18 \rightarrow 42.33$, Science $48.48 \rightarrow 44.75$), whereas FedOPD improves both (59.33 and 51.57). Because the experts score the student's own trajectories rather than replace them, their knowledge is added without overwriting the generalist's strengths.

4.3 Continual Distillation with SPBO

All variants in Table 2 share the client feedback and objective and differ only in how sequential updates are scheduled or regularized. A single global cosine schedule exhibits both failure modes of (C1): its cross-domain Math average falls from 51.01 after the first block to 48.33, and Science, learned last with a decayed step size, is its weakest domain. Blockwise rejuvenation lifts every domain and importance modulation adds to all three; together, SPBO improves the cross-domain average by 3.98 points over the global schedule and by 2.33 over rejuvenation alone, and outperforms the Fisher, MAS, LwF, and SI regularizers and even Mixed training, which requires all clients to be online at once. The gain is smaller across math sub-fields (2.84 points), whose experts share more knowledge, in line with the overlap term κ_ω of Theorem 1.

4.4 Communication Efficiency with TTF

A full FedOPD run moves only 2.18 MB (Table 3), since the uplink carries one scalar per token; TTF reduces it further (Figure 2a,b). Scoring only the 40% or 20% most uncertain tokens keeps accuracy within 0.75 points of full feedback in both settings, and TTF-20 cuts the feedback uplink by 80% at a cross-domain cost of 0.74 points. Quantizing the retained values to INT8 halves the payload again at little cost: at the same token ratio (TTF-60 and TTF-40), INT8 stays within 0.42 points of FP16 in both settings, consistent with the rounding bound of Theorem 2(ii); below about 10% of the payload, accuracy degrades.

4.5 Further Analysis

Signal design. On the math setting, removing trajectory centering, batch whitening, or advantage clipping lowers the average from 52.51 to 51.64, 52.25, and 51.42, respectively. Replacing the per-token gap by its reward-to-go is far worse (45.93), as Proposition 1(ii) predicts for sequence-level token weights. Sensitivity to the learning-rate floor, the PPO clipping range, the gap temperature, and the number of PPO epochs is reported in Appendix C.4.

Number and order of clients. Figure 2c adds a Medical client (MedQA-USMLE, MedMCQA) and a Finance client (FinQA, TAT-QA) one at a time, scoring each domain by the mean of one public benchmark and its in-domain test set (Appendix B.2). Each earlier domain moves by about one point or less, so adding experts does not erode what was learned first. Across the six orders of Math, Code, and Science (Figure 2d), the macro average stays within 49.87–50.72 (Appendix C.3).

5 Conclusion

We presented FedOPD, which turns private domain experts into passive token-level critics: the server samples its own trajectories, and each client returns only teacher-forced log-probabilities, without uploading parameters, generating content, or consulting private outcomes. SPBO extends it to sequentially available clients by rejuvenating the learning rate per domain and modulating updates by parameter importance, and TTF restricts feedback to the tokens on which the student is most uncertain. Across 3B and 7B students, FedOPD surpasses parameter-aggregation and client-generative baselines at about 1% of FedDF's communication. Calibrating noise for a malicious server that may submit arbitrary strings is left to future work.

AI USE STATEMENT

In this work, we did not use generative AI tools for any research activities that require mandatory disclosure. Generative AI tools were used solely for language editing, including improvements to grammar, wording, and overall readability.

Generative AI tools were not used for research ideation, methodology development, theoretical analysis, experimental design, software implementation, data processing, result interpretation, or the formulation of conclusions. All AI-assisted edits were carefully reviewed by the authors to ensure that the technical meaning remained consistent with the authors' original intent.

The authors take full responsibility for the final content of this work.

REFERENCES

Fatima Z Abacha, Sin G Teo, Lucas C Cordeiro, and Mustafa A Mustafa. Synthetic data aided federated learning using foundation models. In *International Workshop on Trustworthy Federated Learning*, pp. 106–118, 2024.

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, 2024.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, 2015.

Frederik Benzing. Unifying importance based regularisation methods for continual learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2022.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *30th USENIX Security Symposium*, pp. 2633–2650, 2021.

Guanzhong Chen, Zhenghan Qin, Mingxin Yang, Yajie Zhou, Tao Fan, Tianyu Du, and Zenglin Xu. Unveiling the vulnerability of private fine-tuning in split-based frameworks for large language models: A bidirectionally enhanced attack. In *Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2904–2918, 2024.

Howard Chen, Noam Razin, Karthik Narasimhan, and Danqi Chen. Retaining by doing: The role of on-policy data in mitigating forgetting. In *International Conference on Machine Learning*, 2026.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V. Le, Sergey Levine, and Yi Ma. SFT memorizes, RL generalizes: A comparative study of foundation model post-training. In *International Conference on Machine Learning*, 2025.

Tao Fan, Yan Kang, Guoqiang Ma, Weijing Chen, Wenbin Wei, Lixin Fan, and Qiang Yang. FATE-LLM: A industrial grade federated learning framework for large language models. *arXiv preprint arXiv:2310.10049*, 2023.

Tao Fan, Hanlin Gu, Xuemei Cao, Chee Seng Chan, Qian Chen, Yiqiang Chen, Yihui Feng, Yang Gu, Jiaxiang Geng, Bing Luo, et al. Ten challenging problems in federated foundation models. *IEEE Transactions on Knowledge and Data Engineering*, 2025.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *International Conference on Learning Representations*, 2024.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Charlie Hou, Mei-Yu Wang, Yige Zhu, Daniel Lazar, and Giulia Fanti. Private federated learning using preference-optimized synthetic data. *arXiv preprint arXiv:2504.16438*, 2025.

Yan Kang, Tao Fan, Hanlin Gu, Xiaojin Zhang, Lixin Fan, and Qiang Yang. Grounding foundation models through federated transfer learning: A general framework. *ACM Transactions on Intelligent Systems and Technology*, 2023.

Daliang Li and Junpu Wang. FedMD: Heterogenous federated learning via model distillation. *arXiv preprint arXiv:1910.03581*, 2019.

Haoran Li, Xinyuan Zhao, Dadi Guo, Hanlin Gu, Ziqian Zeng, Yuxing Han, Yangqiu Song, Lixin Fan, and Qiang Yang. Federated domain-specific knowledge transfer on large language models using synthetic data. *arXiv preprint arXiv:2405.14212*, 2024.

Tao Lin, Lingjing Kong, Sebastian U. Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. In *Advances in Neural Information Processing Systems*, 2020.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pp. 1273–1282, 2017.

Chao Ren, Han Yu, Hongyi Peng, Xiaoli Tang, Bo Zhao, Liping Yi, Alysa Ziying Tan, Yulan Gao, Anran Li, Xiaoxiao Li, et al. Advances and open challenges in federated foundation models. *IEEE Communications Surveys & Tutorials*, 2025.

Xuan Ren, Biao Wu, and Lingqiao Liu. I learn better if you speak my language: Understanding the superior performance of fine-tuning large language models with LLM-generated responses. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024.

Stéphane Ross, Geoffrey J. Gordon, and J. Andrew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pp. 627–635, 2011.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. RL's razor: Why online reinforcement learning forgets less. *arXiv preprint arXiv:2509.04259*, 2025.

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

Thinking Machines Lab. On-policy distillation. `https://thinkingmachines.ai/blog/on-policy-distillation/`, 2025. Blog post.

Zhaorui Yang, Tianyu Pang, Haozhe Feng, Han Wang, Wei Chen, Minfeng Zhu, and Qian Liu. Self-distillation bridges distribution gap in language model fine-tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1028–1043, 2024.

Rui Ye, Wenhao Wang, Jingyi Chai, Dihan Li, Zexi Li, Yinda Xu, Yaxin Du, Yanfeng Wang, and Siheng Chen. OpenFedLLM: Training large language models on decentralized private data via federated learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 6137–6147, 2024.

Liping Yi, Han Yu, Gang Wang, Xiaoguang Liu, and Xiaoxiao Li. pFedLoRA: Model-heterogeneous personalized federated learning with LoRA tuning. *arXiv preprint arXiv:2310.13283*, 2023.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 3987–3995, 2017.

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

Gongxi Zhu, Hanlin Gu, Lixin Fan, Qiang Yang, and Yuxing Han. FedGRPO: Privately optimizing foundation models with group-relative rewards from domain clients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, 2026a.

Kewen Zhu, Liping Yi, Zhiming Zhao, Zhuang Qi, Han Yu, and Qinghua Hu. FedPDPO: Federated personalized direct preference optimization for large language model alignment. *arXiv preprint arXiv:2603.19741*, 2026b.

Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. In *Advances in Neural Information Processing Systems*, 2019.

Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 12878–12889, 2021.