PDF: PATH-DEPENDENT CLIENT SELECTION FOR FEDERATED UNLEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated unlearning (FU) aims to remove target clients' influence from a trained model without full retraining. Existing methods largely treat unlearning as a post-hoc operation, overlooking that client influence is path-dependent: contributions made during critical learning periods (CLPs) can have disproportionate and persistent effects. Moreover, involving all retained clients is costly and impractical because clients differ in their usefulness for unlearning and may be intermittently available. We introduce PDF, a path-dependent client-selection framework that maintains lightweight, target-agnostic, CLP-weighted histories of each client's update direction and global-model alignment. Upon an unlearning request, PDF combines historical global alignment with target decoupling to select top-k retained clients for retraining-based unlearning. Theoretically, we characterize the propagation of client perturbations, derive sufficient conditions for one-step retained-loss reduction and target-loss increase, and bound the initialization selection–utility gap assuming consistency between path-dependence scores and first-order client utility. Experiments on FMNIST, CIFAR-10, and CIFAR-100 under varying data heterogeneity in a label-flipping client-removal setting show that PDF achieves effective forgetting and strong retained utility using only approximately 40% of the retained clients. On non-IID CIFAR-10, PDF achieves 0% forgetting accuracy with an MIA AUC within 0.88 percentage points of Retrain. On CIFAR-100, it improves retained accuracy by up to 6.91 percentage points over the strongest evaluated FU baselines. Across evaluated scenarios, PDF reduces communication by approximately 69% and computation by 57–61% relative to Retrain.

1 INTRODUCTION

As federated learning (FL) gains traction in privacy-sensitive and highly regulated domains (e.g., healthcare) (Theophanous et al., 2026; Hanser et al., 2025), deployed systems must accommodate not only privacy-preserving collaborative training but also the post-training removal of client influence, whether to comply with legal obligations, such as the right to be forgotten under the GDPR (European Parliament & Council of the European Union, 2016), or to mitigate the effects of corrupted, low-quality, or malicious contributions. These operational and regulatory demands have motivated *federated unlearning* (FU), which aims to eliminate the influence of designated target clients from a trained global model while preserving the knowledge contributed by the remaining clients, without costly full retraining.

Existing client-level FU methods, including retraining-based approaches (Liu et al., 2021; Zhang et al., 2023; Xiong et al., 2023) and model-manipulation approaches (Wang et al., 2025b; Khalil et al., 2025; Zhong et al., 2025; Gong et al., 2026; Nguyen et al., 2026a; Weng et al., 2026), face three major limitations in client-level unlearning. First, they largely overlook how client influence develops throughout the original training trajectory. In FL, client participation and update directions vary across rounds; therefore, a client's influence on the final model depends not only on *whether* it participates, but also on *when* it participates. This temporal structure is particularly important during critical learning periods (CLPs), in which client updates can have disproportionately persistent effects on learned representations, optimization trajectory, and final model performance (Achille et al., 2018; Kleinman et al., 2024; Yan et al., 2023; 2022). Moreover, under non-independent and identically distributed (non-IID) data settings, changing participation, or adaptive optimization, critical phases may emerge beyond a fixed early-training window (Huang & Liu, 2025; Wang & Chen,

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.

2026). Second, retraining-based FU methods can incur substantial computation, communication, and storage costs because they reconstruct or recalibrate the model using all retained clients, whose continued availability after training is not guaranteed at all. Third, efficient model-manipulation methods can exhibit a pronounced tension between removing target client influence and preserving retained knowledge (Gong et al., 2026; Chundawat et al., 2024; Weng et al., 2026), and some require costly server-side operations or advance preparation for future unlearning requests (Chundawat et al., 2024; Gong et al., 2026; Zhong et al., 2025; Wang et al., 2025b; Nguyen et al., 2026a).

Taken together, these limitations reveal a missing perspective in existing FU: **the historical structure through which client influence is accumulated is rarely exploited to guide the unlearning process**. We therefore formulate client-level FU from a **path-dependent perspective**. Rather than characterizing a client's influence only through the final trained model or its most recent contribution, we posit that its influence depends on its training history: when the client participated, whether its contributions occurred during `CLPs`, how consistently its updates aligned with the global learning trajectory, and how strongly its accumulated direction became coupled with other clients. This perspective leads to our central hypothesis: **retained clients are not equally valuable for subsequent unlearning**. Clients whose historical contributions consistently aligned with the global trajectory can be particularly useful for preserving retained knowledge, whereas retained clients whose histories are strongly coupled with the target set may carry overlapping information, increasing the likelihood of restoring target-related knowledge during retraining-based unlearning. This raises the central question of this work: ***Can path-dependent training histories identify a subset of retained clients that are particularly useful for unlearning, thereby improving the trade-off among target forgetting, retained utility, and unlearning efficiency?***

To answer this question, we propose a **P**ath-**D**ependent client selection framework for **F**U, named `PDF`. During training, `PDF` maintains lightweight, target-agnostic histories that summarize each client's update direction and alignment with the global optimization trajectory, while placing greater emphasis on contributions made during detected `CLPs`. These statistics are accumulated online and therefore do not require future unlearning targets to be known in advance. Once an unlearning request reveals the target set, `PDF` compares retained clients' historical directions with the aggregate target history to quantify their target coherence. Historical global alignment and target decoupling are then combined into a unified path-dependence (PD) score, from which `PDF` selects top-k retained clients for a short retraining-based unlearning procedure. In this way, `PDF` favors retained clients that are globally informative yet weakly coupled with the target, reducing client participation, communication, and computation while balancing retained-knowledge preservation against target forgetting.

To summarize, our key contributions are as follows: (1) We introduce a path-dependent perspective on retained-client selection for client-level FU, relating clients' usefulness for unlearning to when and how they contributed during training and how strongly their historical updates are coupled with target clients; (2) We propose `PDF`, a `CLP`-aware framework that maintains lightweight, target-agnostic training histories and combines historical global alignment with target decoupling to rank retained clients, enabling retraining-based unlearning with only a top-k subset; (3) We establish a theoretical foundation for PD client selection by showing how subsequent training rounds propagate client perturbations to the final model, deriving sufficient alignment conditions for selected-client updates to simultaneously reduce retained loss and increase target loss, and proving that, under the stated PD–utility consistency condition, `PDF`'s top-k subset achieves a summed first-order utility within $2k\epsilon$ of the optimal size-k subset at the start of unlearning; (4) Experiments on FMNIST, CIFAR-10, and CIFAR-100 under varying data heterogeneity demonstrate effective forgetting and strong retained utility with fewer retained clients and substantially lower communication and computation.

2 Related work

Machine and Federated Unlearning. Machine unlearning aims to remove the influence of designated training data without full retraining (Cao & Yang, 2015). Existing approaches modify either the training process or the learned model through mechanisms such as data partitioning, parameter transformation, pruning, or cached training states (Golatkar et al., 2020; Jia et al., 2023; Brophy & Lowd, 2021). These assume access to training data or intermediate training states that may fail in FL, where data remain distributed across clients (McMahan et al., 2017). FU extends this objective to client-, sample-, or class-level removal (Romandini et al., 2024; Jeong et al., 2026). Particularly,

client-level FU aims to remove the entire influence of target clients, encompassing many samples and classes whose influence has accumulated across multiple training rounds. Existing methods either approximate retraining using historical updates or training-stage artifacts (Liu et al., 2021; Zhang et al., 2023; Xiong et al., 2023), or suppress target influence through parameter-, layer-, gradient-, or representation-level manipulation (Chundawat et al., 2024; Khalil et al., 2025; Gong et al., 2026; Nguyen et al., 2026a; Pan et al., 2025; Weng et al., 2026). While these methods exploit training information or model structure to remove target influence, they do not explicitly model how the training trajectory determines the request-specific value of retained clients for subsequent unlearning.

Critical Learning Periods. `CLPs` are training intervals during which updates exert disproportionately persistent effects on learned representations and final model performance (Achille et al., 2018; Kleinman et al., 2024). In FL, analogous critical rounds emerge, reflecting the temporally non-uniform nature of client influence (Yan et al., 2022; 2023). Existing `CLP`-aware methods leverage these dynamics to improve client participation, aggregation, and training efficiency (Yan et al., 2023; Huang & Liu, 2025; Wang & Chen, 2026). These methods improve the learning process itself, but do not examine whether critical-round information can later help identify retained clients that are useful for unlearning.

Positioning of `PDF`. `PDF` connects these two lines of work by exploiting path-dependent, `CLP`-weighted training histories to estimate the request-specific value of retained clients and select a top-k subset for retraining-based unlearning. Unlike prior FU methods, PDF focuses on which retained clients should participate in unlearning, rather than introducing a new unlearning algorithm.

3 Problem Setup

3.1 Federated Learning Setting

We consider an FL system with a central server and N clients indexed by $\mathcal{I} = \{1, \dots, N\}$. Each client $i \in \mathcal{I}$ holds a private dataset $\mathcal{D}_i = \{(x_j^{(i)}, y_j^{(i)})\}_{j=1}^{n_i}$ of n_i samples. Client datasets are record-disjoint and contain $n = \sum_{i\in\mathcal{I}} n_i$ samples in total. Let $\ell(\boldsymbol{w}; x, y) \in \mathbb{R}_{\geq 0}$ be the sample-wise loss for global model parameters $\boldsymbol{w} \in \mathbb{R}^d$. The local empirical objective function of client i is

$$F_i(\boldsymbol{w}) = \frac{1}{n_i} \sum\nolimits_{j=1}^{n_i} \ell\Big(\boldsymbol{w}; x_j^{(i)}, y_j^{(i)}\Big). \tag{1}$$

With $\pi_i = n_i/n$, $\sum_{i\in\mathcal{I}} \pi_i = 1$, the global objective function is $F(\boldsymbol{w}) = \sum_{i\in\mathcal{I}} \pi_i F_i(\boldsymbol{w})$.

Training proceeds for T_{learn} communication rounds using FedAvg (McMahan et al., 2017). At round t, the server samples $\mathcal{S}^{(t)} \subseteq \mathcal{I}$ with $|\mathcal{S}^{(t)}| = M$, and broadcasts $\boldsymbol{w}^{(t-1)}$. Each participating client performs E local gradient steps with local learning rate $\eta_{\ell,t} > 0$, yielding the local model $\boldsymbol{h}_i^{(t)}(\boldsymbol{w}^{(t-1)})$. The server aggregates

$$\boldsymbol{w}^{(t)} = (1 - \gamma_t)\boldsymbol{w}^{(t-1)} + \gamma_t \sum\nolimits_{i\in\mathcal{S}^{(t)}} p_i^{(t)} \boldsymbol{h}_i^{(t)}(\boldsymbol{w}^{(t-1)}), \tag{2}$$

where $p_i^{(t)} = n_i / \sum_{m\in\mathcal{S}^{(t)}} n_m$ and standard FedAvg corresponds to $\gamma_t = 1$. We denote round-dependent participation by $a_i^{(t)} = \mathbf{1}\{i \in \mathcal{S}^{(t)}\}$. After T_{learn} rounds, the server obtains the trained global model $\boldsymbol{w}^{(T_{\text{learn}})}$, together with the training history which is recorded during training and is target-agnostic: it is constructed *independently* of $\mathcal{I}_{\text{tar}}$ and any subsequent unlearning request (see details in Section 4).

3.2 Federated Unlearning problem

After training, a nonempty target-client set $\mathcal{I}_{\text{tar}} \subseteq \mathcal{I}$ is designated for removal, and the retained set is $\mathcal{I}_{\text{ret}} = \mathcal{I} \setminus \mathcal{I}_{\text{tar}}$. Let $n_{\text{ret}} = \sum_{i\in\mathcal{I}_{\text{ret}}} n_i$ and $n_{\text{tar}} = \sum_{o\in\mathcal{I}_{\text{tar}}} n_o$. The corresponding objectives are

$$F_{\text{ret}}(\boldsymbol{w}) = \sum_{i\in\mathcal{I}_{\text{ret}}} \frac{n_i}{n_{\text{ret}}} F_i(\boldsymbol{w}), \qquad F_{\text{tar}}(\boldsymbol{w}) = \sum_{o\in\mathcal{I}_{\text{tar}}} \frac{n_o}{n_{\text{tar}}} F_o(\boldsymbol{w}). \tag{3}$$

Retraining from scratch after removing the target clients provides the gold reference for client-level FU. Let $\mathcal{A}(\cdot; \xi)$ denote the original FL procedure under randomness ξ, including initialization,

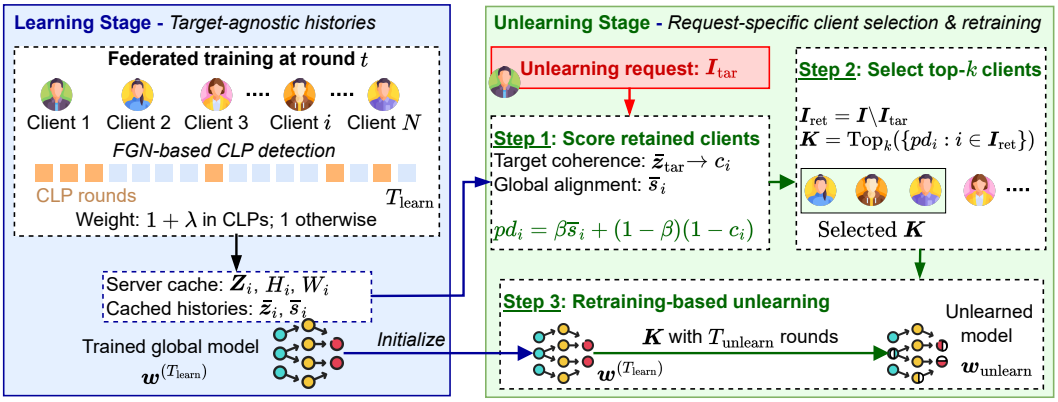


Figure 1: PDF framework with workflow.

client sampling, and local optimization. The retraining reference is $\boldsymbol{w}^{\star}_{\text{retrain}} = \mathcal{A}(\{\mathcal{D}_i\}_{i\in\mathcal{I}_{\text{ret}}}; \xi)$. Given $\boldsymbol{w}^{(T_{\text{learn}})}$, $\mathcal{I}_{\text{tar}}$, and the training history, an unlearning algorithm returns $\boldsymbol{w}_{\text{unlearn}}$. We seek retraining-like behavior on both retained and target objectives:

$$F_{\text{ret}}(\boldsymbol{w}_{\text{unlearn}}) - F_{\text{ret}}(\boldsymbol{w}^{\star}_{\text{retrain}}) \leq \delta, \qquad |F_{\text{tar}}(\boldsymbol{w}_{\text{unlearn}}) - F_{\text{tar}}(\boldsymbol{w}^{\star}_{\text{retrain}})| \leq \varepsilon_f, \tag{4}$$

where $\delta \geq 0$ and $\varepsilon_f \geq 0$ denote the retained-utility and forgetting tolerances, respectively. The challenge is to approach this retraining reference without requiring all retained clients to participate. We address this problem by using path-dependent training histories to identify a small retained subset $\mathcal{K} \subseteq \mathcal{I}_{\text{ret}}$ for retraining-based unlearning.

4 Path-Dependent Client Selection for Federated Unlearning

PDF addresses the client-selection problem defined in Section 3 through two stages, as illustrated in Figure 1. During federated training, the server detects CLPs and maintains lightweight, target-agnostic histories describing each client's accumulated update direction and alignment with global training trajectory. Once an unlearning request reveals the target set $\mathcal{I}_{\text{tar}}$, these histories are used to quantify each retained client's coherence with the target and construct a path-dependence score. The server then selects the top-k retained clients and restricts the subsequent retraining-based unlearning process to this subset. All historical statistics are computed using the classification layer (Wang et al., 2025a; Li et al., 2023; Oh et al., 2022; Yi et al., 2024; Luo et al., 2021), providing a compact representation of client-specific update behavior without the cost of storing per-round model checkpoints. Appendix A.4.3 reports a Spearman correlation of 0.7022 between classification-layer and full-model client utilities in the evaluated FMNIST setting, providing empirical support for the compact representation.

4.1 Learning-Stage History Construction

Critical learning period detection. Client influence can vary substantially across training rounds. Therefore, PDF gives greater importance to contributions made during detected CLPs. Following (Yan et al., 2023), we use the federated gradient norm (FGN) as an online proxy for detecting CLPs. Let $\boldsymbol{g}_i^{(t)} = \nabla F_i(\boldsymbol{w}^{(t-1)})$ denote the gradient of participating client i. Under a first-order approximation, a local gradient step produces a loss decrease proportional to $\eta_{\ell,t}\|\boldsymbol{g}_i^{(t)}\|_2^2$. As a result, we define $\text{FGN}^{(t)} \triangleq \eta_{\ell,t}\sum_{i\in\mathcal{S}^{(t)}} p_i^{(t)} \|\boldsymbol{g}_i^{(t)}\|_2^2$. A training round is classified as critical when $\frac{\text{FGN}^{(t)} - \text{FGN}^{(t-1)}}{\text{FGN}^{(t-1)}} \geq \tau_{\text{CLP}}$, where $\tau_{\text{CLP}} > 0$ is an empirically selected sensitivity threshold. Because detection is performed throughout training, the set $\mathcal{T}_{\text{CLP}}$ of detected CLPs may contain critical periods at different training stages. Appendix A.2 illustrates CLP detection.

Historical global alignment. To characterize whether a client's contribution consistently supports the realized global training trajectory, we compare its classification-layer update with the correspond-

ing aggregated global update, and define them by $\boldsymbol{\delta}_i^{\mathrm{cls},(t)} = \boldsymbol{\theta}_i^{\mathrm{cls},(t)} - \boldsymbol{\theta}^{\mathrm{cls},(t-1)}$, $\boldsymbol{\delta}_g^{\mathrm{cls},(t)} = \boldsymbol{\theta}^{\mathrm{cls},(t)} - \boldsymbol{\theta}^{\mathrm{cls},(t-1)}$, respectively. The round-level global alignment of client i is: $s_i^{(t)} = \widehat{\cos}\Big(\boldsymbol{\delta}_i^{\mathrm{cls},(t)}, \boldsymbol{\delta}_g^{\mathrm{cls},(t)}\Big)$, where $\widehat{\cos}(\boldsymbol{a}, \boldsymbol{b}) = \frac{1}{2}\Big(1 + \frac{\langle \boldsymbol{a}, \boldsymbol{b}\rangle}{(\|\boldsymbol{a}\|_2+\varepsilon_0)(\|\boldsymbol{b}\|_2+\varepsilon_0)}\Big)$ is the normalized cosine similarity and $\varepsilon_0 > 0$ ensures numerical stability. A larger $s_i^{(t)}$ indicates stronger consistency between client i's local update and the realized global update at round t.

CLP-weighted history. To emphasize contributions made during CLPs, we assign each training round the temporal weight: $\phi^{(t)} = \begin{cases} 1+\lambda, & t \in \mathcal{T}_{\mathrm{CLP}}, \\ 1, & \text{otherwise}, \end{cases}$ where $\lambda \geq 0$ controls the additional importance assigned to CLP contributions. For each participating client, we normalize its classification-layer update as follows: $\widehat{\boldsymbol{\delta}}_i^{\mathrm{cls},(t)} = \frac{\boldsymbol{\delta}_i^{\mathrm{cls},(t)}}{\|\boldsymbol{\delta}_i^{\mathrm{cls},(t)}\|_2+\varepsilon_0}$. The server maintains three target-agnostic statistics for each client: an accumulated update direction $\boldsymbol{Z}_i^{(t)}$, an accumulated global-alignment score $H_i^{(t)}$, and a participation weight $W_i^{(t)}$. Initialized as $\boldsymbol{Z}_i^{(0)} = \mathbf{0}$, $H_i^{(0)} = 0$, and $W_i^{(0)} = 0$, they are updated online as: $\boldsymbol{Z}_i^{(t)} = \boldsymbol{Z}_i^{(t-1)} + \phi^{(t)} a_i^{(t)} \widehat{\boldsymbol{\delta}}_i^{\mathrm{cls},(t)}$, $H_i^{(t)} = H_i^{(t-1)} + \phi^{(t)} a_i^{(t)} s_i^{(t)}$, $W_i^{(t)} = W_i^{(t-1)} + \phi^{(t)} a_i^{(t)}$. After T_{learn}, we obtain the client's historical update direction and historical global alignment as

$$\boldsymbol{z}_i = \frac{\boldsymbol{Z}_i^{(T_{\mathrm{learn}})}}{W_i^{(T_{\mathrm{learn}})} + \varepsilon_0}, \quad \bar{\boldsymbol{z}}_i = \frac{\boldsymbol{z}_i}{\|\boldsymbol{z}_i\|_2 + \varepsilon_0}, \quad \bar{s}_i = \frac{H_i^{(T_{\mathrm{learn}})}}{W_i^{(T_{\mathrm{learn}})} + \varepsilon_0}. \tag{5}$$

Here, $\bar{\boldsymbol{z}}_i$ summarizes the dominant direction of client i's historical updates, while $\bar{s}_i$ measures how consistently its updates aligned with the global training trajectory. Because $\phi^{(t)}$ is larger during detected CLPs, both statistics place greater emphasis on contributions made during critical rounds. These histories are target-agnostic and require one d_{cls}-dimensional vector and two scalars per client.

4.2 Request-Time Client Selection and Retraining-Based Unlearning

Target coherence. Once an unlearning request reveals the target-client set $\mathcal{I}_{\mathrm{tar}}$, PDF aggregates the normalized historical update directions of the target clients as $\bar{\boldsymbol{z}}_{\mathrm{tar}} = \frac{\sum_{o\in\mathcal{I}_{\mathrm{tar}}} \bar{\boldsymbol{z}}_o}{\|\sum_{o\in\mathcal{I}_{\mathrm{tar}}} \bar{\boldsymbol{z}}_o\|_2+\varepsilon_0}$. For each retained client $i \in \mathcal{I}_{\mathrm{ret}}$, its target coherence is $c_i(\mathcal{I}_{\mathrm{tar}}) = \cos(\bar{\boldsymbol{z}}_i, \bar{\boldsymbol{z}}_{\mathrm{tar}})$. A larger $c_i(\mathcal{I}_{\mathrm{tar}})$ indicates stronger directional coherence between the retained client's historical updates and those of the target set, whereas $1 - c_i(\mathcal{I}_{\mathrm{tar}})$ represents target decoupling. *Selecting retained clients with larger target decoupling reduces the tendency of retained-client updates to restore target-related knowledge during the subsequent unlearning process* (see Table 3).

Path-dependence score. PDF combines historical global alignment and target decoupling to quantify the request-specific value of each retained client:

$$pd_i(\mathcal{I}_{\mathrm{tar}}) = \beta \bar{s}_i + (1-\beta)\big[1 - c_i(\mathcal{I}_{\mathrm{tar}})\big], \tag{6}$$

where $\beta \in [0, 1]$ controls their relative importance. Increasing β places greater emphasis on retained-knowledge preservation, whereas decreasing β prioritizes clients whose historical directions are weakly coupled with those of the target clients, thereby favoring target forgetting. Then, the server selects the k highest-scoring retained clients, that tend to be both globally informative and weakly coupled with the target history: $\mathcal{K} = \mathrm{Top}_k\left(\{pd_i(\mathcal{I}_{\mathrm{tar}}) : i \in \mathcal{I}_{\mathrm{ret}}\}\right)$, with $k \leq |\mathcal{I}_{\mathrm{ret}}|$.

Retraining-based unlearning. Starting from the trained global model $\boldsymbol{w}^{(T_{\mathrm{learn}})}$, PDF performs T_{unlearn} rounds of retraining-based FL using only clients in $\mathcal{K}$ and target clients do not participate. The underlying local optimization and server aggregation remain unchanged from the standard FL procedure. Hence, PDF modifies which retained clients participate in unlearning, rather than introducing a new retraining algorithm.

4.3 Theoretical Analysis

We provide theoretical support for three aspects of PDF: (i) Client influence is contingent on the learning trajectory; (ii) the effect of a selected retained-client subset can be characterized through its

alignment with retained and target objectives; and (iii) PD-based top-k selection is near-optimal when the learning-stage score is sufficiently consistent with request-time client utility. We assume standard smoothness conditions, and their formal statements and all proofs are provided in Appendix A.3.

Path-dependent client influence. Consider a client $o \in \mathcal{I}_{\text{tar}}$ and a counterfactual (i.e., leave-one-client-out) learning trajectory $\boldsymbol{w}_{-o}^{(t)}$ generated under the same initialization and training randomness as the factual trajectory, except that client o is excluded whenever it participates. We define $\boldsymbol{\Delta}_o^{(t)} = \boldsymbol{w}^{(t)} - \boldsymbol{w}_{-o}^{(t)}$ as the accumulated model deviation attributable to client o, and let $\boldsymbol{b}_o^{(t)}$ denote its direct perturbation. Under smoothness assumption, the local update map is $\rho_t = (1 + \eta_{\ell,t} L)^E$-Lipschitz.

Proposition 1 (Path-dependent client influence) *The final deviation caused by client o satisfies*

$$\left\| \boldsymbol{\Delta}_o^{(T_{\text{learn}})} \right\|_2 \leq \sum_{t=1}^{T_{\text{learn}}} \omega_t \left\| \boldsymbol{b}_o^{(t)} \right\|_2 , \tag{7}$$

where $\omega_t = \prod_{s=t+1}^{T_{\text{learn}}} \Gamma_s$, $\Gamma_t = 1 - \gamma_t + \gamma_t \rho_t$, and the empty product equals one.

Proposition 1 shows that a client contribution's final influence depends on its direct round-level perturbation $\left\| \boldsymbol{b}_o^{(t)} \right\|_2$ and how that perturbation propagates through the remaining training rounds, as upper-bounded by ω_t. Therefore, client contributions with similar magnitudes may have different effects on the final model depending on when they occur, motivating temporally weighted training histories rather than treating all rounds uniformly. `PDF` uses FGN as an online indicator of potentially critical rounds. Appendix A.4.1 formulates a `CLP`-influence hypothesis using propagated upper-bound client contributions (see Hypothesis 1). Complementary one-round client-omission counterfactual experiments show that *contributions made during detected `CLP` rounds have greater average final-model parameter influence than those made during `non-CLP` rounds* (see Table 5 in Appendix A.4.1).

Selected-Client Utility and Forgetting. At an unlearning iterate $\boldsymbol{w}$, let $\boldsymbol{v}_i$ denote the local unlearning update produced by selected retained client i. For a selected subset $\mathcal{K}$, the aggregated local unlearning direction and server update are given by $\boldsymbol{v}_{\mathcal{K}} = \sum_{i \in \mathcal{K}} q_i \boldsymbol{v}_i$, $q_i \geq 0$, $\sum_{i \in \mathcal{K}} q_i = 1$, and $\boldsymbol{w}^+ = \boldsymbol{w} - \eta_u \boldsymbol{v}_{\mathcal{K}}$, where $\eta_u > 0$ is the unlearning rate. We characterize this update through its retained- and target-gradient alignments: $A_{\mathcal{K}}(\boldsymbol{w}) = \langle \nabla F_{\text{ret}}(\boldsymbol{w}), \boldsymbol{v}_{\mathcal{K}} \rangle$ and $C_{\mathcal{K}}(\boldsymbol{w}) = \langle \nabla F_{\text{tar}}(\boldsymbol{w}), \boldsymbol{v}_{\mathcal{K}} \rangle$.

Proposition 2 (One-step retained and target objective bounds) *Under Assumption 2, we have*

$$F_{\text{ret}}(\boldsymbol{w}^+) - F_{\text{ret}}(\boldsymbol{w}) \leq -\eta_u A_{\mathcal{K}}(\boldsymbol{w}) + \frac{L_{\text{ret}} \eta_u^2}{2} \left\| \boldsymbol{v}_{\mathcal{K}} \right\|_2^2 , \tag{8}$$

$$F_{\text{tar}}(\boldsymbol{w}^+) - F_{\text{tar}}(\boldsymbol{w}) \geq -\eta_u C_{\mathcal{K}}(\boldsymbol{w}) - \frac{L_{\text{tar}} \eta_u^2}{2} \left\| \boldsymbol{v}_{\mathcal{K}} \right\|_2^2 . \tag{9}$$

Therefore, sufficient conditions for a decrease in the retained objective and an increase in the target objective are, respectively, as follows:

$$A_{\mathcal{K}}(\boldsymbol{w}) > \frac{L_{\text{ret}} \eta_u}{2} \left\| \boldsymbol{v}_{\mathcal{K}} \right\|_2^2 , \quad C_{\mathcal{K}}(\boldsymbol{w}) < -\frac{L_{\text{tar}} \eta_u}{2} \left\| \boldsymbol{v}_{\mathcal{K}} \right\|_2^2 . \tag{10}$$

Proposition 2 quantifies how selected-client alignment affects the one-step retained and target objectives. Positive retained alignment and negative target alignment favor a retained-loss decrease and a target-loss increase to first order, while Eq. 10 provides sufficient conditions. These bounds motivate the request-time client utility introduced below. The proof is provided in Appendix A.3.3, while Appendix A.4.2 reports empirical retained- and target-gradient alignments together with the observed loss changes.

Conditional near-optimality of PD-based selection. At the client selection point $\boldsymbol{w} = \boldsymbol{w}^{(T_{\text{learn}})}$, we define the first-order utility of retained client i as follows:

$$Q_i(\boldsymbol{w}) = \langle \nabla F_{\text{ret}}(\boldsymbol{w}), \boldsymbol{v}_i \rangle - \mu \langle \nabla F_{\text{tar}}(\boldsymbol{w}), \boldsymbol{v}_i \rangle , \tag{11}$$

where $\mu \geq 0$ balances retained-utility preservation and target forgetting, as motivated by Proposition 2. However, evaluating $Q_i(\boldsymbol{w})$ requires current client unlearning updates and objective gradients. `PDF` therefore uses the learning-stage path-dependence score $pd_i(\mathcal{I}_{\text{tar}})$ as a selection proxy.

Assumption 1 (PD–utility consistency) *At $\boldsymbol{w}_0 = \boldsymbol{w}^{(T_{\text{learn}})}$, there exist $\alpha_Q > 0$, $b_Q \in \mathbb{R}$, and $\epsilon \geq 0$:*

$$|Q_i(\boldsymbol{w}_0) - (\alpha_Q pd_i(\mathcal{I}_{\text{tar}}) + b_Q)| \leq \epsilon, \quad \forall i \in \mathcal{I}_{\text{ret}}. \tag{12}$$

This condition requires the same affine approximation-error bound ϵ for every retained client.

Corollary 1 (Conditional near-optimality of PD-based selection) *Let $\mathcal{K}$ contain the k highest-scoring retained clients, and let $\mathcal{K}^\star$ maximize the summed initialization utility among all size-k retained subsets. Under Assumption 1, we have*

$$\sum_{i \in \mathcal{K}^\star} Q_i(\boldsymbol{w}_0) - \sum_{i \in \mathcal{K}} Q_i(\boldsymbol{w}_0) \leq 2k\epsilon. \tag{13}$$

Equivalently, the average unlearning utility of the PD-selected subset is within 2ϵ of the maximum achievable by any size-k subset. Corollary 1 characterizes the top-k client selection quality at unlearning initialization, while subsequent unlearning performance is evaluated in Section 5. The proof and empirical evaluation are provided in Appendix A.3.4 and Appendix A.4.3.

5 Experimental evaluation

5.1 Experimental setup

Unlearning protocol and dataset. We consider a label-flipping scenario, where one malicious client is designated for unlearning (Zhong et al., 2025). Experiments use FMNIST (Xiao et al., 2017) with a CNN (see Appendix A.5.2), CIFAR-10 and CIFAR-100 (Krizhevsky et al., 2009) with ResNet-18 (He et al., 2016). Client data are partitioned using Dirichlet distributions with $\alpha \in \{0.1, 1.0\}$, where smaller α shows greater data heterogeneity.

Training and unlearning. We simulate an FL system of $N = 64$ clients, sampling $M = 16$ clients per learning round with $\eta_{l,t} = 0.01$, $T_{\text{learn}} = 150$, $E = 3$, and batch size 32. Unless otherwise stated, PDF uses $\tau_{\text{CLP}} = 0.01$, $\lambda = 0.7$, and $k = 25$ retained clients for $T_{\text{unlearn}} = 20$ rounds. We evaluate $\beta \in \{0.1, 0.5, 0.9\}$ to vary the emphasis on retained knowledge preservation and target forgetting. Unlearning starts from the trained model and uses FedAvg for local optimization and aggregation.

Baselines and metrics. We compare PDF against Retrain, CONDA (Chundawat et al., 2024; Foster et al., 2024), FedRecovery (Zhang et al., 2023), FOUL (Nguyen et al., 2026a), FedEraser (Liu et al., 2021), Exact-Fun (E-F) (Xiong et al., 2023), FUSED (Zhong et al., 2025), and GDFA (Weng et al., 2026) (see Appendix A.5.3). We evaluate performance using the following active testing metrics (Nguyen et al., 2026b;a; Zhong et al., 2025; Gu et al., 2024; Chundawat et al., 2024; Weng et al., 2026):

- **FA & RA** (%): Forgetting accuracy (FA) is computed on the target-client data as the mean of class-wise accuracies using flipped labels (see Appendix A.5.1), where lower FA indicates stronger removal of target influence. Retained accuracy (RA) measures accuracy on the retained data, where higher RA indicates better preservation of retained knowledge.
- **MIA AUC** (%): We assess residual privacy leakage using membership inference attacks (MIA) (Shokri et al., 2017) (see details in Appendix A.5.1). AUC measures member–non-member distinguishability across decision thresholds, with 0.5 (50%) corresponding to random discrimination. We use Retrain as the reference and report the MIA-AUC gap to Retrain, where a smaller gap indicates more retraining-like privacy behavior.
- **Comm** (GB) & **Comp** (PFLOPs): Total data transmitted and total floating-point operations (FLOPs), respectively, each aggregated over training and unlearning. We include training because several learning-to-unlearn baselines modify the training protocol, so measuring the unlearning stage alone would understate their cost.
- **Space Complexity**: We measure the additional memory required to store model checkpoints, historical information, and intermediate model updates during training and unlearning.

Table 1: FU performance on three datasets under $\alpha \in \{0.1, 1.0\}$. All PDF variants select $k = 25$ clients and subscripts denote $\beta \in \{0.1, 0.5, 0.9\}$ (forgetting-oriented, balanced, and utility-oriented, respectively). Parentheses report differences from Retrain. **Red** and *green* indicate the best and the second-best unlearning results, excluding Retrain, ranked by FA (↓), RA (↑), and |ΔMIA| (↓).

Metric	α	Retrain	FUSED	CONDA	E-F	FedEraser	FOUL	FedRecovery	GDFA	$\text{PDF}_{0.1}$	$\text{PDF}_{0.5}$	$\text{PDF}_{0.9}$
FMNIST-CNN												
FA	0.1	0.44	0.44 (+0.00)	0.70 (+0.26)	0.56 (+0.12)	1.11 (+0.67)	0.75 (+0.31)	0.60 (+0.16)	*0.28 (−0.16)*	**0.00 (−0.44)**	0.72 (+0.28)	0.44 (+0.00)
	1.0	0.92	0.82 (−0.10)	0.94 (+0.02)	0.45 (−0.47)	0.53 (−0.39)	0.77 (−0.15)	0.87 (−0.06)	**0.14 (−0.78)**	*0.28 (−0.64)*	*0.28 (−0.64)*	0.92 (+0.00)
RA	0.1	88.20	**87.14 (−1.06)**	85.17 (−3.02)	81.89 (−6.31)	61.04 (−27.16)	85.60 (−2.60)	85.22 (−2.98)	81.83 (−6.37)	82.60 (−5.60)	85.97 (−2.23)	*86.89 (−1.31)*
	1.0	90.45	*89.60 (−0.85)*	89.40 (−1.05)	85.94 (−4.50)	76.42 (−14.03)	**89.64 (−0.81)**	**89.64 (−0.81)**	88.25 (−2.20)	89.34 (−1.11)	89.35 (−1.10)	89.55 (−0.90)
MIA AUC	0.1	50.32	*50.05 (−0.27)*	**50.11 (−0.20)**	49.84 (−0.48)	49.92 (−0.40)	49.71 (−0.61)	50.01 (−0.31)	50.00 (−0.32)	50.01 (−0.31)	50.00 (−0.32)	49.91 (−0.41)
	1.0	50.23	49.79 (−0.44)	50.09 (−0.15)	**50.25 (+0.02)**	50.08 (−0.15)	49.82 (−0.41)	50.16 (−0.07)	50.17 (−0.06)	50.13 (−0.10)	**50.25 (+0.02)**	*50.19 (−0.04)*
Comm	0.1/1.0	15.62	4.11	3.97	7.82	4.36	6.08	3.97	3.98	4.79	4.79	4.79
Comp	0.1	2.09	1.01 (−1.09)	0.71 (−1.39)	1.22 (−0.87)	0.72 (−1.37)	1.71 (−0.38)	0.71 (−1.39)	0.71 (−1.38)	0.83 (−1.26)	0.88 (−1.22)	0.89 (−1.21)
	1.0	2.10	1.01 (−1.09)	0.71 (−1.39)	1.23 (−0.87)	0.73 (−1.37)	1.72 (−0.38)	0.71 (−1.39)	0.71 (−1.39)	0.83 (−1.27)	0.83 (−1.27)	0.84 (−1.26)
CIFAR-10-ResNet												
FA	0.1	6.93	6.14 (−0.79)	7.52 (+0.59)	2.21 (−4.72)	4.17 (−2.76)	1.86 (−5.07)	0.77 (−6.16)	4.45 (−2.48)	0.42 (−6.51)	**0.00 (−6.93)**	*0.39 (−6.54)*
	1.0	0.72	1.38 (+0.66)	11.18 (+10.46)	3.93 (+3.20)	7.97 (+7.25)	1.96 (+1.24)	5.87 (+5.15)	**0.33 (−0.39)**	*0.89 (+0.17)*	1.09 (+0.37)	0.94 (+0.22)
RA	0.1	55.28	49.85 (−5.43)	14.68 (−40.60)	23.10 (−32.18)	9.68 (−45.60)	52.10 (−3.18)	37.12 (−18.16)	38.61 (−16.67)	45.22 (−10.06)	*54.48 (−0.80)*	**57.10 (+1.82)**
	1.0	86.93	*83.56 (−3.38)*	9.01 (−77.93)	52.40 (−34.53)	24.67 (−62.26)	82.65 (−4.28)	46.83 (−40.10)	69.34 (−17.59)	82.28 (−4.65)	82.84 (−4.09)	**84.08 (−2.85)**
MIA AUC	0.1	55.64	61.43 (+5.79)	62.98 (+7.34)	50.13 (−5.51)	49.87 (−5.78)	60.73 (+5.09)	**55.16 (−0.49)**	57.53 (+1.89)	54.62 (−1.02)	*54.76 (−0.88)*	56.69 (+1.05)
	1.0	52.18	54.30 (+2.12)	53.38 (+1.20)	49.33 (−2.85)	50.54 (−1.65)	53.11 (+0.93)	51.21 (−0.98)	**51.82 (−0.36)**	51.74 (−0.44)	*51.78 (−0.40)*	51.58 (−0.60)
Comm	0.1/1.0	845.48	217.29	214.73	423.39	235.88	329.25	214.73	215.35	259.46	259.46	259.46
Comp	0.1	47.45	22.38 (−25.07)	16.01 (−31.45)	27.69 (−19.76)	16.42 (−31.03)	38.45 (−9.00)	16.01 (−31.45)	16.02 (−31.43)	18.73 (−28.72)	19.30 (−28.15)	20.02 (−27.43)
	1.0	47.63	22.50 (−25.14)	16.10 (−31.53)	27.84 (−19.79)	16.49 (−31.14)	38.66 (−8.97)	16.10 (−31.53)	16.11 (−31.52)	18.70 (−28.93)	18.76 (−28.87)	18.88 (−28.75)
CIFAR-100-ResNet												
FA	0.1	0.27	1.26 (+0.99)	0.86 (+0.59)	1.46 (+1.19)	0.84 (+0.56)	1.50 (+1.23)	0.86 (+0.59)	0.86 (+0.59)	**0.00 (−0.27)**	0.86 (+0.59)	*0.52 (+0.25)*
	1.0	0.60	0.91 (+0.30)	0.54 (−0.07)	*0.40 (−0.21)*	1.25 (+0.64)	1.00 (+0.40)	1.44 (+0.83)	**0.08 (−0.52)**	0.44 (−0.16)	0.91 (+0.31)	0.52 (−0.08)
RA	0.1	55.17	36.34 (−18.83)	4.35 (−50.82)	16.98 (−38.20)	3.98 (−51.19)	42.22 (−12.95)	1.03 (−54.14)	42.15 (−13.02)	*48.63 (−6.54)*	48.43 (−6.74)	**49.13 (−6.04)**
	1.0	58.92	31.83 (−27.09)	2.12 (−56.80)	23.80 (−35.12)	9.10 (−49.82)	39.81 (−19.11)	14.46 (−44.46)	47.50 (−11.42)	**54.38 (−4.54)**	*54.24 (−4.68)*	54.21 (−4.71)
MIA AUC	0.1	51.44	54.02 (+2.59)	55.48 (+4.04)	55.21 (+3.77)	*50.71 (−0.73)*	53.09 (+1.65)	49.96 (−1.48)	52.42 (+0.98)	**52.13 (+0.69)**	52.57 (+1.13)	52.43 (+0.99)
	1.0	62.15	*53.91 (−8.24)*	50.20 (−11.95)	**54.90 (−7.25)**	51.36 (−10.79)	50.49 (−11.66)	50.45 (−11.70)	49.73 (−12.42)	53.03 (−9.12)	52.39 (−9.76)	52.98 (−9.17)
Comm	0.1/1.0	848.97	218.19	215.61	425.14	236.86	330.60	215.61	216.24	260.53	260.53	260.53
Comp	0.1	47.75	22.63 (−25.11)	16.22 (−31.53)	28.02 (−19.72)	16.62 (−31.13)	38.93 (−8.82)	16.22 (−31.53)	16.23 (−31.52)	18.83 (−28.92)	18.87 (−28.88)	18.96 (−28.78)
	1.0	47.77	22.59 (−25.18)	16.17 (−31.60)	27.96 (−19.81)	16.57 (−31.20)	38.82 (−8.95)	16.17 (−31.60)	16.18 (−31.59)	18.72 (−29.05)	18.73 (−29.04)	18.75 (−29.02)

5.2 Main Results

Table 1 shows that PDF achieves effective forgetting and strong retained utility using only $k = 25$ unlearning clients. On FMNIST, PDF reaches 86.89% RA under $\alpha = 0.1$, while the forgetting-oriented variant achieves 0% FA. Its advantage is clearer on the CIFAR tasks. On CIFAR-10 with $\alpha = 0.1$, $\text{PDF}_{0.9}$ achieves 57.10% RA, exceeding Retrain (55.28%) and FOUL (52.10%), while $\text{PDF}_{0.5}$ combines 54.48% RA with 0% FA. On CIFAR-100, all three variants outperform the strongest competing unlearning baseline in RA under both data settings, reaching up to 49.13% versus 42.22% at $\alpha = 0.1$ and 54.38% versus 47.50% at $\alpha = 1.0$. Across PDF variants, MIA AUC differs from Retrain by at most 1.13 percentage points in all settings except CIFAR-100 with $\alpha = 1.0$, suggesting similar behavior under the evaluated attack. In that setting, AUC is 9.12–9.76 points below Retrain, indicating a remaining gap in matching the retrained reference. ***These results support the central hypothesis that retained clients differ in their unlearning value, and that exploiting path-dependent client selection enables better retained utility preservation with limited participation***. Increasing β favors historical global alignment and preserves retained utility, whereas smaller β prioritizes target decoupling. The weaker variation on CIFAR-100 indicates lower sensitivity to β in this setting. Relative to Retrain, PDF also reduces communication by approximately 69% and computation by about 57–61%, demonstrating a favorable trade-off between unlearning efficacy and efficiency.

As shown in Table 2, PDF is space-efficient because it stores only one accumulated classification-layer update and two scalars per client, requiring $\mathcal{O}(Nd_{\text{cls}})$ storage independent of the number of learning rounds, where $d_{\text{cls}} \ll P$. During unlearning, PDF operates on k retained clients, so the model and update tensors instantiated during computation scale with $\mathcal{O}(kP)$, where $k < |\mathcal{I}_{\text{ret}}|$. Retrain and FUSED involve all retained clients, although FUSED reduces per-client storage and computation using lightweight LoRA parameters. CONDA and FOUL maintain client-wise full-model information, while GDFA constructs and fine-tunes multiple perturbed models. FedEraser stores checkpointed model histories, whereas E-F and FedRecovery retain full-model information across learning rounds.

Table 2: Space complexity analysis.

Method	Space Complexity
Retrain	$\mathcal{O}(P + \lvert\mathcal{I}_{\text{ret}}\rvert \cdot P)$
FUSED	$\mathcal{O}(P + \lvert\mathcal{I}_{\text{ret}}\rvert \cdot A)$
CONDA	$\mathcal{O}(N \cdot P + P)$
E-F	$\mathcal{O}(T_{\text{learn}} \cdot M \cdot P + P)$
FedEraser	$\mathcal{O}(H \cdot M \cdot P + \lvert\mathcal{I}_{\text{ret}}\rvert \cdot P)$
FOUL	$\mathcal{O}(P + N \cdot P + N^2)$
FedRecovery	$\mathcal{O}(T_{\text{learn}} \cdot M \cdot P + P)$
GDFA	$\mathcal{O}(K_{\text{per}} \cdot P + \lvert\mathcal{I}_{\text{tar}}\rvert \cdot P)$
PDF	$\mathcal{O}(N \cdot d_{\text{cls}} + k \cdot P)$

* $P = |\boldsymbol{w}|$ is the number of full-model parameters.
† A is the number of LoRA unlearning parameters, $A \ll P$.
†† $d_{\text{cls}} \ll P$.
‡ H is the number of FedEraser checkpoints, $H < T_{\text{learn}}$.
• K_{per} is the number of GDFA perturbed models.
◇ k is the number of retained clients selected by PDF, $k < |\mathcal{I}_{\text{ret}}|$.

5.3 Ablation study

Variants of PDF. Table 3 evaluates the proposed client selection rule while holding the unlearning procedure fixed. Under $\alpha = 0.1$, PDF improves RA over reversed ranking approach (Rev) by 3.69 percentage points on FMNIST and 13.39 points on CIFAR-10. Selecting highly target-coherent clients (high−Coh) increases CIFAR-10 FA from 0% to 7.65% and reduces RA from 54.48% to 37.65%, as compared with PDF. Moreover, removing CLP weighting (non−CLP) reduces RA by 0.62 points on FMNIST and 4.88 points on CIFAR-10. Under $\alpha = 1.0$, the differences are smaller, although PDF retains the highest RA on both datasets. Across these variants, $|\Delta\text{MIA}|$ remains at most 1.28 percentage points from Retrain. **These show that the proposed ranking direction, target decoupling, and CLP-aware history contribute to selecting retained clients that preserve utility while maintaining effective forgetting.**

Table 3: Ablation of PDF against its selection variants, with $\beta = 0.5$, $\lambda = 0.7$, and $k = 25$ clients selected for unlearning. ΔMIA is the signed deviation of MIA AUC from the retrained reference.

	$\alpha = 0.1$			$\alpha = 1.0$		
Method	FA	RA	ΔMIA	FA	RA	ΔMIA
FMNIST-CNN (Retrain MIA AUC: 50.32 / 50.23)						
PDF	0.72	**85.97**	−0.32	**0.28**	**89.35**	+**0.02**
non−CLP	**0.28**	85.35	−0.16	0.43	89.26	+**0.02**
Rev	1.36	82.28	+0.04	1.33	89.14	−0.06
high−Coh	1.24	82.15	−0.34	1.23	89.04	+0.03
CIFAR-10-ResNet (Retrain MIA AUC: 55.64 / 52.18)						
PDF	**0.00**	**54.48**	−0.88	**1.09**	**82.84**	−0.40
non−CLP	0.19	49.60	−1.10	1.51	81.51	+0.12
Rev	4.44	41.09	−0.77	1.30	81.91	+0.15
high−Coh	7.65	37.65	−1.28	1.20	81.74	+0.18

Table 4: Ablation study on the number of selected unlearning clients k and retained utility–forgetting trade-off β on FMNIST under $\alpha = 0.1$ with $\mathcal{I}_{\text{tar}} = \{0\}$. MIA AUC values in parentheses denote differences from Retrain (50.32). The shaded column denotes the default $k = 25$ in Table 1.

	k	10	15	20	25	30	35
$\beta = 0.1$	FA	0.00	0.00	0.00	0.00	0.28	0.28
	RA	75.65	80.51	81.26	82.60	84.45	85.18
	MIA AUC	49.96 (−0.36)	50.47 (+0.15)	50.09 (−0.23)	50.01 (−0.31)	50.02 (−0.30)	49.95 (−0.37)
	Comp	0.76	0.78	0.81	0.83	0.85	0.87
$\beta = 0.5$	FA	0.28	0.28	0.28	0.72	0.44	0.28
	RA	79.26	82.76	84.43	85.97	86.40	86.37
	MIA AUC	49.99 (−0.33)	50.13 (−0.19)	50.04 (−0.28)	50.00 (−0.32)	50.13 (−0.19)	50.05 (−0.27)
	Comp	0.79	0.82	0.85	0.88	0.89	0.91
$\beta = 0.9$	FA	0.72	1.27	0.99	0.44	0.44	0.28
	RA	84.24	85.49	85.99	86.89	86.99	86.76
	MIA AUC	50.26 (−0.06)	50.22 (−0.10)	50.16 (−0.15)	49.91 (−0.41)	50.07 (−0.25)	50.05 (−0.27)
	Comp	0.81	0.85	0.87	0.89	0.91	0.92
Comm		4.30	4.46	4.63	4.79	4.96	5.13

In Table 4, increasing k generally improves retained utility, but the gains diminish as more clients participate. At β=0.5, for example, RA rises from 79.26% at k=10 to 85.97% at k=25, but increases only marginally to 86.40% at k=30. A similar saturation occurs for β=0.9, where increasing k from 20 to 25 improves RA by 0.9 percentage points, compared with only 0.1 from 25 to 30. In contrast, communication and computation increase steadily with k, revealing a clear utility–efficiency trade-off. MIA AUC remains within 0.41 percentage points of Retrain throughout the sweep, indicating that this trade-off does not alter retraining-like privacy behavior. The observed variation across β is consistent with the forgetting-utility behavior discussed in Table 1. We therefore adopt k=25 as the default.

6 Conclusion

We presented PDF, a path-dependent framework that uses CLP-weighted training contribution histories to identify a small subset of retained clients that are particularly useful for client-level FU. Together, our theoretical and empirical results establish path-dependent influence as a principled basis for deciding which clients should participate in unlearning, thereby improving the forgetting-utility-efficiency trade-off. Notably, PDF achieves these gains without introducing a specialized unlearning mechanism: it leaves the learning stage, underlying local optimization and aggregation rules unchanged and modifies only retained-client participation. This modularity positions PDF as a foundation for more advanced FU systems, where path-dependent selection can be integrated with adaptive participation, specialized update-correction or aggregation mechanisms, and support for multiple or sequential unlearning requests.

AI USE STATEMENT

In this work, we used generative AI tools to assist in revising the presentation of mathematical claims and proofs, provide feedback on research methodology, and support the interpretation of experimental results. Generative AI was not used to develop the underlying theoretical models or conceptual framework, provide critical ingredients for the proofs, generate synthetic datasets, implement the proposed method, or conduct experiments; the remaining required disclosure tasks were not applicable. We additionally used generative AI for brainstorming, searching for information, drafting and editing portions of the manuscript to improve clarity and readability, and editing software code. All AI-assisted mathematical content, methodological suggestions, result interpretations, code, and text were independently checked and revised by the authors. The implementation and experiments were conducted and verified by the authors. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

We provide hyperparameter configurations and implementation details for our experiments in Section 5 and Appendix A.5. The code necessary to reproduce the results reported is available in the following anonymous repository: https://anonymous.4open.science/r/PDF-for-federated-unlearning. All proofs, assumptions, and hypotheses underlying our theoretical analysis, together with the corresponding empirical evaluations are presented in Appendices A.3 and A.4.

REFERENCES

Alessandro Achille, Matteo Rovere, and Stefano Soatto. Critical learning periods in deep networks. In *International Conference on Learning Representations*, 2018.

Jonathan Brophy and Daniel Lowd. Machine unlearning for random forests. In *International Conference on Machine Learning*, pp. 1092–1104, 2021.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE Symposium on Security and Privacy*, pp. 463–480, 2015.

Vikram S Chundawat, Pushkar Niroula, Prasanna Dhungana, Stefan Schoepf, Murari Mandal, and Alexandra Brintrup. CONDA: Fast federated unlearning with contribution dampening. *arXiv preprint arXiv:2410.04144*, 2024.

European Parliament and Council of the European Union. General Data Protection Regulation. [Online]. Available: "`https://gdpr-info.eu/`", 2016. Accessed: 2026-09-16.

Jack Foster, Stefan Schoepf, and Alexandra Brintrup. Fast machine unlearning without retraining through selective synaptic dampening. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 12043–12051, 2024.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9304–9312, 2020.

Qinghui Gong, Xue Yang, Xunlei Chen, Jinshan Lai, Hua Meng, and Xiaohu Tang. FedOrtho: Efficient federated unlearning via orthogonal convolution and adaptive soft pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8009–8018, 2026.

Hanlin Gu, Win K Ong, Chee S Chan, and Lixin Fan. Ferrari: federated feature unlearning via optimizing feature sensitivity. *Advances in Neural Information Processing Systems*, 37:24150–24180, 2024.

Thierry Hanser, Ernst Ahlberg, Alexander Amberg, Lennart T Anger, Chris Barber, Richard J Brennan, Alessandro Brigo, Annie Delaunois, Susanne Glowienke, Nigel Greene, et al. Data-driven federated learning in drug discovery with knowledge distillation. *Nature Machine Intelligence*, 7(3):423–436, 2025.

Elizabeth Liz Harding, Jarno J Vanto, Reece Clark, L Hannah Ji, and Sara C Ainsworth. Understanding the scope and impact of the california consumer privacy act of 2018. *Journal of Data Protection & Privacy*, 2(3):234–253, 2019.

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*, pp. 770–778, 2016.

Chengxiang Huang and Bingyan Liu. PA3Fed: Period-aware adaptive aggregation for improved federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 17395–17403, 2025.

Hyejun Jeong, Shiqing Ma, and Amir Houmansadr. Survey on federated unlearning: Challenges and opportunities. *IEEE Transactions on Big Data*, 12(3):702–720, 2026.

Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. Model sparsity can simplify machine unlearning. *Advances in Neural Information Processing Systems*, 36:51584–51605, 2023.

Yasser H Khalil, Leo Brunswic, Soufiane Lamghari, Xu Li, Mahdi Beitollahi, and Xi Chen. NoT: Federated unlearning via weight negation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 25759–25769, 2025.

Michael Kleinman, Alessandro Achille, and Stefano Soatto. Critical learning periods emerge even in deep linear networks. In *International Conference on Learning Representations*, 2024.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The CIFAR-10 and CIFAR-100 datasets. `https://cave.cs.toronto.edu/kriz/cifar.html`, 2009. Accessed: 2026-09-16.

Zexi Li, Xinyi Shang, Rui He, Tao Lin, and Chao Wu. No fear of classifier biases: Neural collapse inspired federated learning with synthetic and fixed classifier. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5319–5329, 2023.

Gaoyang Liu, Xiaoqiang Ma, Yang Yang, Chen Wang, and Jiangchuan Liu. FedEraser: Enabling efficient client-level data removal from federated learning models. In *2021 IEEE/ACM 29th international symposium on quality of service (IWQOS)*, pp. 1–10, 2021.

Mi Luo, Fei Chen, Dapeng Hu, Yifan Zhang, Jian Liang, and Jiashi Feng. No Fear of Heterogeneity: Classifier calibration for federated learning with non-iid data. *Advances in Neural Information Processing Systems*, 34:5972–5984, 2021.

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

Minh-Duong Nguyen, Senura Wanasekara, Le-Tuan Nguyen, Quoc-Viet Pham, Ken-Tye Yong, Nguyen H Tran, and Dung D Le. Computation and communication efficient federated unlearning via on-server gradient conflict mitigation and expression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3347–3357, 2026a.

Thanh Linh Nguyen, Marcela Tuler De Oliveira, An Braeken, Aaron Yi Ding, and Quoc-Viet Pham. Toward Verifiable Federated Unlearning: Framework, Challenges, and the Road Ahead . *IEEE Internet Computing*, 30(01):59–70, 2026b.

Jaehoon Oh, SangMook Kim, and Se-Young Yun. FedBABU: Toward enhanced representation for federated image classification. In *International Conference on Learning Representations*, 2022.

Zibin Pan, Zhichao Wang, Chi Li, Kaiyan Zheng, Boqi Wang, Xiaoying Tang, and Junhua Zhao. Federated unlearning with gradient descent and conflict mitigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 19804–19812, 2025.

Nicolò Romandini, Alessio Mora, Carlo Mazzocca, Rebecca Montanari, and Paolo Bellavista. Federated unlearning: A survey on methods, design guidelines, and evaluation metrics. *IEEE Transactions on Neural Networks and Learning Systems*, 36(7):11697–11717, 2024.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy*, pp. 3–18, 2017.

Stelios Theophanous, Per-Ivar Lønne, Ananya Choudhury, Maaike Berbee, Charlotte Deijen, Andre Dekker, Matthew Field, Maria Antonietta Gambacorta, Alexandra Gilbert, Marianne Grønlie Guren, et al. An international multi-centre study to develop and validate federated learning-based prognostic models for anal cancer. *Nature Communications*, 17(1):3956, 2026.

Guanjie Wang and Chen Chen. GR-Gauge: Cost-efficient training configuration by gauging the gradient redundancy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12934–12943, 2026.

Peng Wang, Shoupeng Lu, Hao Yin, Banglie Yang, Tianli Zhu, and Cheng Dai. FedCM: Client clustering and migration in federated learning via gradient path similarity and update direction deviation. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 6343–6351, 2025a.

Weiqi Wang, Chenhan Zhang, Zhiyi Tian, and Shui Yu. FedU: Federated unlearning via user-side influence approximation forgetting. *IEEE Transactions on Dependable and Secure Computing*, 22 (3):2550–2562, 2025b.

Xiuting Weng, Ruizhi Pu, Yuanhang Yao, Kun Yue, Zhiwen Tang, and Lixing Yu. GDFA: Geometry-driven federated unlearning with directional task vector alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10346–10356, 2026.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *arXiv:1708.07747*, 2017.

Zuobin Xiong, Wei Li, Yingshu Li, and Zhipeng Cai. Exact-Fun: An exact and efficient federated unlearning approach. In *2023 IEEE International Conference on Data Mining (ICDM)*, pp. 1439–1444, 2023.

Gang Yan, Hao Wang, and Jian Li. Seizing critical learning periods in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8788–8796, 2022.

Gang Yan, Hao Wang, Xu Yuan, and Jian Li. CriticalFL: A critical learning periods augmented client selection framework for efficient federated learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2898–2907, 2023.

Liping Yi, Han Yu, Zhuan Shi, Gang Wang, Xiaoguang Liu, Lizhen Cui, and Xiaoxiao Li. FedSSA: semantic similarity-based aggregation for efficient model-heterogeneous personalized federated learning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 5371–5379, 2024.

Lefeng Zhang, Tianqing Zhu, Haibin Zhang, Ping Xiong, and Wanlei Zhou. FedRecovery: Differentially private machine unlearning for federated learning frameworks. *IEEE Transactions on Information Forensics and Security*, 18:4732–4746, 2023.

Zhengyi Zhong, Weidong Bao, Ji Wang, Shuai Zhang, Jingxuan Zhou, Lingjuan Lyu, and Wei Yang Bryan Lim. Unlearning through knowledge overwriting: Reversible federated unlearning via selective sparse adapter. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 30661–30670, 2025.

A Appendix

A.1 Details Of Related Work

A.1.1 Machine Unlearning & Federated Unlearning

Machine unlearning (MU) has received growing attention as a mechanism for enforcing data-removal rights, such as the Right to be Forgotten under Article 17 of the GDPR (European Parliament & Council of the European Union, 2016). Given a trained model and a removal request, MU aims

to eliminate the influence of target data without repeatedly retraining the model from scratch (Cao & Yang, 2015). Existing MU methods can be broadly categorized into data-level and model-level approaches. Data-level methods modify or reorganize the training data, for example by perturbing or replacing target samples, or by partitioning the dataset so that only the affected subsets need to be retrained. In contrast, model-level methods directly operate on the trained model, including pruning model components associated with target data (Jia et al., 2023), transforming model parameters to counteract the contribution of the data to be forgotten (Golatkar et al., 2020), or unlearning on cached intermediate sub-models from the learning stage (Brophy & Lowd, 2021).

However, centralized MU methods typically assume access to the training data, intermediate training states, or sufficient retraining resources. These assumptions are impractical in FL, where data are distributed across clients, raw data are inaccessible to the server, and participants may be intermittently available (McMahan et al., 2017). Motivated by data ownership, privacy preservation, and regulatory compliance in distributed learning systems (McMahan et al., 2017; European Parliament & Council of the European Union, 2016; Harding et al., 2019), FU has therefore emerged as an important extension of MU, aiming to remove the influence of target clients from a collaboratively trained global model. Existing FU studies can be broadly categorized into client-level, sample-level, and class-level unlearning (Romandini et al., 2024; Jeong et al., 2026). Among them, client-level FU is particularly relevant in cross-silo settings, where an entire client may request data removal or may need to be excluded due to malicious or low-quality contributions.

Existing client-level FU methods mainly follow two technical directions: historical-update replay and direct model manipulation (Jeong et al., 2026). Historical-update replay methods approximate retraining by reusing cached training states from the learning phase. For example, FedEraser (Liu et al., 2021) stores interval historical updates from retained clients and reconstructs an unlearned model through calibrated aggregation, thereby reducing the cost of retraining from scratch. Similarly, FedRecovery (Zhang et al., 2023) performs server-side unlearning by removing the accumulated residual influence of target clients from the trained model and injecting controlled noise so that the resulting model is statistically close to the ideal retrained model. Another line of work directly manipulates model parameters, gradients, or representations to suppress the influence of target clients (Nguyen et al., 2026a; Chundawat et al., 2024; Gong et al., 2026; Pan et al., 2025). For instance, CONDA (Chundawat et al., 2024) dampens parameters that are disproportionately affected by the forget clients, while NoT (Khalil et al., 2025) applies server-side weight negation to selected layers of the trained global model, followed by a few rounds of FL fine-tuning on retained clients to recover utility. FedOrtho (Gong et al., 2026) enforces convolutional-kernel orthogonality during FL training and later uses client-computed activation differences to guide server-side adaptive soft pruning of forget-related kernels. More recently, gradient-direction correction methods have also been explored to mitigate conflicts between forgetting and utility preservation during FU (Pan et al., 2025).

A.1.2 Critical Learning Periods

As discussed above, existing unlearning methods often rely on information from the learning stage, including critical deep layers (Khalil et al., 2025; Zhong et al., 2025; Gong et al., 2026), and cached checkpoints or historical updates (Liu et al., 2021; Chundawat et al., 2024; Zhang et al., 2023). However, these methods largely overlook the temporal non-uniformity of training rounds, implicitly treating client contributions as equally important to subsequent unlearning. This assumption neglects the existence of `CLPs`, during which client updates may disproportionately influence the trained model, other clients' updates, and, consequently, the difficulty of removing that influence.

`CLPs` were initially characterized as early training stages of deep neural network optimization, in which model representations and feature coupling are formed strongly, after which their structure becomes increasingly difficult to alter (Achille et al., 2018; Kleinman et al., 2024). Subsequent studies have demonstrated analogous behavior in FL, showing that critical training rounds can have a disproportionate and persistent effect on final global-model performance (Yan et al., 2022; 2023). Building on this temporal non-uniformity, existing `CLP`-aware FL methods exploit temporally non-uniform training dynamics to improve learning performance. For example, to improve accuracy and convergence, CriticalFL (Yan et al., 2023) allocates more clients during `CLPs`, whereas PA3Fed (Huang & Liu, 2025) monitors period-dependent learning dynamics throughout training and adaptively weights model aggregation. Wang et al. propose GR-Gauge (Wang & Chen, 2026), a

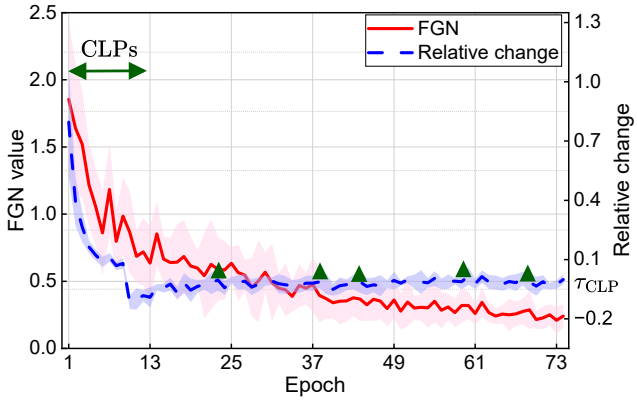


Figure 2: Illustration of CLP detection using the FGN proxy on FMNIST (Xiao et al., 2017), where the green double-headed arrows and triangles indicate the detected CLPs.

cost-efficient hyperparameter optimization method that leverages critical learning periods to diagnose whether early training is proceeding under a healthy configuration. Specifically, they measure temporal and spatial gradient redundancy during the critical period and use these signals to guide learning-rate/batch-size selection, early termination, and state reuse. While their work focuses on training configuration rather than federated unlearning, it reinforces the view that early training dynamics provide disproportionately informative signals for later model behavior.

A.2 Critical Learning Period Detection

The onset and duration of CLPs vary across training trajectories and are not confined to a fixed early interval (see Fig. 2) (Huang & Liu, 2025). Let $\mathcal{T}_{\text{CLP}} = \bigcup_{q=1}^{Q} \mathcal{T}_{\text{CLP}}^{(q)} \subseteq \{1, \ldots, T_{\text{learn}}\}$ denote the set of detected critical rounds, where Q is the number of identified critical periods. To detect CLPs online, the server employs the federated gradient norm (FGN) proxy (Yan et al., 2023). Given the round-t local gradient $\boldsymbol{g}_i^{(t)}$, one local training step produces the first-order loss change: $\Delta F_i^{(t)} = F_i\big(\boldsymbol{w}^{(t-1)} - \eta_{\ell,t}\,\boldsymbol{g}_i^{(t)}\big) - F_i\big(\boldsymbol{w}^{(t-1)}\big) \approx -\,\eta_{\ell,t}\big\|\boldsymbol{g}_i^{(t)}\big\|_2^2$. Accordingly, the round-level loss-decrease proxy is defined as $\text{FGN}^{(t)} \triangleq \eta_{\ell,t}\sum_{i\in\mathcal{S}^{(t)}} p_i^{(t)}\big\|\boldsymbol{g}_i^{(t)}\big\|_2^2$. A round t is flagged as critical (i.e. $t \in \mathcal{T}_{\text{CLP}}$), when $\frac{\text{FGN}^{(t)} - \text{FGN}^{(t-1)}}{\text{FGN}^{(t-1)}} \geq \tau_{\text{CLP}}$, where $\tau_{\text{CLP}} > 0$ is an empirically selected sensitivity threshold (Yan et al., 2023; Huang & Liu, 2025). Evaluating this criterion throughout training enables the detection of both early and late critical periods.

A.3 Supporting Theoretical Analysis

This section provides the assumptions and proofs supporting Section 4.3. We first characterize how client-specific perturbations propagate during the learning stage, then analyze a selected-client unlearning update at an unlearning iterate, and finally establish a conditional bound on PD-based selection at unlearning initialization. The empirical relationship between FGN-detected CLPs and counterfactual influence is assessed separately in Appendix A.4.

A.3.1 Assumptions and Counterfactual Construction

Throughout the proofs, we consider the FedAvg protocol in Eq. 2. For a target client o, the factual trajectory $\boldsymbol{w}^{(t)}$ and the counterfactual trajectory $\boldsymbol{w}_{-o}^{(t)}$ share the same initialization and coupled training randomness. The counterfactual trajectory excludes o whenever it participates and preserves the sampled clients other than o. Thus, the same local-update maps are evaluated along the two trajectories.

Assumption 2 (Smoothness) *Each local objective F_i is L-smooth: $\|\nabla F_i(\boldsymbol{u}) - \nabla F_i(\boldsymbol{u}')\|_2 \leq L\|\boldsymbol{u} - \boldsymbol{u}'\|_2, \forall \boldsymbol{u}, \boldsymbol{u}' \in \mathbb{R}^d$. The retained and target objectives F_{ret} and F_{tar} are L_{ret}-smooth and L_{tar}-smooth, respectively.*

When $o \in \mathcal{S}^{(t)}$, the remaining aggregation weights are renormalized as follows:

$$p_{i,-o}^{(t)} = \frac{p_i^{(t)}}{\sum_{j \in \mathcal{S}^{(t)} \setminus \{o\}} p_j^{(t)}} = \frac{p_i^{(t)}}{1 - p_o^{(t)}},\ i \in \mathcal{S}^{(t)} \setminus \{o\}. \tag{14}$$

For readability, we define the factual and counterfactual server-update maps, respectively, by:

$$\Phi_t(\boldsymbol{x}) = (1 - \gamma_t)\boldsymbol{x} + \gamma_t \sum_{i \in \mathcal{S}^{(t)}} p_i^{(t)} \boldsymbol{h}_i^{(t)}(\boldsymbol{x}), \tag{15}$$

$$\Phi_{t,-o}(\boldsymbol{x}) = (1 - \gamma_t)\boldsymbol{x} + \gamma_t \sum_{i \in \mathcal{S}^{(t)} \setminus \{o\}} p_{i,-o}^{(t)} \boldsymbol{h}_i^{(t)}(\boldsymbol{x}). \tag{16}$$

In case of $o \notin \mathcal{S}^{(t)}$, we set $\Phi_{t,-o} = \Phi_t$. Therefore, at learning round t, we have $\boldsymbol{w}^{(t)} = \Phi_t(\boldsymbol{w}^{(t-1)})$ and $\boldsymbol{w}_{-o}^{(t)} = \Phi_{t,-o}(\boldsymbol{w}_{-o}^{(t-1)})$.

We then define the counterfactual deviation to the global model caused by client o at round t as

$$\boldsymbol{\Delta}_o^{(t)} = \boldsymbol{w}^{(t)} - \boldsymbol{w}_{-o}^{(t)} = \Phi_t(\boldsymbol{w}^{(t-1)}) - \Phi_{t,-o}(\boldsymbol{w}_{-o}^{(t-1)}). \tag{17}$$

The direct perturbation introduced by client o at round t is presented by:

$$\boldsymbol{b}_o^{(t)} = \Phi_t(\boldsymbol{w}_{-o}^{(t-1)}) - \Phi_{t,-o}(\boldsymbol{w}_{-o}^{(t-1)}), \tag{18}$$

with $\boldsymbol{b}_o^{(t)} = \mathbf{0}$ when $o \notin \mathcal{S}^{(t)}$. Thus, $\boldsymbol{\Delta}_o^{(t)}$ measures the accumulated difference between the factual and counterfactual trajectories, whereas $\boldsymbol{b}_o^{(t)}$ isolates the contribution of client o at round t.

A.3.2 Proof of Proposition 1 - Path-dependent client influence

Proof 1 *Under Assumption 2, one local gradient step is $(1+\eta_{\ell,t}L)$-Lipschitz. Since the local-update map $\boldsymbol{h}_i^{(t)}$ consists of E local steps, we have*

$$\left\|\boldsymbol{h}_i^{(t)}\left(\boldsymbol{w}^{(t-1)}\right) - \boldsymbol{h}_i^{(t)}\left(\boldsymbol{w}_{-o}^{(t-1)}\right)\right\|_2 \leq \rho_t \left\|\boldsymbol{w}^{(\boldsymbol{t-1})} - \boldsymbol{w}_{\boldsymbol{-o}}^{(\boldsymbol{t-1})}\right\|_2, \quad \textit{where } \rho_t = (1 + \eta_{\ell,t}L)^E. \tag{19}$$

Based on Eq. 17 and Eq. 18, we have

$$\begin{aligned}
\boldsymbol{\Delta}_o^{(t)} &= \Phi_t(\boldsymbol{w}^{(t-1)}) - \Phi_t(\boldsymbol{w}_{-o}^{(t-1)}) + \boldsymbol{b}_o^{(t)} \\
&= (1 - \gamma_t)\left(\boldsymbol{w}^{(t-1)} - \boldsymbol{w}_{-o}^{(t-1)}\right) + \gamma_t \sum_{i \in \mathcal{S}^{(t)}} p_i^{(t)} \left[\boldsymbol{h}_i^{(t)}\left(\boldsymbol{w}^{(t-1)}\right) - \boldsymbol{h}_i^{(t)}\left(\boldsymbol{w}_{-o}^{(t-1)}\right)\right] + \boldsymbol{b}_o^{(t)} \\
&= (1 - \gamma_t)\boldsymbol{\Delta}_o^{(t-1)} + \gamma_t \sum_{i \in \mathcal{S}^{(t)}} p_i^{(t)} \left[\boldsymbol{h}_i^{(t)}\left(\boldsymbol{w}^{(t-1)}\right) - \boldsymbol{h}_i^{(t)}\left(\boldsymbol{w}_{-o}^{(t-1)}\right)\right] + \boldsymbol{b}_o^{(t)}.
\end{aligned} \tag{20}$$

Applying the triangle inequality to Eq. 20, and using Eq. 19 and Eq. 17, we obtain the following inequality:

$$\begin{aligned}
\left\|\boldsymbol{\Delta}_o^{(t)}\right\|_2 &\leq (1 - \gamma_t)\left\|\boldsymbol{\Delta}_o^{(t-1)}\right\|_2 + \gamma_t\rho_t\left\|\boldsymbol{\Delta}_o^{(t-1)}\right\|_2 + \left\|\boldsymbol{b}_o^{(t)}\right\|_2 \\
&= \Gamma_t\left\|\boldsymbol{\Delta}_o^{(t-1)}\right\|_2 + \left\|\boldsymbol{b}_o^{(t)}\right\|_2,
\end{aligned} \tag{21}$$

where

$$\Gamma_t = 1 - \gamma_t + \gamma_t\rho_t. \tag{22}$$

Moreover, the factual and counterfactual trajectories share the same initialization, as a result, $\boldsymbol{\Delta}_o^{(0)} = \mathbf{0}$*. We recursively apply the one-round inequality in Eq. 21, therefore, we have*

$$\left\|\boldsymbol{\Delta}_o^{(T_{\text{learn}})}\right\|_2 \leq \sum_{t=1}^{T_{\text{learn}}} \left(\prod_{s=t+1}^{T_{\text{learn}}} \Gamma_s\right) \left\|\boldsymbol{b}_o^{(t)}\right\|_2 . \tag{23}$$

Then, we define $\omega_t = \prod_{s=t+1}^{T_{\text{learn}}} \Gamma_s$*, thereby obtaining Eq. 7. This completes the proof.*

A.3.3 Proof of Proposition 2 - One-step retained and target objective bounds

Proof 2 *Using the notation introduced in Section 4.3, the selected-client unlearning update satisfies* $\boldsymbol{w}^+ - \boldsymbol{w} = -\eta_u \boldsymbol{v}_{\mathcal{K}}$*. For any* L_F*-smooth objective function* F*, we have*

$$\left|F(\boldsymbol{w}^+) - F(\boldsymbol{w}) - \left\langle \nabla F(\boldsymbol{w}), \boldsymbol{w}^+ - \boldsymbol{w} \right\rangle\right| \leq \frac{L_F}{2} \left\|\boldsymbol{w}^+ - \boldsymbol{w}\right\|_2^2 . \tag{24}$$

Applying the upper bound to F_{ret} *and the lower bound to* F_{tar}*, and using the definitions of* $A_{\mathcal{K}}(\boldsymbol{w})$ *and* $C_{\mathcal{K}}(\boldsymbol{w})$*, we have the following inequalities, respectively,*

$$\begin{aligned} F_{\text{ret}}(\boldsymbol{w}^+) - F_{\text{ret}}(\boldsymbol{w}) &\leq -\eta_u \left\langle \nabla F_{\text{ret}}(\boldsymbol{w}), \boldsymbol{v}_{\mathcal{K}} \right\rangle + \frac{L_{\text{ret}} \eta_u^2}{2} \left\|\boldsymbol{v}_{\mathcal{K}}\right\|_2^2 \qquad (25) \\ &= -\eta_u A_{\mathcal{K}}(\boldsymbol{w}) + \frac{L_{\text{ret}} \eta_u^2}{2} \left\|\boldsymbol{v}_{\mathcal{K}}\right\|_2^2 , \\ F_{\text{tar}}(\boldsymbol{w}^+) - F_{\text{tar}}(\boldsymbol{w}) &\geq -\eta_u \left\langle \nabla F_{\text{tar}}(\boldsymbol{w}), \boldsymbol{v}_{\mathcal{K}} \right\rangle - \frac{L_{\text{tar}} \eta_u^2}{2} \left\|\boldsymbol{v}_{\mathcal{K}}\right\|_2^2 \qquad (26) \\ &= -\eta_u C_{\mathcal{K}}(\boldsymbol{w}) - \frac{L_{\text{tar}} \eta_u^2}{2} \left\|\boldsymbol{v}_{\mathcal{K}}\right\|_2^2 . \end{aligned}$$

The sufficient conditions in Eq. 10 follow directly by requiring the corresponding upper bound to be negative and the corresponding lower bound to be positive, respectively. This completes the proof.

A.3.4 Proof of Corollary 1 - Conditional Near-Optimality of PD-Based Selection

Let $Q_i^0 = Q_i(\boldsymbol{w}^{(T_{\text{learn}})})$, $\boldsymbol{w}_0 = \boldsymbol{w}^{(T_{\text{learn}})}$, and $pd_i = pd_i(\mathcal{I}_{\text{tar}})$. Under Assumption 1, every retained client satisfies

$$\alpha_Q \, pd_i + b_Q - \epsilon \leq Q_i^0 \leq \alpha_Q \, pd_i + b_Q + \epsilon. \tag{27}$$

Proof 3 *Let* $\mathcal{K}$ *be the PD-selected size-*k *subset and* $\mathcal{K}^\star$ *the initialization-utility-optimal size-*k *subset (defined in Corollary 1).*

Because $\alpha_Q > 0$ *and* $\mathcal{K}$ *contains the* k *largest PD scores,*

$$\sum_{i \in \mathcal{K}} pd_i(\mathcal{I}_{\text{tar}}) \geq \sum_{i \in \mathcal{K}^\star} pd_i(\mathcal{I}_{\text{tar}}). \tag{28}$$

Based on Eq. 27 and Eq. 28, we have the following inequality:

$$\begin{aligned} \sum_{i \in \mathcal{K}} Q_i^0 &\geq \alpha_Q \sum_{i \in \mathcal{K}} pd_i + k b_Q - k\epsilon \\ &\geq \alpha_Q \sum_{i \in \mathcal{K}^\star} pd_i(\mathcal{I}_{\text{tar}}) + k b_Q - k\epsilon \\ &\geq \sum_{i \in \mathcal{K}^\star} Q_i^0 - 2k\epsilon. \end{aligned} \tag{29}$$

Consequently, we have $0 \leq \sum_{i \in \mathcal{K}^\star} Q_i^0 - \sum_{i \in \mathcal{K}} Q_i^0 \leq 2k\epsilon$*. This proves Eq. 13.*

Furthermore, under uniform selected-client aggregation, we have

$$A_{\mathcal{K}}(\boldsymbol{w}_0) - \mu C_{\mathcal{K}}(\boldsymbol{w}_0) = \frac{1}{k} \sum_{i \in \mathcal{K}} Q_i^0. \tag{30}$$

As a result, we have

$$[A_{\mathcal{K}^\star}(\boldsymbol{w}_0) - \mu C_{\mathcal{K}^\star}(\boldsymbol{w}_0)] - [A_{\mathcal{K}}(\boldsymbol{w}_0) - \mu C_{\mathcal{K}}(\boldsymbol{w}_0)] \leq 2\epsilon. \tag{31}$$

This completes the proof.

A.4 Empirical assessment of the theoretical analysis

In this section, we assess the influence of FGN-detected CLPs as discussed in Proposition 1, the selected-client alignments in Proposition 2, and the unlearning initialization-utility approximation and selection quality underlying Assumption 1 and Corollary 1.

A.4.1 Influence of FGN-detected CLPs

Hypothesis 1 (CLP influence informativeness) *At training round t, we define the average propagated influence-bound client contributions as follows:*

$$\zeta^{(t)} = \frac{1}{|\mathcal{S}^{(t)}|} \sum_{i \in \mathcal{S}^{(t)}} \omega_t \left\| \boldsymbol{b}_i^{(t)} \right\|_2. \tag{32}$$

For $0 < |\mathcal{T}_{\mathrm{CLP}}| < T_{\mathrm{learn}}$, we hypothesize that FGN-detected CLP rounds satisfy $\mathcal{T}_{\mathrm{CLP}}$ satisfies

$$\frac{1}{|\mathcal{T}_{\mathrm{CLP}}|} \sum_{t \in \mathcal{T}_{\mathrm{CLP}}} \zeta^{(t)} \geq \frac{1}{T_{\mathrm{learn}} - |\mathcal{T}_{\mathrm{CLP}}|} \sum_{t \notin \mathcal{T}_{\mathrm{CLP}}} \zeta^{(t)} + \nu, \tag{33}$$

where $\nu \geq 0$ is an average influence-separation margin.

Hypothesis 1 does not require the detected CLPs to occur within an early training window. The detected set $\mathcal{T}_{\mathrm{CLP}} = \bigcup_{q=1}^{Q} \mathcal{T}_{\mathrm{CLP}}^{(q)}$ may include multiple early or late critical periods. The FGN proxy does not estimate $\zeta^{(t)}$ directly and serves as an online indicator of potentially influential training intervals.

Counterfactual evaluation protocol. For each of six random seeds, we run factual federated training and record the round-level checkpoints, participating clients, aggregation weights, and random seeds. For each sampled client–round pair (i, t), we initialize a separate counterfactual from the factual checkpoint immediately before round t, omit client i only at that round, and renormalize the remaining participating clients' data-size FedAvg weights. Subsequent rounds are rerun using the original participation sequence and random seeds, with local updates recomputed along the counterfactual trajectory. Client i may participate again in later rounds. Each counterfactual therefore contains exactly one omission, without accumulating removals across experiments. This design isolates the effect of one specific client contribution at one specific round. Therefore, the factual final model is always compared with a counterfactual containing only that omission. Here, we quantify the contribution's final-model parameter influence after the training finishes by

$$P_{i,t} = \left\| \boldsymbol{w}^{(T_{\mathrm{learn}})} - \boldsymbol{w}_{-(i,t)}^{(T_{\mathrm{learn}})} \right\|_2, \tag{34}$$

where the norm is computed over the flattened trainable parameters. Within each seed, we average the sampled client–round influences separately for the CLP and non-CLP groups, giving equal weight to each sampled pair. Their difference defines the seed-wise empirical margin.

Empirical validation of CLP influence and Hypothesis 1. As shown in Table 5, the mean final-model deviation is 1.1356 ± 0.2087 for contributions in CLPs compared to that of 0.7176 ± 0.1219 for contributions in non-CLP rounds. The corresponding paired mean difference is $\hat{\nu} = 0.4179$, with a 95% confidence interval of $[0.2915,\ 0.5444]$, ***supporting greater average final-model parameter influence of sampled CLP contributions in the evaluated setting*** as presented in Hypothesis 1.

Table 5: Empirical assessment of CLP influence and Hypothesis 1 on FMNIST with $\alpha = 0.1$. Results are reported as mean $\pm$ standard deviation over six random seeds. The 95% confidence interval is a paired Student's t-interval for the seed-wise CLP−non−CLP influence margin.

CLP	non−CLP	Margin $\hat{\nu}$	95% CI	Spearman ρ	Kendall τ
1.1356 ± 0.2087	0.7176 ± 0.1219	0.4179 ± 0.1205	$[0.2915, 0.5444]$	0.2660 ± 0.1292	0.1815 ± 0.0939

Rank correlations measure the association between the round-level perturbation and the resulting final-model parameter deviation.

Moreover, the relatively weak association between the perturbation magnitude at round t and final-model deviation (i.e., Spearman $\rho = 0.2660$ and Kendall $\tau = 0.1815$) indicates that contribution magnitude alone does not reliably rank final influence. These findings provide empirical support for CLP-aware temporal weighting and are consistent with the path-dependent propagation motivating Proposition 1, which emphasizes the important role of CLPs.

A.4.2 Selected-client gradient alignment

For the PD-selected top-25 subset, the retained-gradient alignment is $A_{\mathcal{K}} = 0.0169$, while the target-gradient alignment is $C_{\mathcal{K}} = -0.1679$. Under the update $\boldsymbol{w}^{+} = \boldsymbol{w} - \eta_u \boldsymbol{v}_{\mathcal{K}}$, these signs favor a retained-loss decrease and a target-loss increase, as characterized by Proposition 2. Over the complete unlearning trajectory, the retained loss decreases by 0.0840, while the target loss increases by 4.0127. These cumulative changes provide complementary evidence that the selected $\mathcal{K}$ supports retained-utility preservation and target forgetting.

A.4.3 PD score–first-order client utility consistency and client selection quality

Table 6: Empirical validation of PD–utility ranking agreement and affine approximation error on FMNIST under $\alpha = 0.1, \mathcal{I}_{\text{tar}} = \{0\}$ and $\beta = 0.5$. Results are reported as mean $\pm$ standard deviation over six independent-seed runs.

Spearman ρ_{S}	95% CI	Kendall τ	PairAcc
0.5934 ± 0.1094	$[0.4785, 0.7082]$	0.4352 ± 0.0975	0.7176 ± 0.0488

The normalized maximum affine approximation error is $\epsilon_{\text{aff}} / \operatorname{range}(Q) = 0.4309 \pm 0.0321$.

PD score–utility consistency. At $\boldsymbol{w}_0 = \boldsymbol{w}^{(T_{\text{learn}})}$, we compare the historical score pd_i with the initialization utility Q_i^0, defined in Eq. 11. Table 6 shows a Spearman rank correlation of 0.5934, with a 95% confidence interval of $[0.4785, 0.7082]$, a Kendall rank correlation of 0.4352, and pairwise ranking accuracy of 0.7176. Thus, the PD score correctly orders approximately 71.76% of all retained-client pairs according to Q_i^0. Furthermore, we also conducted an experiment of calculating the classification-layer and full-model utilities. Results show a Spearman correlation of 0.7022, while the initialization utility Q_i and the realized unlearning utility have a correlation of 0.7460, indicating agreement between their client rankings. *As a result, PD score provides a meaningful, although imperfect, rank-consistent proxy for unlearning-stage utility.*

Affine approximation error. For each seed, we fit a positive-slope affine mapping from the PD score to Q_i and compute ϵ_{aff} as the maximum absolute residual across retained clients. Besides, we normalize this residual by the observed utility range $\operatorname{range}(Q) = \max_i Q_i(\boldsymbol{w}_0) - \min_i Q_i(\boldsymbol{w}_0)$. The resulting normalized error is $\epsilon_{\text{aff}} / \operatorname{range}(Q) = 0.4309 \pm 0.0321$. This quantity measures the worst-case discrepancy of the fitted affine approximation and supplies an empirical value ϵ for Assumption 1.

PD-based top-k selection quality. For $k = 25$, the mean overlap $\frac{|\mathcal{K} \cap \mathcal{K}^{\star}|}{k}$ is 0.7, with a 95% confidence interval of $[0.6421, 0.7579]$, corresponding to approximately 18 shared clients out of 25 on average. The aggregate initialization utility of the PD-selected subset is 1.1041, compared with 1.4259 for the utility-optimal subset, resulting in a gap of 0.3218. These results show that PD recovers

a substantial portion of the utility-optimal selection using only the learning-stage histories and satisfy Eq. 13. Moreover, the observed gap is only 0.0899 of the affine worst-case bound, indicating that the theoretical guarantee is valid but conservative. Corollary 1 concerns client selection quality, while subsequent unlearning performance is evaluated in Section 5.

A.5 Implementation Details

A.5.1 Experiment evaluation protocol

Data partition. Each client's training partition of the original dataset is divided into four disjoint splits: approximately 64% client training, 16% client evaluation, 16% proxy training, and 4% proxy evaluation, using class-stratified sampling. In label-flipping settings, label $y \in \{0, ..., C-1\}$ belonging to a designated forgetting client are replaced by $(y+1) \mod C$, where C is the number of classes. This transformation is applied to all four splits of that client.

Membership inference attack. For each FU method, we train five auxiliary shadow models using the same learning and unlearning procedure. For each shadow model, output probabilities on its training samples are labeled as *members*, whereas outputs on disjoint evaluation samples are labeled as *non-members*. These probability vectors are used to train a three-layer MLP attack classifier. We then query the actual unlearned model produced by the corresponding method using original FL training samples and held-out client-evaluation samples and pass the resulting probability vectors to the attack classifier. Thus, the attack evaluates membership leakage through the unlearned model's predictions without accessing its parameters. Member and non-member samples are balanced, and performance is reported using MIA AUC.

A.5.2 CNN Architecture for Fashion-MNIST

Fashion-MNIST images are resized from 28×28 to 32×32 before being passed to the CNN, which consists of a feature extractor and a classifier. The feature extractor comprises three convolutional blocks. The first block applies a `Conv2d` layer with 1 input channel and 32 output channels (kernel size 3), followed by Batch Normalization, ReLU activation, and a `Dropout2d` layer (rate 0.3). The second block applies a `Conv2d` layer with 32 input channels and 64 output channels (kernel size 3, stride 1), followed by Batch Normalization, ReLU activation, and a 2×2 max-pooling layer. The third block applies a `Conv2d` layer with 64 input and output channels (kernel size 3), followed by Batch Normalization, ReLU activation, a 4×4 max-pooling layer, and a `Dropout2d` layer (rate 0.2). The resulting feature map of size $64 \times 3 \times 3$ is flattened into a 576-dimensional vector. The classifier consists of a fully connected layer ($576 \to 256$) with ReLU activation and Dropout (rate 0.1), followed by a final linear layer ($256 \to 10$). All convolutional and linear layers are initialized with Kaiming uniform initialization with zero biases.

A.5.3 Federated Unlearning Baselines

Below are details of state-of-the-art FU approaches which are used to evaluate the efficacy, effectiveness, and efficiency of `PDF`.

- **Retrain**. As the gold-standard unlearning baseline, Retrain removes the target clients entirely and trains from scratch on the remaining clients. While this guarantees exact unlearning, it is prohibitively expensive in practice due to substantial computational overhead and reliance on the participation availability of all remaining clients.
- **FedEraser** (Liu et al., 2021). FedEraser approximates the retrained model without incurring its full cost by replaying periodic historical parameter updates cached from remaining clients through a calibrated aggregation procedure, achieving significant reductions in both time and storage overhead relative to full retraining.
- **Exact-Fun** (E-F) (Xiong et al., 2023). Exact-Fun (E-F) leverages quantized FL to stabilize the global model, enabling exact unlearning with high efficiency. Upon a deletion request, only the affected client retrains locally. Retraining of the global model is triggered solely when the quantized model state changes, allowing many requests to be resolved with minimal overhead.

- **CONDA** (Chundawat et al., 2024; Foster et al., 2024). CONDA is a server-side unlearning method that eliminates a target client's influence without client participation. It tracks per-client parameter contributions throughout training, estimates the forgotten client's relative influence across model parameters, and applies a targeted dampening factor to parameters most strongly associated with the forgotten client.
- **FedRecovery** (Zhang et al., 2023). FedRecovery is a server-side unlearning method that does not require retraining-based calibration and fine-tuning. It removes the target client's accumulated gradient residuals from the trained global model cached from the learning phase. It also injects calibrated Gaussian noise to mask the remaining distance between the recovered model and the ideal retrained model, thereby making the two models statistically indistinguishable.
- **FUSED** (Zhong et al., 2025). FUSED confines unlearning to the most task-critical layers identified during training, attaching lightweight sparse adapters exclusively to these layers and fine-tuning them on remaining clients to overwrite the forgotten knowledge. This design substantially reduces unlearning cost while preserving reversibility.
- **FOUL** (Nguyen et al., 2026a). FOUL is an on-server unlearning method that first disentangles the model into causal and non-causal components during the learning-to-unlearn stage, and then performs unlearning on the domain-specific non-causal component. In the unlearning stage, FOUL constructs an aggregated unlearning gradient by optimizing learnable coefficients over client gradients, encouraging alignment with retain-client gradients while reducing alignment with forget-client gradients. This design preserves retain-client knowledge while removing forget-client-specific influence without requiring the server to access client data.
- **GDFA** (Weng et al., 2026). GDFA is a model-manipulation-based federated unlearning method that exploits loss-landscape geometry and task-vector alignment to improve unlearning under data heterogeneity. Upon an unlearning request, GDFA constructs multiple perturbed models around the trained global model and fine-tunes them on the forgetting data to derive task vectors representing the target knowledge; directionally consistent components are then merged through sign consensus and negated from the global model. By operating around flatter loss regions, GDFA aims to improve robustness to parameter modification and preserve retained knowledge, but requires request-time access to the forgetting data and incurs additional computation from generating and fine-tuning multiple perturbed models.