Fast Federated Machine Unlearning with Nonlinear Functional Theory

Tianshi Che [1] **Yang Zhou** [1] **Zijie Zhang** [1] **Lingjuan Lyu** [2] **Ji Liu** [3] **Da Yan** [4] **Dejing Dou** [5] **Jun Huan** [6]

[6]

Abstract

Federated machine unlearning (FMU) aims to remove the influence of a specified subset of training data upon request from a trained federated learning model. Despite achieving remarkable performance, existing FMU techniques suffer from inefficiency due to two sequential operations of training and retraining/unlearning on large-scale datasets. Our prior study, PCMU, was proposed to improve the efficiency of centralized machine unlearning (CMU) with certified guarantees, by simultaneously executing the training and unlearning operations. This paper proposes a fast FMU algorithm, FFMU, for improving the FMU efficiency while maintaining the unlearning quality. The PCMU method is leveraged to train a local machine learning (MU) model on each edge device. We propose to employ nonlinear functional analysis techniques to refin the local MU models as output functions of a Nemytskii operator. We conduct theoretical analysis to derive that the Nemytskii operator has a global Lipschitz constant, which allows us to bound the difference between two MU models regarding the distance between their gradients. Based on the Nemytskii operator and average smooth local gradients, the global MU model on the server is guaranteed to achieve close performance to each local MU model with the certified guarantees.

1. Introduction

Machine unlearning (MU) aims to give data holders the right to remove the influence of a certain subset of data from a trained machine learning (ML) model, while maintaining the accuracy of the ML model on remaining data (Cao & Yang, 2015; Golatkar et al., 2020a; Shibata et al., 2021; Ginart et al., 2019; Guo et al., 2020; Garg et al., 2020; Gupta et al., 2021; Wu et al., 2022b; Nguyen et al., 2022). Existing research efforts on machine unlearning can be divided into two groups: (1) Centralized machine unlearning (CMU), where all holders' training data collected by the server to unlearn centralized ML models (Golatkar et al., 2020a;b; Guo et al., 2020; Wu et al., 2020b; Nguyen et al., 2020; Izzo et al., 2021; Neel et al., 2021; Khan & Swaroop, 2021; Bourtoule et al., 2021; Ullah et al., 2021; Chen et al., 2021; 2022; Fu et al., 2022; Lu et al., 2022; Setlur et al., 2022; Suriyakumar & Wilson, 2022; Chundawat et al., 2022; Zeng et al., 2022; Liu et al., 2022b; Chourasia et al., 2023; Warnecke et al., 2023; Jagielski et al., 2023; Pawelczyk et al., 2023; Jang et al., 2023) and (2) Federated machine unlearning (FMU) for forgetting the target data from federated learning (FL) models when full access to all training data becomes unavailable (Liu et al., 2020; 2021a; Gong et al., 2021a; Liu et al., 2021b; Gong et al., 2021b; Yuan et al., 2022; Pan et al., 2022; Fraboni et al., 2022; Wang et al., 2022a; Liu et al., 2022c; Wu et al., 2022c;a; Gao et al., 2022; Halimi et al., 2022; Cao et al., 2023).

To the best of our knowledge, a common property of the above methods in either CMU or FMU settings need to sequentially perform two expensive operations: training a ML model on the whole dataset and producing an unlearning model, by either retraining a new ML model on the remaining data or directly unlearning the original ML model. This strategy of sequential execution is computationally expensive when training complex models over large datasets. The above efficiency issue becomes much worse in the FMU, since edge devices in the FMU often have limited computational resources (Dhar et al., 2021), such as smartphones, image sensors, Internet-of-Things devices, and wearable devices. The combination of high-dimensional models and constrained edge devices drastically limits the applicability of the FMU models in real world.

Our prior study, PCMU (Zhang et al., 2022b), presented a prompt certified MU approach based on randomized gradient smoothing and quantization. It is simultaneously executes the training and unlearning operations for improving the CMU efficiency. The PCMU method leverages randomized smoothing (RS) for certified robustness (CR) on

[1] Auburn University, USA [2] Sony AI, Japan [3] Baidu Research, China [4] University of Alabama at Birmingham, USA [5] Boston Consulting Group, USA [6] AWS AI Labs, USA. Correspondence to: Yang Zhou <yangzhou@auburn.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

[6] Work Done Before Joining AWS.

classification (Cohen et al., 2019) to do RS for CMU on gradient quantization. The authors treat data removals in the CMU as perturbations on the whole dataset. Analogously, they consider adversarial attacks in the CR as perturbations on the data samples. In addition, they analogize output quantized gradients in the CMU to output discrete class labels in the CR. Since the output class labels in the CR with RS are able to keep unchanged against adversarial attacks within a certified radius, the output quantized gradients in the CMU with RS can keep unchanged against data removals within a certified budget. This implies that the CMU model with RS directly trained on the whole dataset shares the same gradients (and parameters) with the one retrained on only the remaining data. The authors also derive the certified radius regarding the gradient change before and after data removals and the certified budget of data removals (i.e., the maximally allowed amount of escaped data samples).

It is challenging to directly extend the idea of PCMU to the FMU settings. Due to FL's privacy requirement, the access by the server to local training data on the edge devices becomes unavailable. It is impossible to utilize the PCMU method to train a CMU model on the server. On the other hand, if local MU models are trained on the edge devices and the standard aggregation methods (e.g., FedAvg (McMahan et al., 2017)) are used to generate a global MU model on the server, then it is difficult to guarantee the certified radius and budget of data removals of the global MU model.

This work aims to extend the PCMU technique to the FMU setting for improving the FMU efficiency while maintaining the unlearning quality, by leveraging the theory of nonlinear functional analysis, including Nemytskii operator and Fréchet differentiable smooth manifolds.

First, given a specific ML task (e.g., image classification), a local ML model $f(x;\bar{G}_k)$ ($x \in \mathbb{R}^n$) is trained on each edge device k with only its local data, where x is a data sample and $\bar{G}_k \in \mathbb{R}^T$ is the gradient of the local ML model. The PCMU method (Zhang et al., 2022b) is leveraged to transform the original gradient $\bar{G}_k$ into its smooth and quantized version $Q(\bar{G}_k+\varepsilon)$ for the purpose of CMU, where Q is a gradient quantizatier to map the continuous gradients over a discrete three-class space $\{-1,0,1\}$ and $\varepsilon \sim \mathcal{N}\left(0,\sigma^2 I\right)$ is the isotropic Gaussian distribution (See Eqs.(20)-(21) in the PCMU paper). Thus, a local MU model $g(x;Q(\bar{G}_k)) = \underset{\varepsilon}{\mathbb{E}}\big(f(x;Q(\bar{G}_k+\varepsilon))\big)$, $\varepsilon \sim \mathcal{N}(0,\sigma^2 I)$ is generated on device k. For ease of presentation, we use symbol q_k to replace $Q(\bar{G}_k)$.

Second, by leveraging the theory of nonlinear functional analysis, the local MU models $g(x;q_k)$ are reformulated as output functions of a Nemytskii operator $O(q_k)(x)$, which maps the gradient space to a function space. The theoretical analysis demonstrates that the Nemytskii operator $O(q_k)(\cdot)$ is smooth and induces a Fréchet differentiable smooth manifold $O(\mathbb{R}^3)$. The further analysis shows that the smooth manifold has a global Lipschitz constant $\frac{C}{\sqrt{2\pi}\sigma}$, i.e. $\|O(q_k)(x)-O(q_l)(x)\| \leq \frac{\|q_k-q_l\|_2}{\sqrt{2\pi}\sigma} \leq \frac{C\|\bar{G}_k-\bar{G}_l\|_2}{\sqrt{2\pi}\sigma}$ for any $\bar{G}_k,\bar{G}_l \in \mathbb{R}^T$, where C is the Lipschitz constant of Q. This global Lipschitz property of $O(q_k)$ allows to bound the difference between two MU models $g(\cdot;q_k)$ and $g(\cdot;q_l)$ regarding the distance $\|\bar{G}_k-\bar{G}_l\|_2$ between their gradients.

Last but not least, a global gradient $\bar{G}$ is aggregated on the server by averaging the gradients of all the local ML models, i.e., $\bar{G}=\frac{\bar{G}_1+\cdots+\bar{G}_K}{K}$, where K is the number of edge devices in the FMU. Thus, the global MU model $O(q)(\cdot)$ on the server is parameterized with a smooth gradient $Q(\bar{G}+\varepsilon)$, where $q=Q(\bar{G})$. Based on the global Lipschitz property of $O(q)$, we theoretically derive the global MU model $O(q)(x)$ on the server has a certified guarantee of $\frac{(K-1)Cd}{\sqrt{2\pi}K\sigma}$, where $d=\max\limits_{1\leq k,l\leq K}\|\bar{G}_k-\bar{G}_l\|$. Namely, for $k=1,\cdots,K$, it holds $|O(q)(x)-O(q_k)(x)| \leq \frac{(K-1)Cd}{\sqrt{2\pi}K\sigma}$ for any input sample x. This certified guarantee of $O(q)(x)$ ensures that the global MU model $O(q)(x)$ is close to each local MU model $O(q_k)(x)$ on the edge devices within distance $\frac{(K-1)Cd}{\sqrt{2\pi}K\sigma}$, which implies $O(q)(x)$ with bounded errors can maintain the certified radius and budget of data removals of the local MU models to a certain degree.

Our FMU method based on the theory of nonlinear functional analysis exhibits three compelling strengths against the existing FMU techniques: (1) It inherits the superior efficiency of the PCMU method by simultaneously executing the training and unlearning operations; (2) The global MU model on the server is guaranteed to achieve close performance to each local MU model with the certified CMU guarantees; and (3) The MU training is conducted on only the edge devices, which satisfies the FL's privacy requirement. Empirical evaluation on real datasets demonstrates the superior performance of our FMU model against several state-of-the-art techniques.

2. Preliminaries

2.1. Federated Machine Unlearning

First, given a ML task (e.g., image classification), K edge devices with their local training data $D=\{D_1,\cdots,D_K\}$, and a server, federated learning (FL) aims to learn a global ML model on the server by optimizing the problem below.

$$\begin{aligned}&\min_{W\in\mathbb{R}^d}\mathcal{L}(W)=\sum_{k=1}^{K}\frac{N_k}{N}L_k(W)\\&\text{where } L_k(W)=\frac{1}{N_k}\sum_{\{x_i,y_i\}\in D_k} l_i(W)\end{aligned} \tag{1}$$

where $l_i(W)=l(x_i,y_i;W)$ denotes the loss function of the prediction on data example $\{x_i,y_i\}\in D_k$ made with a global model parameter W. $N_k = |D_k|$ denotes the size of local dataset D_k. N is the size of total training data D, i.e., $N = N_1 + \cdots + N_K$. In the FL, the global model parameter W is iteratively updated with the aggregation of all local model parameters $W_1, \cdot, W_K$ on K devices in each round, i.e., $W = \sum_{k=1}^{K} \frac{N_k}{N} W_k$.

Second, the devices submit data removal requests at a certain time. The complete training data D is partitioned into two subsets: $D^f \subseteq D$ denoting the data which we wish the ML model to forget and $D^r \subseteq D$ specifying the data which we want the model to remember ($D = D^f \cup D^r$). The goal of federated machine unlearning (FMU) is to unlearn the forgotten data D^f, i.e., eliminate the influence of D^f from W. A straightforward solution is to use the remembered data D^r as the training data to retrain new local models on the edge devices with the data removal requests from scratch and to produce a new global model W^r on the server. However, this naive method is often time-consuming over large-scale datasets. An efficient FMU algorithm is to directly generate a sanitized model $\tilde{W}^r$ based on the deployed model W, D, and D^f to approximate W^r, i.e., $\tilde{W}^r \approx W^r$.

2.2. Randomized Gradient Smoothing and Quantization for Centralized Machine Unlearning

Let $\bar{G} = \frac{1}{N} \sum_{\{x_i, y_i\} \in D} G(x_i, y_i)$ be the gradient average on all data samples, where D is the training data, N is the size of training data D, i.e., $N = |D|$, $G(x_i, y_i) \in \mathbb{R}^T$ is the gradient of a ML model on a data sample $\{x_i, y_i\} \in D$, T is the dimension of the gradient $G(x_i, y_i)$.

The randomized gradient smoothing for centralized machine unlearning (CMU) on gradient quantization in the PCMU method (Zhang et al., 2022b) is given as follows.

$$S(\bar{G}) = \operatorname*{argmax}_{c \in \{-1,0,1\}} \mathop{\mathbb{P}}_{\varepsilon \sim \mathcal{D}} (Q(\bar{G} + \varepsilon) = c) \tag{2}$$

where $\mathcal{D} = \mathcal{N}\left(0, \sigma^2 I\right)$ is a Gaussian distribution. Q is a gradient quantizatier to map each dimension of the continuous gradient $G(x, y) \in \mathbb{R}^T$ over a discrete three-class space $\{-1, 0, 1\}$, for mimicking the classification in the randomized smoothing for certified robustness. S is a smooth version of Q. S returns whichever gradient classes Q^t is most likely to return when $\bar{G}$ is perturbed by noise ε.

The authors theoretically derive the certified radius R regarding the data change and the certified radius R' about the gradient change of the MU model before and after data removals. The following theorem analyzes the correlation between two types of certified radii and the certified budget of data removals (Zhang et al., 2022b).

Theorem 1. *Let L be the Lipschitz constant of gradient $G(x, y) \in \mathbb{R}^T$, then*

$$R \geq \frac{\sqrt{T}}{L} R' \tag{3}$$

The certified budget B' of data removal from R' is

$$B' \leq N - \frac{36 d L^2}{T(\Phi^{-1}(\underline{p_A}') - \Phi^{-1}(\overline{p_B}'))^2} \tag{4}$$

where Φ^{-1} is the inverse of the standard Gaussian CDF. Let $p_c(\bar{G})$ be the output probability of Q over gradient class c, i.e., $p_c(\bar{G}) = \mathop{\mathbb{P}}_{\varepsilon \sim \mathcal{D}}(Q(\bar{G} + \varepsilon) = c)$. p'_A and p'_B are the probabilities on the most probable class c_A and the runner-up class c_B respectively.

The above theorem shows that the smooth gradient quantizatier S can always output the correct and unchanged quantized gradients as long as the data removals B' (i.e., the number of escaped data samples in D) is within a certified budget of $N - \frac{36dL^2}{T(\Phi^{-1}(\underline{p_A}') - \Phi^{-1}(\overline{p_B}'))^2}$. This implies that the CMU model with randomized gradient smoothing and quantization directly trained on the whole dataset D shares the same gradients (and parameters) with the one retrained on only the remembered data D^r, which is the gold standard for evaluating the MU performance.

3. Fast Federated Machine Unlearning

In this work, we first train a local ML model $f(x; \bar{G}_k)$ ($x \in \mathbb{R}^n$) on each edge device k and leverage the PCMU method (Zhang et al., 2022b) to transform the original model gradient $\bar{G}_k$ into its smooth and quantized version $Q(\bar{G}_k + \varepsilon)$ for the fast MU on the edge devices. We further generate a local MU model $g(x; Q(\bar{G}_k)) = \mathop{\mathbb{E}}_{\varepsilon}\big(f(x; Q(\bar{G}_k + \varepsilon))\big)$, $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$ on device k. By leveraging the theory of nonlinear functional analysis, we reformulate $g(x; q_k)$ as output functions of a Nemytskii operator $O(q_k)(x)$, where $q_k = Q(\bar{G}_k)$. We theoretically prove that the Nemytskii operator $N(q_k)(\cdot)$ is smooth and induces a Fréchet differentiable smooth manifold $N(\mathbb{R}^3)$. The further analysis shows that the smooth manifold has a global Lipschitz constant $\frac{C}{\sqrt{2\pi}\sigma}$. Based on global Lipschitz property of $O(\mathbb{R}^3)$, we derive the bounded difference between the global MU model $O(q)(x)$ on the server and the local MU model $O(q_k)(x)$ on each device k, where $q = Q(\bar{G})$. Thus, the global MU model on the server is guaranteed to achieve close performance to each local MU model with the certified CMU guarantees. This help $O(q)(x)$ maintain the certified radius and budget of data removals of the local MU models based on the PCMU method to a certain degree.

Nonlinear functional analysis is a branch of mathematical analysis that deals with nonlinear mappings (i.e., nonlinear operators) between infinite-dimensional vector spaces and certain classes of nonlinear spaces (Riesz & Sz.-Nagy, 1955). The following definitions describe the differentiability properties of operators between Banach spaces and the Nemytskii operator.

In functional analysis, a Banach space is a complete normed vector space. It is often used for the computation of vector length and distance between vectors.

Definition 1. *A Banach space is a complete normed space $(X, \|\cdot\|)$, where it is complete if any Cauchy sequence in X has a limit. Namely, for every Cauchy sequence $x_1, x_2, \cdots \in X$, there exists some $x \in X$, such that the sequence's convergence to x can be expressed as $\lim\limits_{n\to\infty} \|x_i - x\| = 0$. The norm $\|\cdot\|$ of a normed space $(X, \|\cdot\|)$ is a complete norm if $(X, \|\cdot\|)$ is a Banach space.*

L^p spaces are function spaces defined using a natural generalization of the p-norm for finite-dimensional vector spaces. Specifically, $(\mathbb{R}^n, \|\cdot\|_{L^p})$ is the vector space $\mathbb{R}^n$ equipped with the L^p-norm, i.e.,

$$\begin{aligned}&\|(x_1, x_2, \cdots, x_n)\|_{L^p} = (\sum_{i=1}^{n} |x_i|^p)^{1/p} \text{for } 1 \leq p < \infty;\\ &\|(x_1, x_2, \cdots, x_n)\|_{L^\infty} = \sup_{1\leq i\leq n} |x_i|\end{aligned} \tag{5}$$

The L^p function spaces are defined using a natural generalization of the L^p-norm for finite-dimensional vector spaces.

Next, we define $L^p(\Omega)$ Space. When $1 \leq p < \infty$,

$$L^p(\Omega) = \{h(x) : \Omega \subset \mathbb{R}^n \to \mathbb{R}, \int_\Omega |h(x)|^p dx < \infty\} \tag{6}$$

equipped with the norm $\|h(x)\|_{L^p} = (\int_\Omega |h(x)|^p dx)^{1/p}$.

When $p = \infty$,

$$L^\infty(\Omega) = \{h(x) : \Omega \subset \mathbb{R}^n \to \mathbb{R}, \sup_{x\in\Omega}|h(x)| < \infty\} \tag{7}$$

equipped with the norm $\|h(x)\|_{L^\infty} = \sup\limits_{x\in\Omega}|h(x)|$.

Definition 2. *Let X, Y be two Banach spaces. The operator $L : X \to Y$ is linear if and only if $L(\alpha x_1 + \beta x_2) = \alpha L(x_1) + \beta L(x_2)$ for any $\alpha, \beta \in \mathbb{R}$ and $x_1, x_2 \in X$. The operator norm of L is defined by $\|L\|_{op} = \sup\limits_{x\in X, x\neq 0} \frac{\|L(x)\|_Y}{\|x\|_X}$.*

Definition 3. *Let $\Omega \subset \mathbb{R}^n$ be a domain and Y be a Banach space. Given a functional $F : \Omega \times X \to \mathbb{R}$ for any $y \in Y$, a new functional $O(y) : \Omega \to \mathbb{R}$ is defined as $O(y)(x) = F(x; y)$. The operator O is a Nemytskii operator.*

In order to better understand the concept of Nemytskii operators, we use linear functions as an example to explain it. The linear functions are essentially linear functionals from Banach spaces to $\mathbb{R}$, as well as a Nemytskii operator from certain Banach spaces to the space of bounded linear functionals from L^p to $\mathbb{R}$, i.e., $B(L^p)$. Consider a linear function $h(x) = qx$, where $q, x \in \mathbb{R}^n$. If the domain of x is a Banach space $(\mathbb{R}^n, \|\cdot\|_{L^p})$, then $g(x)$ can be treated as a functional from L^p to $\mathbb{R}$. Based on the Hölder Inequality, we can get $|h(x)| = |qx| \leq \|q\|_{L^{p'}} \|x\|_{L^p}$, where p' is the conjugacy of p satisfying $\frac{1}{p} + \frac{1}{p'} = 1$. This implies that the operator norm $\|h\|_{op}$ of the linear functional $h(x)$ is $\|h\|_{op} = \sup\limits_{x\in\mathbb{R}^n, x\neq 0} \frac{|h(x)|}{\|x\|_p} = \|q\|_{p'}$. Thus, $h(x)$ is a bounded linear functional from L^p to $\mathbb{R}$ for any $q \in L^{p'}$, i.e., $g(x) \in B(L^p)$ for any $q \in L^{p'}$. If we further consider q as a vector, the mapping $P(q)$:$q \to h(x) \in S(L^p)$, i.e., $P(q)(x) = h(x)$, can be viewed as an operator from the Banach space $L^{p'}$ to the Banach space $S(L^p)$. The operator $P(q) : L^{p'} \to S(L^p)$ is a Nemytskii operator.

In terms of the above analysis, we define a hypothesis space $\mathcal{H}$ ($\mathcal{H} = \{f(x; Q(\bar{G}))|\forall Q(\bar{G}))\}$) as a Banach manifold in L^p space. We reformulate the non-smooth models $f(x; Q(\bar{G}))$ as output functions of a Nemytskii operator $P : \mathbb{R}^3 \to L^p$ with $P(q)(x) = f(x; Q(\bar{G}))$. We define another Nemytskii operator $O : \mathbb{R}^3 \to L^p$ with the randomized gradient smoothing and quantization as follows.

$$O(q)(\cdot) = \mathop{\mathbb{E}}_{\varepsilon}\big(f(\cdot; Q(\bar{G} + \varepsilon))\big),\ \varepsilon \sim \mathcal{N}(0, \sigma^2 I) \tag{8}$$

Namely, we have

$$\begin{aligned}O(q)(\cdot) &= O(Q(\bar{G}))(\cdot)\\ &= \frac{1}{(2\pi\sigma^2)^{3/2}} \int_{\mathbb{R}^3} P(Q(\bar{G} + \varepsilon))(\cdot)e^{-\frac{\|\varepsilon\|_2^2}{2\sigma^2}} d\varepsilon\end{aligned} \tag{9}$$

Definition 4. *Let X and Y be two Banach spaces, and o is an operator from X to Y. The operator o is called Fréchet differentiable at $x \in X$ if there exists a bounded linear operator $L : X \to Y$, such that*

$$\lim_{\|\Delta\|_X\to 0} \frac{\|o(x + h) - o(x) - L(x)\Delta\|_Y}{\|\Delta\|_X} \to 0 \tag{10}$$

The linear operator $L(x)$ is called the Fréchet derivative of o at x.

The following theory demonstrates that the Nemytskii operator $O(q)(\cdot)$ is smooth and induces a Fréchet differentiable smooth manifold $O(\mathbb{R}^3)$.

Theorem 2. *O is Fréchet differentiable and $O(\mathbb{R}^3) \subset L^p$ is a smooth manifold.*

Please refer to Appendix A.1 for detailed proof of Theorem 2.

Now, we analyze the advantage of the randomized gradient smoothing techniques and the reason why we choose Nemytskii operator $O(q)(x)$ or $O(q_k)(x)$ as global model or local model in the FMU setting. Since $O(\mathbb{R}^3)(\cdot)$ is a smooth manifold, it guarantees that the gradient quantizatiers q_k are trainable. Most importantly, the smooth manifold $O(\mathbb{R}^3)(\cdot)$ has a global Lipschitz constant which is independent of the input data, as demonstrated in Theorem 3. This independence property ensures that the certified MU guarantees of the smooth local models can be maintained to a certain degree against any data removals within the certified budget. On the other hand, the manifold $P(\mathbb{R}^3)(\cdot)$ without the randomized gradient smoothing is only locally Lipschitz with respect to q_k and the Lipschitz constant is determined by the input data. In addition, the Lipschitz constant regarding q_k could be rather large since it is hard to control the amplification of difference through propagation over neural networks and thus the Lipschitz constant keeps increasing with the number of layers, which prevents the global model from preserving the certified MU guarantees of the local models in the FMU.

Theorem 3. *Let $\bar{G}_k \in \mathbb{R}^T$ be the gradient of the local ML model on edge device k, Q be the gradient quantizatier, C be the Lipschitz constant of Q, $q_k = Q(\bar{G}_k)$, and O be the smooth Nemytskii operator. If $\|\bar{G}_k - \bar{G}_l\|_2 \leq d$ for any $q_k, q_l \in \mathbb{R}^3$, then we have*

$$|\Omega|^{-1}\|O(q_k)(x) - O(q_l)(x)\|_{L^p(\Omega)} \leq \frac{Cd}{\sqrt{2\pi}\sigma} \text{ if } 1 \leq p < \infty,$$
$$\|O(q_k)(x) - O(q_l)(x)\|_{L^p(\Omega)} \leq \frac{Cd}{\sqrt{2\pi}\sigma} \text{ if } p = \infty \tag{11}$$

for any $\Omega \subset \mathbb{R}^n$.

Please refer to Appendix A.1 for detailed proof of Theorem 3.

According to Theorem 3, the following theorem demonstrates that the smooth global model $O(q)(x)$ on the server is close to each smooth local model $O(q_k)(x)$ on the edge devices within distance $\frac{(K-1)Cd}{\sqrt{2\pi}K\sigma}$ for the preservation of the MU certificates of the smooth local models based on the PCMU method to a certain degree.

Theorem 4. *Let $\bar{G}_k \in \mathbb{R}^T$ be the gradient of the local ML model on edge device k, Q be the gradient quantizatier, C be the Lipschitz constant of Q, $q_k = Q(\bar{G}_k)$, O be the smooth Nemytskii operator, and $O(q_k)(x)$ be the smooth local model on device k. Let $\bar{G} = \frac{\bar{G}_1+\cdots+\bar{G}_K}{K}$ be the gradient of the global ML model on the server by averaging the gradients of all the local ML models, $q = Q(\bar{G})$, and $O(q)(x)$ be the smooth global model on the server. If $\max_{1\leq k,l\leq K}\|\bar{G}_k - \bar{G}_l\|_2 = d$, then we have*

$$|\Omega|^{-1}\|O(q)(x) - O(q_k)(x)\|_{L^p(\Omega)} \leq \frac{(K-1)Cd}{\sqrt{2\pi}K\sigma} \text{ if } 1 \leq p < \infty,$$
$$\|O(q)(x) - O(q_k)(x)\|_{L^p(\Omega)} \leq \frac{(K-1)Cd}{\sqrt{2\pi}K\sigma} \text{ if } p = \infty \tag{12}$$

for any $\Omega \subset \mathbb{R}^n$.

Please refer to Appendix A.1 for detailed proof of Theorem 4.

Therefore, $O(q)(x)$ on the server has a certified guarantee of $\frac{(K-1)Cd}{\sqrt{2\pi}K\sigma}$ in L^p space for any $1 \leq p \leq \infty$. Namely, this certified guarantee of $O(q)(x)$ ensures that the smooth global model $O(q)(x)$ is close to each smooth local model $O(q_k)(x)$ on the edge devices within distance $\frac{(K-1)Cd}{\sqrt{2\pi}K\sigma}$. Notice that Theorem 4 is satisfied for any $\Omega \subset \mathbb{R}^n$, that is, the certified guarantee is independent of the input data. Therefore, it assures the closeness between the smooth global and local models on any data sample $x \in \mathbb{R}^n$.

The standard deviation σ in Gaussian noise in the randomized gradient smoothing serves as a tradeoff hyperparameter to well balance the MU performance and prediction accuracy achieved by the smooth global model. A larger σ results in higher closeness between the smooth global and local models and thus better preservation of the certified MU guarantees by the smooth local models, while a smaller σ leads to better prediction accuracy. Especially, when $\sigma \to 0$, the smooth global model $O(q)(x)$ on the server converges to the non-smooth one $P(q)(x)$, which is validated by Theorem 5. In the PCMU method (Zhang et al., 2022b), the randomized gradient smoothing is a necessary step to provide the certified MU guarantees (i.e., the certified radius regarding the gradient change and the certified budget of data removals) of the smooth models. the randomized gradient smoothing, $P(q)(x)$ fails to provide any certificate guarantees for the FMU task.

Theorem 5. *Let $\bar{G}$ be the gradient of the global ML model on the server, Q be the gradient quantizatier, $q = Q(\bar{G})$, and O and P be the smooth and non-smooth Nemytskii operators respectively. $O(q)(x) \to P(q)(x)$ for any data sample $x \in \mathbb{R}^n$ if $\sigma \to 0$.*

Please refer to Appendix A.1 for detailed proof of Theorem 5.

FFMU model training. On the device side, a local ML model $f(x; \bar{G}_k)$ ($x \in \mathbb{R}^n$) is trained on each edge device k with only its local data. The PCMU method (Zhang et al., 2022b) is leveraged to transform the original gradient $\bar{G}_k$ into its smooth and quantized version $Q(\bar{G}_k + \varepsilon)$. The local MU model is reformulated as output functions of a smooth Nemytskii operator $O(q_k)(x)$.

$$O(q_k)(x) = \mathbb{E}_{\varepsilon}\big(f(x; Q(\bar{G}_k + \varepsilon))\big),\ \varepsilon \sim \mathcal{N}(0, \sigma^2 I) \tag{13}$$

On the server side, the gradient $\bar{G}$ of the global ML model on the server is aggregated by averaging the gradients of all the local ML models.

$$\bar{G} = \frac{\bar{G}_1 + \cdots + \bar{G}_K}{K} \tag{14}$$

Table 1: Performance with 10% data removal and CNN on Fashion-MNIST

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	84.40	15.60	14.70	15.75	211	1,636	1,847
Knowledge Distillation	81.05	18.95	17.51	20.28	213	2,734	2,947
Rapid Retraining	81.86	18.14	17.56	17.50	**198**	1,850	2,048
MacForget	82.36	17.64	16.82	16.81	227	1,830	2,057
FedEraser	81.64	18.36	17.70	16.72	369	3,610	3,979
VeriFi	77.31	22.69	22.29	19.89	234	1,943	2,177
Class-Discriminative Pruning	81.10	18.90	17.61	19.03	232	1,719	1,951
UN	80.87	19.13	17.90	16.69	234	1,965	2,199
RCAD	79.44	20.56	19.92	18.97	493	5,865	6,358
IJ	82.47	17.53	17.22	13.24	232	1,846	2,078
Noisy-GD	83.56	16.44	16.01	14.37	232	1,866	2,098
FFMU	**84.55**	**15.45**	**14.88**	**15.07**	1,288	**0**	**1,288**

Table 2: Performance with 20% data removal and CNN on Fashion-MNIST

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	84.39	15.61	14.74	15.12	198	2,340	2,538
Knowledge Distillation	81.58	18.42	17.63	19.69	227	5,459	5,686
Rapid Retraining	82.34	17.66	16.49	16.71	**187**	3,658	3,845
MacForget	81.02	18.98	18.37	16.75	213	3,608	3,821
FedEraser	79.51	20.49	19.33	18.65	369	3,610	3,979
VeriFi	81.65	18.35	16.53	19.82	219	3,908	4,127
Class-Discriminative Pruning	81.89	18.11	16.93	18.59	240	3,409	3,649
UN	82.20	17.80	16.43	18.85	221	3,935	4,156
RCAD	79.27	20.73	20.40	18.38	346	7,091	7,437
IJ	82.17	17.83	16.51	19.00	232	3,622	3,854
Noisy-GD	81.98	18.02	17.07	17.68	244	3,751	3,995
FFMU	**84.55**	**15.45**	**14.83**	**15.19**	1,288	**0**	**1,288**

The global gradient $\bar{G}$ is converted to its smooth and quantized version $Q(\bar{G}+\varepsilon)$.

The global MU model is reformulated as output functions of a smooth Nemytskii operator $O(q)(x)$.

$$O(q)(x) = \mathop{\mathbb{E}}_{\varepsilon}\big(f(x; Q(\bar{G}+\varepsilon))\big),\ \varepsilon \sim \mathcal{N}(0, \sigma^2 I) \qquad (15)$$

After the model training, $O(q)(x)$ will be the output of our FFMU algorithm for fast federated machine unlearning.

4. Experiments

In this section, we have evaluated the effectiveness of our FFMU model and other comparison methods for federated machine unlearning over three popular image classification datasets: Fashion-MNIST (Xiao et al., 2017; Gupta et al., 2021; Fu et al., 2022), CIFAR-10 (Krizhevsky, 2009; Golatkar et al., 2020a;b; Thudi et al., 2021a; Gupta et al., 2021; Fu et al., 2022), and SVHN (Netzer et al., 2011; Guo et al., 2020; Bourtoule et al., 2021). We train the classifiers on the training set and test them on the test set for three datasets. We train a convolutional neural network (CNN) on Fashion-MNIST for clothing classification. We train LeNet over CIFAR-10 for image classification. We apply the ResNet-18 architecture on SVHN for street view house number identification. We evaluate the performance of various machine unlearning methods on three datasets with the ratio of data removal between 5% and 20%.

Baselines. We compare the FFMU model with ten state-of-the-art federated machine unlearning models. **Knowledge Distillation** is a federated unlearning method to eliminate a client's contribution by subtracting the accumulated historical updates from the model and leveraging the knowledge distillation method to restore the model's performance without using any data from the clients (Wu et al., 2022a). **Rapid Retraining** is a rapid retraining approach to fully erase data samples from a trained FL model (Liu et al., 2022c). **MacForget** introduced a mask gradient generator that continuously generates mask gradients, and apply them to the neurons of the neural network and stimulate them to unlearn the memorization of the given samples (Liu et al., 2020). **FedEraser** is a federated unlearning methodology that can eliminate the influence of a federated client's data on the global federated learning (FL) model while significantly reducing the time used for constructing the unlearned FL model (Liu et al., 2021a). **VeriFi** is a unified framework integrating federated unlearning and verification that allows systematic analysis of the unlearning and quantification of its effect, with different combinations of multiple unlearning and verification methods (Gao et al., 2022). **Class-Discriminative Pruning** proposed to utilize CNN channel pruning to guide the federated machine unlearning process (Wang et al., 2022b). **UN** performs unlearning at the client (to be erased) by reversing the learning process, i.e., training a model to maximize the local empirical loss (Halimi et al., 2022). **RCAD** can unlearn spurious features in the

Table 3: Performance with 10% data removal and LeNet on CIFAR-10

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	53.66	46.34	42.67	47.76	165	1,666	1,831
Knowledge Distillation	52.32	47.68	47.09	49.66	143	1,999	2,142
Rapid Retraining	49.68	50.32	47.82	49.56	**144**	1,345	1,489
MacForget	51.98	48.02	47.75	48.21	151	1,358	1,509
FedEraser	52.84	47.16	45.89	46.28	375	4,160	4,543
VeriFi	49.56	50.44	49.05	51.88	**144**	1,429	1,573
Class-Discriminative Pruning	50.07	49.93	48.05	53.84	194	1,123	1,317
UN	52.13	47.87	46.21	48.50	187	1,441	1,628
RCAD	51.74	48.26	47.10	46.76	311	6,142	6,453
IJ	52.01	47.99	47.33	46.75	172	1,358	1,530
Noisy-GD	50.78	49.22	47.18	49.96	145	1,467	1,612
FFMU	**54.84**	**45.16**	**42.78**	**47.68**	986	**0**	**986**

Table 4: Performance with 20% data removal and LeNet on CIFAR-10

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	54.97	45.03	41.20	45.86	166	3,152	3,318
Knowledge Distillation	51.62	48.38	47.53	48.46	145	3,869	4,014
Rapid Retraining	51.67	48.33	46.79	44.97	**141**	2,664	2,805
MacForget	50.71	49.29	48.89	48.71	148	2,725	2,873
FedEraser	52.53	47.47	45.51	47.54	367	6,993	7,360
VeriFi	50.57	49.43	48.73	49.92	149	2,812	2,961
Class-Discriminative Pruning	52.27	47.73	46.46	47.88	165	2,297	2,462
UN	53.51	46.49	43.81	46.44	184	2,919	3,103
RCAD	49.60	50.40	49.73	47.84	313	7,890	8,203
IJ	51.39	48.61	47.60	46.90	183	2,825	3,008
Noisy-GD	51.77	48.23	46.75	47.50	145	2,960	3,105
FFMU	**54.84**	**45.16**	**42.59**	**46.00**	986	**0**	**986**

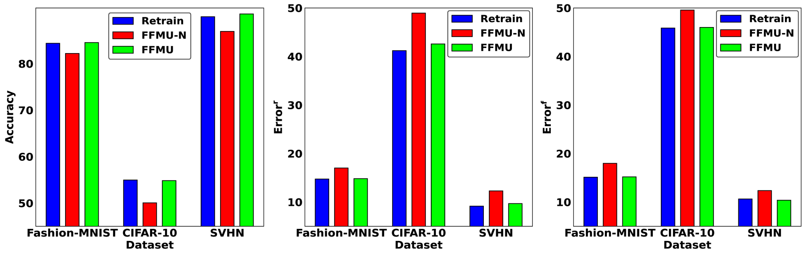
training data by increasing entropy only on examples generated along the adversarial direction (Setlur et al., 2022). **IJ** is an online unlearning algorithm that is both computationally and memory efficient by leveraging the infintesimal jacknife (Suriyakumar & Wilson, 2022). **Noisy-GD** is a robust data-deletion guarantee that can satisfy differential privacy to ensure true data deletion (Chourasia et al., 2023).

Variants of FFMU model. We evaluate two versions of FFMU to show the strengths of different techniques. FFMU-N uses the FedAvg method to aggregate the local MU models on the edge devices into a global MU model on the server. FFMU performs the aggregation process with Nemytskii operator transformation and average smooth local gradients. FFMU can guarantee the global MU model on the server to achieve close performance to each local MU model with the certified guarantees.

Evaluation metrics. By following the same settings in representative machine unlearning models (Golatkar et al., 2020a;b; Thudi et al., 2021a; Fu et al., 2022), we use four popular measures in machine unlearning to verify the performance of different methods: $Accuracy$, $Error^f$ (classification errors on the forgotten data D^f), $Error^r$ (errors on the remembered data D^r), and $Error^t$ (errors on the test data). Since the model W^r (**Retrain**) that uses only the remembered data D^r as the training data retrained from scratch has never seen the forgotten data D^f, it is often used as the gold standard for evaluating the unlearning performance (Golatkar et al., 2020a; Fu et al., 2022). Ideally, the accuracy and three errors of the unlearning models $\tilde{W}^r$ should match that of the retrained model W^r.

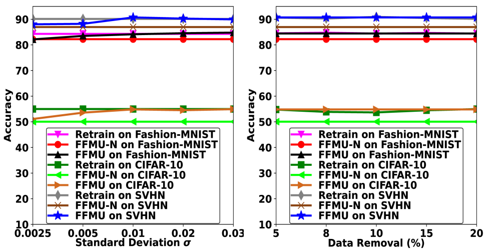
Machine unlearning accuracy with varying ratios of data removal. Tables 1-4 exhibit the accuracy obtained by twelve federated machine unlearning approaches by varying the ratio of unlearning request / data removal between 10% and 20%. Retrain represents the model retrained on only the remembered data D^r from scratch, without the knowledge of the forgotten data D^f. A federated machine unlearning algorithm with more similar performance to the Retrain model achieves a better unlearning result. It is observed that among eleven approaches except the Retrain model, no matter how large the ratios of data removal are, the FFMU method achieves the closest accuracy to the Retrain model in all tests, showing the effectiveness of FFMU to the federated machine unlearning. Compared to the absolute performance difference between other baselines and the Retrain model, FFMU, on average, achieves at least 22.09% and 9.43% improvement of absolute accuracy difference on Fashion-MNIST and CIFAR-10 respectively. In addition, the promising performance of FFMU over Fashion-MNIST and CIFAR-10 implies that FFMU has great potential as a general federated machine unlearning solution to other image datasets, which is desirable in practice.

Machine unlearning error with varying ratios of data removal. Tables 1-4 also show the classification errors on



(a) $Accuracy$ (b) $Error^r$ (c) $Error^f$

Figure 1: Performance of FFMU variants with 20% data removal



(a) Standard Deviation (b) Data Removal (%)

Figure 2: Performance with varying parameters

the deleted data D^f ($Error^f$), errors on the remembered data D^r ($Error^r$), and errors on the test data ($Error^t$) by twelve federated machine unlearning methods respectively. We have observed that the performance of our FFMU method behaves similarly and achieves at least 20.35% and 14.98% descrease of absolute error difference on two datasets respectively. FFMU substantially outperforms the performance of other baselines in most tests, especially on Fashion-MNIST. In addition, the errors by our FFMU are not sensitive to the ratio of data removals. This is because that our FFMU method performs one-time operation of simultaneous training and unlearning when addressing a series of machine unlearning requests, as long as the ratio of actual data removals is below the certified budget of data removals in our FFMU. However, other baselines need to sequentially handle these unlearning requests one by one.

Ablation study. Figure 1 exhibits the unlearning performance with the Retrain model and two variants of FFMU on three datasets. We have observed the FFMU achieves the closest accuracy and errors to the Retrain model over three datasets, which are obviously better than FFMU-N. A reasonable explanation is that FFMU leverages the PCMU method to utilize the randomized gradient smoothing and quantization for supporting certified MU on the edge devices. The global Lipschitz property of Nemytskii operator ensures the global MU model to achieve close performance to each local MU model, which implies the certified guarantees on the devices are maintained in the global MU model.

Running time. Tables 1-4 report the running time achieved by all comparison methods over two dataset to produce machine unlearning results respectively. We observe that FFMU scales well with deep neural network architectures over different image datasets and shows good efficiency for federated machine unlearning. Our FFMU method achieves better efficiency than other baseline methods in most experiments. In addition, our FFMU method performs one-time operation of simultaneous training and unlearning when addressing a series of machine unlearning requests. However, other baseline methods need to sequentially handle these machine unlearning requests one by one. This is clearly a computationally expensive process when the number of machine unlearning requests is huge.

Impact of standard deviation. Figure 2 (a) measures the performance effect of standard deviation of the Gaussian distribution in the randomized smoothing for machine unlearning by varying σ from 0.025 to 0.03. Notice that the Retrain and FFMU-N models do not contain the module of randomized smoothing. Thus, their accuracy scores keep unchanged with varying σ. We have witnessed the performance curves by FFMU initially increase quickly and then become stable or even slight drop when σ continuously increases. Initially, a large σ can help utilize the strength of randomized gradient smoothing and quantization for directly training a machine unlearning model in advance. Later on, when σ continues to increase and goes beyond some thresholds, the performance curves become stable. A rational guess is that after the data removals have been already certified at a certain threshold and considered in the training of machine unlearning models, our FFMU model is able to generate a good machine unlearning result. When σ continuously increases, this does not affect the performance of federated machine unlearning any more.

Impact of data removal ratio. Figure 2 (b) evaluates the accuracy impact of data removal ratios varying from 5% to 20% on three datasets of Fishion-MNIST, CIFAR-10, and SVHN. It is observed that when changing data removal ratios, the accuracy by our FFMU model matches well with that of the retrained model from scratch. The performance by our FFMU model keeps relatively stable, since our method directly trains a unlearning model based on the certified budget of data removals in advance and performs one-time operation of simultaneous training and unlearning, as long as the ratio of actual data removals is below the certified budget of data removals. Thus, we do not need to re-unlearn the model when a new unlearning request is coming. This shows the potential of our FFMU model on addressing a series of federated machine unlearning requests.

5. Related Work

(1) Centralized Machine Unlearning. Trustworthy machine learning has attracted active research in recent years (Palanisamy et al., 2018; Zhou et al., 2020b; Zhang et al., 2020; Zhou et al., 2021; Zhao et al., 2021; Ren et al., 2021; Zhang et al., 2021c;a; Zhou et al., 2022b; Jin et al., 2022b; Zhang et al., 2022b; Zhou et al., 2010; 2009; Cheng et al., 2011; Zhou & Liu, 2012; Cheng et al., 2012; Zhou & Liu, 2013; Su et al., 2013; Zhou & Liu, 2014; Su et al., 2015; Zhou & Liu, 2015; Zhou et al., 2015a; 2016; 2018b;a; Ren et al., 2019; Zhou et al., 2019b;a;c; Zhou & Liu, 2019; Wu et al., 2020a; 2021a; Zhou et al., 2020c;a; Jin et al., 2021; Wu et al., 2021b). Machine unlearning, one of important research topics in the trustworthy machine learning, is gaining attention in recent years (Cao & Yang, 2015; Ginart et al., 2019; Guo et al., 2020; Golatkar et al., 2020a; Garg et al., 2020; Shibata et al., 2021; Gupta et al., 2021; Wu et al., 2022b; Nguyen et al., 2022). Machine unlearning can be broadly grouped into two categories: exact unlearning and approximate unlearning methods. In exact unlearning, the impact of the data to be forgotten is excluded from the model, as if retraining the model on the remaining data from scratch (Cauwenberghs & Poggio, 2000; Karasuyama & Takeuchi, 2009; Cao & Yang, 2015; Ginart et al., 2019; Chen et al., 2019; Schelter, 2020; Li et al., 2021; Mahadevan & Mathioudakis, 2021; Brophy & Lowd, 2021; Schelter et al., 2021; Chen et al., 2022). In approximate unlearning, the methods aim to approximate the parameters that would have been obtained if the model was trained without using the data to be unlearned (Baumhauer et al., 2020; Graves et al., 2021; Golatkar et al., 2021; Thudi et al., 2021b; Liu et al., 2021a; Marchant et al., 2022; Zeng et al., 2022; Liu et al., 2022b; Chourasia et al., 2023).

Certified Machine Unlearning. Subsequent works follow similar approximate definitions in (Ginart et al., 2019) to provide certified unlearning guarantees for strongly-convex learning problems (Guo et al., 2020; Neel et al., 2021; Sekhari et al., 2021). certified removal is a certified-removal mechanism that applies a Newton step on the model parameters that largely remove the influence of the deleted data points (Guo et al., 2020). PCMU executes one-time operation of simultaneous training and unlearning in advance for a series of machine unlearning requests, without the knowledge of the forgotten data (Zhang et al., 2022b).

(2) Federated Machine Unlearning. Parallel, distributed, and federated learning have been extensively studied in recent years (Lee et al., 2019; Wu et al., 2021a; Goswami et al., 2020; Zhang et al., 2021b; Zhou et al., 2022a; Guo et al., 2022; Jin et al., 2022a; Zhang et al., 2022a; Che et al., 2022; Yan et al., 2022a; Liu et al., 2022a; Yan et al., 2022b;c; Liu et al., 2023; Li et al., 2022; Liu et al., 20213; Zhou & Liu, 2013; Zhou et al., 2014; Bao et al., 2015; Zhou et al., 2015b;c; Lee et al., 2015; Zhou, 2017). Although the Centralized machine unlearning techniques dominate the existing research efforts, federated machine unlearning has attracted active research in recent years (Liu et al., 2020; 2021a; Gong et al., 2021b; Liu et al., 2021b; Yuan et al., 2022; Pan et al., 2022; Fraboni et al., 2022). Wang et al. proposed to utilize CNN channel pruning to guide the federated machine unlearning process (Wang et al., 2022b). Liu et al. proposed a rapid retraining approach to fully erase data samples from a trained FL model (Liu et al., 2022c). Wu et al. developed a general pipeline for simultaneously three common types of federated unlearning requests: class unlearning, client unlearning, and sample unlearning (Wu et al., 2022c). Wu et al. proposed a novel federated unlearning method to eliminate a client's contribution by subtracting the accumulated historical updates from the model and leveraging the knowledge distillation method to restore the model's performance without using any data from the clients (Wu et al., 2022a). VeriFi is a unified framework integrating federated unlearning and verification that allows systematic analysis of the unlearning and quantification of its effect, with different combinations of multiple unlearning and verification methods (Gao et al., 2022). UN performs unlearning at the client (to be erased) by reversing the learning process, i.e., training a model to maximize the local empirical loss (Halimi et al., 2022). FedRecover can recover an accurate global model from poisoning attacks with small cost for the clients, by using the server to estimate the clients' model updates (Cao et al., 2023).

PCMU is the only method to simultaneously execute the training and unlearning operations for dramatically improving the unlearning efficiency in centralized setting (Zhang et al., 2022b). To the best of our knowledge, other machine unlearning methods iTo the best of our knowledge, a common property of the above methods in either centralized or federated settings need to sequentially perform two expensive operations: training a ML model on the whole dataset and producing an unlearning model, by either retraining a new ML model on the remaining data or directly unlearning the original ML model. This strategy of sequential execution is computationally expensive when training complex models over large datasets. Motivated the idea of PCMU, this work is the first to simultaneously execute the training and unlearning operations for improving the FMU efficiency while maintaining the unlearning quality, by leveraging the theory of nonlinear functional analysis, including Nemytskii operator and Fréchet differentiable smooth manifolds.

6. Conclusions

In this work, we proposed a fast FMU algorithm for improving the FMU efficiency while maintaining the unlearning quality. First, the PCMU method is leveraged to train a local MU model on each edge device. Second, the local MU models are reformulated as output functions of a Nemytskii operator. Based on the Nemytskii operator and average smooth local gradients, the global MU model on the server is guaranteed to achieve close performance to each local MU model with the certified guarantees. Finally, the theoretical analysis is conducted to bound the difference between two MU models regarding the distance between their gradients.

References

Bao, X., Liu, L., Xiao, N., Zhou, Y., and Zhang, Q. Policy-driven autonomic configuration management for nosql. In *Proceedings of the 2015 IEEE International Conference on Cloud Computing (CLOUD'15)*, pp. 245–252, New York, NY, June 27-July 2 2015.

Baumhauer, T., Schöttle, P., and Zeppelzauer, M. Machine unlearning: Linear filtration for logit-based classifiers. *CoRR*, abs/2002.02730, 2020.

Bourtoule, L., Chandrasekaran, V., Choquette-Choo, C., Jia, H., Travers, A., Zhang, B., Lie, D., and Papernot, N. Machine unlearning. In *Proceedings of the 42nd IEEE Symposium on Security and Privacy*, 2021.

Brophy, J. and Lowd, D. Machine unlearning for random forests. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 1092–1104. PMLR, 2021.

Cao, X., Jia, J., Zhang, Z., and Gong, N. Z. Fedrecover: Recovering from poisoning attacks in federated learning using historical information. In *44th IEEE Symposium on Security and Privacy, SP 2023, San Francisco, CA, USA, May 22-25, 2023*. IEEE, 2023.

Cao, Y. and Yang, J. Towards making systems forget with machine unlearning. In *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17-21, 2015*, pp. 463–480. IEEE Computer Society, 2015.

Cauwenberghs, G. and Poggio, T. A. Incremental and decremental support vector machine learning. In Leen, T. K., Dietterich, T. G., and Tresp, V. (eds.), *Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA*, pp. 409–415. MIT Press, 2000.

Che, T., Zhang, Z., Zhou, Y., Zhao, X., Liu, J., Jiang, Z., Yan, D., Jin, R., and Dou, D. Federated fingerprint learning with heterogeneous architectures. In *Proceedings of the 22nd IEEE International Conference on Data Mining (ICDM'22)*, pp. 31–40, Orlando, FL, November 28-December 1 2022.

Chen, C., Sun, F., Zhang, M., and Ding, B. Recommendation unlearning. In Laforest, F., Troncy, R., Simperl, E., Agarwal, D., Gionis, A., Herman, I., and Médini, L. (eds.), *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*, pp. 2768–2777. ACM, 2022.

Chen, M., Zhang, Z., Wang, T., Backes, M., Humbert, M., and Zhang, Y. When machine unlearning jeopardizes privacy. In Kim, Y., Kim, J., Vigna, G., and Shi, E. (eds.), *CCS '21: 2021 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, Republic of Korea, November 15 - 19, 2021*, pp. 896–911. ACM, 2021.

Chen, Y., Xiong, J., Xu, W., and Zuo, J. A novel online incremental and decremental learning algorithm based on variable support vector machine. *Clust. Comput.*, 22 (Supplement):7435–7445, 2019.

Cheng, H., Zhou, Y., and Yu, J. X. Clustering large attributed graphs: A balance between structural and attribute similarities. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(2):1–33, 2011.

Cheng, H., Zhou, Y., Huang, X., and Yu, J. X. Clustering large attributed information networks: An efficient incremental computing approach. *Data Mining and Knowledge Discovery (DMKD)*, 25(3):450–477, 2012.

Chourasia, R., Shah, N., and Shokri, R. Forget unlearning: Towards true data-deletion in machine learning. In *11th International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, Conference Track Proceedings*, 2023.

Chundawat, V. S., Tarun, A. K., Mandal, M., and Kankanhalli, M. S. Zero-shot machine unlearning. *CoRR*, abs/2201.05629, 2022.

Cohen, J. M., Rosenfeld, E., and Kolter, J. Z. Certified adversarial robustness via randomized smoothing. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pp. 1310–1320, 2019.

Dhar, S., Guo, J., Liu, J. J., Tripathi, S., Kurup, U., and Shah, M. A survey of on-device machine learning: An algorithms and learning theory perspective. *ACM Trans. Internet Things*, 2(3):15:1–15:49, 2021.

Fraboni, Y., Vidal, R., Kameni, L., and Lorenzi, M. Sequential informed federated unlearning: Efficient and provable client unlearning in federated optimization. *CoRR*, abs/2211.11656, 2022.

Fredrikson, M., Jha, S., and Ristenpart, T. Model inversion attacks that exploit confidence information and basic countermeasures. In Ray, I., Li, N., and Kruegel, C. (eds.), *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-16, 2015*, pp. 1322–1333. ACM, 2015.

Fu, S., He, F., and Tao, D. Knowledge removal in sampling-based bayesian inference. In *10th International Conference on Learning Representations, ICLR 2022, Online, April 25-29, 2022, Conference Track Proceedings*, 2022.

Gao, X., Ma, X., Wang, J., Sun, Y., Li, B., Ji, S., Cheng, P., and Chen, J. Verifi: Towards verifiable federated unlearning. *CoRR*, abs/2205.12709, 2022.

Garg, S., Goldwasser, S., and Vasudevan, P. N. Formalizing data deletion in the context of the right to be forgotten. In Canteaut, A. and Ishai, Y. (eds.), *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part II*, volume 12106 of *Lecture Notes in Computer Science*, pp. 373–402. Springer, 2020.

Ginart, A., Guan, M. Y., Valiant, G., and Zou, J. Making AI forget you: Data deletion in machine learning. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 3513–3526, 2019.

Golatkar, A., Achille, A., and Soatto, S. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 9301–9309. Computer Vision Foundation / IEEE, 2020a.

Golatkar, A., Achille, A., and Soatto, S. Forgetting outside the box: Scrubbing deep networks of information accessible from input-output observations. In Vedaldi, A., Bischof, H., Brox, T., and Frahm, J. (eds.), *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXIX*, volume 12374 of *Lecture Notes in Computer Science*, pp. 383–398. Springer, 2020b.

Golatkar, A., Achille, A., Ravichandran, A., Polito, M., and Soatto, S. Mixed-privacy forgetting in deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pp. 792–801. Computer Vision Foundation / IEEE, 2021.

Gong, J., Simeone, O., and Kang, J. Bayesian variational federated learning and unlearning in decentralized networks. In *22nd IEEE International Workshop on Signal Processing Advances in Wireless Communications, SPAWC 2021, Lucca, Italy, September 27-30, 2021*, pp. 216–220. IEEE, 2021a.

Gong, J., Simeone, O., Kassab, R., and Kang, J. Forget-svgd: Particle-based bayesian federated unlearning. *CoRR*, abs/2111.12056, 2021b.

Goswami, S., Pokhrel, A., Lee, K., Liu, L., Zhang, Q., and Zhou, Y. Graphmap: Scalable iterative graph processing using nosql. *The Journal of Supercomputing (TJSC)*, 76 (9):6619–6647, 2020.

Graves, L., Nagisetty, V., and Ganesh, V. Amnesiac machine learning. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 11516–11524. AAAI Press, 2021.

Guo, C., Goldstein, T., Hannun, A. Y., and van der Maaten, L. Certified data removal from machine learning models. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3832–3842. PMLR, 2020.

Guo, G., Yan, D., Yuan, L., Khalil, J., Long, C., Jiang, Z., and Zhou, Y. Maximal directed quasi-clique mining. In *Proceedings of the 38th IEEE International Conference on Data Engineering (ICDE'22)*, pp. 1900–1913, Kuala Lumpur, Malaysia, May 9-12 2022.

Gupta, V., Jung, C., Neel, S., Roth, A., Sharifi-Malvajerdi, S., and Waites, C. Adaptive machine unlearning. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 16319–16330, 2021.

Halimi, A., Kadhe, S., Rawat, A., and Baracaldo, N. Federated unlearning: How to efficiently erase a client in fl? *CoRR*, abs/2207.05521, 2022. doi: 10.48550/arXiv.2207.05521. URL https://doi.org/10.48550/arXiv.2207.05521.

He, K., Zhang, X., Ren, S., and Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pp. 1026–1034, 2015.

Izzo, Z., Smart, M. A., Chaudhuri, K., and Zou, J. Approximate data deletion from machine learning models. In Banerjee, A. and Fukumizu, K. (eds.), *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event*, volume 130 of *Proceedings of Machine Learning Research*, pp. 2008–2016. PMLR, 2021.

Jagielski, M., Thakkar, O., Tramèr, F., Ippolito, D., Lee, K., Carlini, N., Wallace, E., Song, S., Thakurta, A., Papernot, N., and Zhang, C. Measuring forgetting of memorized training examples. In *11th International Conference on*

Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, Conference Track Proceedings, 2023.

Jang, J., Yoon, D., Yang, S., Cha, S., Lee, M., Logeswaran, L., and Seo, M. Knowledge unlearning for mitigating privacy risks in language models. In *11th International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, Conference Track Proceedings*, 2023.

Jin, J., Ren, J., Zhou, Y., Lv, L., Liu, J., and Dou, D. Accelerated federated learning with decoupled adaptive optimization. In *Proceedings of the 39th International Conference on Machine Learning (ICML'22)*, pp. 10298–10322, Baltimore, MD, July 17-23 2022a.

Jin, J., Zhang, Z., Zhou, Y., and Wu, L. Input-agnostic certified group fairness via gaussian parameter smoothing. In *Proceedings of the 39th International Conference on Machine Learning (ICML'22)*, pp. 10340–10361, Baltimore, MD, July 17-23 2022b.

Jin, R., Li, D., Gao, J., Liu, Z., Chen, L., and Zhou, Y. Towards a better understanding of linear models for recommendation. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'21)*, pp. 776–785, Virtual Event, Singapore, August 14-18 2021.

Karasuyama, M. and Takeuchi, I. Multiple incremental decremental learning of support vector machines. In Bengio, Y., Schuurmans, D., Lafferty, J. D., Williams, C. K. I., and Culotta, A. (eds.), *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada*, pp. 907–915. Curran Associates, Inc., 2009.

Khan, M. E. and Swaroop, S. Knowledge-adaptation priors. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 19757–19770, 2021.

Krizhevsky, A. Learning multiple layers of features from tiny images. *Technical Report*, 2009.

Lee, K., Liu, L., Schwan, K., Pu, C., Zhang, Q., Zhou, Y., Yigitoglu, E., and Yuan, P. Scaling iterative graph computations with graphmap. In *Proceedings of the 27th IEEE international conference for High Performance Computing, Networking, Storage and Analysis (SC'15)*, pp. 57:1–57:12, Austin, TX, November 15-20 2015.

Lee, K., Liu, L., Ganti, R. L., Srivatsa, M., Zhang, Q., Zhou, Y., and Wang, Q. Lightwieight indexing and querying services for big spatial data. *IEEE Transactions on Services Computing (TSC)*, 12(3):343–355, 2019.

Legislature, C. S. California consumer privacy act of 2018. cal. civ. code §1798.100, 2018.

Li, G., Hu, Y., Zhang, M., Liu, J., Yin, Q., Peng, Y., and Dou, D. Fedhisyn: A hierarchical synchronous federated learning framework for resource and data heterogeneity. In *Int. Conf. on Parallel Processing (ICPP)*, pp. 1–11, 2022.

Li, Y., Wang, C., and Cheng, G. Online forgetting process for linear regression models. In Banerjee, A. and Fukumizu, K. (eds.), *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event*, volume 130 of *Proceedings of Machine Learning Research*, pp. 217–225. PMLR, 2021.

Liu, G., Ma, X., Yang, Y., Wang, C., and Liu, J. Federaser: Enabling efficient client-level data removal from federated learning models. In *29th IEEE/ACM International Symposium on Quality of Service, IWQOS 2021, Tokyo, Japan, June 25-28, 2021*, pp. 1–10. IEEE, 2021a.

Liu, J., Wu, Z., Yu, D., Ma, Y., Feng, D., Zhang, M., Wu, X., Yao, X., and Dou, D. Heterps: Distributed deep learning with reinforcement learning based scheduling in heterogeneous environments. *Future Generation Computer Systems*, 20213. To appear.

Liu, J., Huang, J., Zhou, Y., Li, X., Ji, S., Xiong, H., and Dou, D. From distributed machine learning to federated learning: A survey. *Knowledge and Information Systems (KAIS)*, 64(4):885–917, 2022a.

Liu, J., Jia, J., Ma, B., Zhou, C., Zhou, J., Zhou, Y., Dai, H., and Dou, D. Multi-job intelligent scheduling with cross-device federated learning. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 34(2):535–551, 2023.

Liu, Y., Ma, Z., Liu, X., Liu, J., Jiang, Z., Ma, J., Yu, P., and Ren, K. Learn to forget: Machine unlearning via neuron masking. *CoRR*, abs/2003.10933, 2020.

Liu, Y., Ma, Z., Yang, Y., Liu, X., Ma, J., and Ren, K. Revfrf: Enabling cross-domain random forest training with revocable federated learning. *IEEE Transactions on Dependable and Secure Computing*, 2021b.

Liu, Y., Fan, M., Chen, C., Liu, X., Ma, Z., Wang, L., and Ma, J. Backdoor defense with machine unlearning. In *41st IEEE Conference on Computer Communications, INFOCOM 2022, Virtual, May 2-5, 2022*. IEEE, 2022b.

Liu, Y., Xu, L., Yuan, X., Wang, C., and Li, B. The right to be forgotten in federated learning: An efficient realization with rapid retraining. In *41st IEEE Conference on Computer Communications, INFOCOM 2022, Virtual, May 2-5, 2022*. IEEE, 2022c.

Lu, X., Welleck, S., Jiang, L., Hessel, J., Qin, L., West, P., Ammanabrolu, P., and Choi, Y. Quark: Controllable text generation with reinforced unlearning. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022 (NeurIPS'22)*, New Orleans, LA, November 28-December 9 2022.

Mahadevan, A. and Mathioudakis, M. Certifiable machine unlearning for linear models. *CoRR*, abs/2106.15093, 2021.

Marchant, N. G., Rubinstein, B. I. P., and Alfeld, S. Hard to forget: Poisoning attacks on certified machine unlearning. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence, (AAAI'22), February 22-March 1, 2022, Vancouver, Canada*, 2022.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In Singh, A. and Zhu, X. J. (eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282. PMLR, 2017.

Neel, S., Roth, A., and Sharifi-Malvajerdi, S. Descent-to-delete: Gradient-based methods for machine unlearning. In Feldman, V., Ligett, K., and Sabato, S. (eds.), *Algorithmic Learning Theory, 16-19 March 2021, Virtual Conference, Worldwide*, volume 132 of *Proceedings of Machine Learning Research*, pp. 931–962. PMLR, 2021.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning, 2011*, 2011.

Nguyen, Q. P., Low, B. K. H., and Jaillet, P. Variational bayesian unlearning. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Nguyen, Q. P., Oikawa, R., Divakaran, D. M., Chan, M. C., and Low, B. K. H. Markov chain monte carlo-based machine unlearning: Unlearning what needs to be forgotten. In *Proceedings of the 2022 ACM Asia Conference on Computer and Communications Security (ASIA CCS '22), May 30-June 3, 2022, Nagasaki, Japan*, 2022.

of the EU, C. Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation), 2016.

Palanisamy, B., Liu, L., Zhou, Y., and Wang, Q. Privacy-preserving publishing of multilevel utility-controlled graph datasets. *ACM Transactions on Internet Technology (TOIT)*, 18(2):24:1–24:21, 2018.

Pan, C., Sima, J., Prakash, S., Rana, V., and Milenkovic, O. Machine unlearning of federated clusters. *CoRR*, abs/2210.16424, 2022.

Pawelczyk, M., Leemann, T., Biega, A., and Kasneci, G. On the trade-off between actionable explanations and the right to be forgotten. In *11th International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, Conference Track Proceedings*, 2023.

Ren, J., Zhou, Y., Jin, R., Zhang, Z., Dou, D., and Wang, P. Dual adversarial learning based network alignment. In *Proceedings of the 19th IEEE International Conference on Data Mining (ICDM'19)*, pp. 1288–1293, Beijing, China, November 8-11 2019.

Ren, J., Zhang, Z., Jin, J., Zhao, X., Wu, S., Zhou, Y., Shen, Y., Che, T., Jin, R., and Dou, D. Integrated defense for resilient graph matching. In *Proceedings of the 38th International Conference on Machine Learning (ICML'21)*, pp. 8982–8997, Virtual Event, July 18-24 2021.

Riesz, F. and Sz.-Nagy, B. *Functional Analysis*. Dover Publications, 1955.

Schelter, S. "amnesia" - machine learning models that can forget user data very fast. In *10th Conference on Innovative Data Systems Research, CIDR 2020, Amsterdam, The Netherlands, January 12-15, 2020, Online Proceedings*. www.cidrdb.org, 2020.

Schelter, S., Grafberger, S., and Dunning, T. Hedgecut: Maintaining randomised trees for low-latency machine unlearning. In Li, G., Li, Z., Idreos, S., and Srivastava, D. (eds.), *SIGMOD '21: International Conference on Management of Data, Virtual Event, China, June 20-25, 2021*, pp. 1545–1557. ACM, 2021.

Sekhari, A., Acharya, J., Kamath, G., and Suresh, A. T. Remember what you want to forget: Algorithms for machine unlearning. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 18075–18086, 2021.

Setlur, A., Eysenbach, B., Smith, V., and Levine, S. Adversarial unlearning: Reducing confidence along adversarial directions. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022 (NeurIPS'22)*, New Orleans, LA, November 28-December 9 2022.

Shibata, T., Irie, G., Ikami, D., and Mitsuzumi, Y. Learning with selective forgetting. In Zhou, Z. (ed.), *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pp. 989–996. ijcai.org, 2021.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pp. 3–18. IEEE Computer Society, 2017.

Su, Z., Liu, L., Li, M., Fan, X., and Zhou, Y. Servicetrust: Trust management in service provision networks. In *Proceedings of the 10th IEEE International Conference on Services Computing (SCC'13)*, pp. 272–279, Santa Clara, CA, June 27-July 2 2013.

Su, Z., Liu, L., Li, M., Fan, X., and Zhou, Y. Reliable and resilient trust management in distributed service provision networks. *ACM Transactions on the Web (TWEB)*, 9(3): 1–37, 2015.

Suriyakumar, V. M. and Wilson, A. C. Algorithms that approximate data removal: New results and limitations. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022 (NeurIPS'22)*, New Orleans, LA, November 28-December 9 2022.

Thudi, A., Deza, G., Chandrasekaran, V., and Papernot, N. Unrolling SGD: understanding factors influencing machine unlearning. *CoRR*, abs/2109.13398, 2021a.

Thudi, A., Jia, H., Shumailov, I., and Papernot, N. On the necessity of auditable algorithmic definitions for machine unlearning. *CoRR*, abs/2110.11891, 2021b.

Ullah, E., Mai, T., Rao, A., Rossi, R. A., and Arora, R. Machine unlearning via algorithmic stability. In Belkin, M. and Kpotufe, S. (eds.), *Conference on Learning Theory, COLT 2021, 15-19 August 2021, Boulder, Colorado, USA*, volume 134 of *Proceedings of Machine Learning Research*, pp. 4126–4142. PMLR, 2021.

Veale, M., Binns, R., and Edwards, L. Algorithms that remember: Model inversion attacks and data protection law. *CoRR*, abs/1807.04644, 2018.

Wang, J., Guo, S., Xie, X., and Qi, H. Protect privacy from gradient leakage attack in federated learning. In *IEEE INFOCOM 2022 - IEEE Conference on Computer Communications, London, United Kingdom, May 2-5, 2022*, pp. 580–589. IEEE, 2022a.

Wang, J., Guo, S., Xie, X., and Qi, H. Federated unlearning via class-discriminative pruning. In Laforest, F., Troncy, R., Simperl, E., Agarwal, D., Gionis, A., Herman, I., and Médini, L. (eds.), *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*, pp. 622–632. ACM, 2022b.

Warnecke, A., Pirch, L., Wressnegger, C., and Rieck, K. Get a model! model hijacking attack against machine learning models. In *30th Annual Network and Distributed System Security Symposium, NDSS 2022, San Diego, California, USA, February 28- March 4, 2023*. The Internet Society, 2023.

Wu, C., Zhu, S., and Mitra, P. Federated unlearning with knowledge distillation. *CoRR*, abs/2201.09441, 2022a.

Wu, G., Hashemi, M., and Srinivasa, C. PUMA: performance unchanged model augmentation for training data removal. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence, (AAAI'22), February 22-March 1, 2022, Vancouver, Canada*, 2022b.

Wu, L., Guo, S., Wang, J., Hong, Z., Zhang, J., and Ding, Y. Federated unlearning: Guarantee the right of clients to forget. *IEEE Netw.*, 36(5):129–135, 2022c.

Wu, S., Li, Y., Zhang, D., Zhou, Y., and Wu, Z. Diverse and informative dialogue generation with context-specific commonsense knowledge awareness. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL'20)*, pp. 5811–5820, Online, July 5-10 2020a.

Wu, S., Li, Y., Zhang, D., Zhou, Y., and Wu, Z. Topicka: Generating commonsense knowledge-aware dialogue responses towards the recommended topic fact. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI'20)*, pp. 3766–3772, Online, January 7-15 2021a.

Wu, S., Wang, M., Zhang, D., Zhou, Y., Li, Y., and Wu, Z. Knowledge-aware dialogue generation via hierarchical infobox accessing and infobox-dialogue interaction graph network. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI'21)*, pp. 3964–3970, Virtual Event / Montreal, Canada, August 19-27 2021b.

Wu, Y., Dobriban, E., and Davidson, S. B. Deltagrad: Rapid retraining of machine learning models. In *Proceedings* *of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10355–10366. PMLR, 2020b.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017.

Yan, D., Qu, W., Guo, G., Wang, X., and Zhou, Y. Prefixfpm: A parallel framework for general-purpose mining of frequent and closed patterns. *The VLDB Journal (VLDBJ)*, 31(2):253–286, 2022a.

Yan, D., Zhou, Y., and Guo, G. Think-like-a-task programming model. In Albert Zomaya, J. T. and Sakr, S. (eds.), *Encyclopedia of Big Data Technologies*. Springer, 2022b.

Yan, D., Zhou, Y., Guo, G., and Liu, H. Parallel graph processing. In Albert Zomaya, J. T. and Sakr, S. (eds.), *Encyclopedia of Big Data Technologies*. Springer, 2022c.

Yuan, W., Yin, H., Wu, F., Zhang, S., He, T., and Wang, H. Federated unlearning for on-device recommendation. *CoRR*, abs/2210.10958, 2022.

Zeng, Y., Chen, S., Park, W., Mao, Z., Jin, M., and Jia, R. Adversarial unlearning of backdoors via implicit hypergradient. In *10th International Conference on Learning Representations, ICLR 2022, Online, April 25-29, 2022, Conference Track Proceedings*, 2022.

Zhang, G., Zhou, Y., Wu, S., Zhang, Z., and Dou, D. Cross-lingual entity alignment with adversarial kernel embedding and adversarial knowledge translation. *CoRR*, abs/2104.07837, 2021a.

Zhang, H., Liu, J., Jia, J., Zhou, Y., Dai, H., and Dou, D. Fedduap: Federated learning with dynamic update and adaptive pruning using shared data on the server. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI'22)*, pp. 2776–2782, Messe Wien, Vienna, Austria, July 23-29 2022a.

Zhang, Z., Zhang, Z., Zhou, Y., Shen, Y., Jin, R., and Dou, D. Adversarial attacks on deep graph matching. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020 (NeurIPS'20)*, Virtual, December 6-12 2020.

Zhang, Z., Jin, J., Zhang, Z., Zhou, Y., Zhao, X., Ren, J., Liu, J., Wu, L., Jin, R., and Dou, D. Validating the lottery ticket hypothesis with inertial manifold theory. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021 (NeurIPS'21)*, pp. 30196–30210, Virtual, December 6-14 2021b.

Zhang, Z., Zhang, Z., Zhou, Y., Wu, L., Wu, S., Han, X., Dou, D., Che, T., and Yan, D. Adversarial attack against cross-lingual knowledge graph alignment. In *Proceedings of the 26th Conference on Empirical Methods in Natural Language Processing (EMNLP'21)*, pp. 5320–5337, Virtual Event / Punta Cana, Dominican Republic, November 7-11 2021c.

Zhang, Z., Zhou, Y., Zhao, X., Che, T., and Lyu, L. Prompt certified machine unlearning with randomized gradient smoothing and quantization. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022 (NeurIPS'22)*, New Orleans, LA, November 28-December 9 2022b.

Zhao, X., Zhang, Z., Zhang, Z., Wu, L., Jin, J., Zhou, Y., Jin, R., Dou, D., and Yan, D. Expressive 1-lipschitz neural networks for robust multiple graph learning against adversarial attacks. In *Proceedings of the 38th International Conference on Machine Learning (ICML'21)*, pp. 12719–12735, Virtual Event, July 18-24 2021.

Zhou, C., Liu, J., Jia, J., Zhou, J., Zhou, Y., Dai, H., and Dou, D. Efficient device scheduling with multi-job federated learning. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI'22)*, pp. 9971–9979, Vancouver, Canada, February 22-March 1 2022a.

Zhou, Y. *Innovative Mining, Processing, and Application of Big Graphs*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, USA, 2017.

Zhou, Y. and Liu, L. Clustering analysis in large graphs with rich attributes. In Holmes, D. E. and Jain, L. C. (eds.), *Data Mining: Foundations and Intelligent Paradigms: Volume 1: Clustering, Association and Classification*. Springer, 2012.

Zhou, Y. and Liu, L. Social influence based clustering of heterogeneous information networks. In *Proceedings of the 19th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'13)*, pp. 338–346, Chicago, IL, August 11-14 2013.

Zhou, Y. and Liu, L. Activity-edge centric multi-label classification for mining heterogeneous information networks. In *Proceedings of the 20th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'14)*, pp. 1276–1285, New York, NY, August 24-27 2014.

Zhou, Y. and Liu, L. Social influence based clustering and optimization over heterogeneous information networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 10(1):1–53, 2015.

Zhou, Y. and Liu, L. Approximate deep network embedding for mining large-scale graphs. In *Proceedings of the 2019 IEEE International Conference on Cognitive Machine Intelligence (CogMI'19)*, pp. 53–60, Los Angeles, CA, December 12-14 2019.

Zhou, Y., Cheng, H., and Yu, J. X. Graph clustering based on structural/attribute similarities. *Proceedings of the VLDB Endowment (PVLDB)*, 2(1):718–729, 2009.

Zhou, Y., Cheng, H., and Yu, J. X. Clustering large attributed graphs: An efficient incremental approach. In *Proceedings of the 10th IEEE International Conference on Data Mining (ICDM'10)*, pp. 689–698, Sydney, Australia, December 14-17 2010.

Zhou, Y., Seshadri, S., Chiu, L., and Liu, L. Graphlens: Mining enterprise storage workloads using graph analytics. In *Proceedings of the 2014 IEEE International Congress on Big Data (BigData'14)*, pp. 1–8, Anchorage, AK, June 27-July 2 2014.

Zhou, Y., Liu, L., and Buttler, D. Integrating vertex-centric clustering with edge-centric clustering for meta path graph analysis. In *Proceedings of the 21st ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'15)*, pp. 1563–1572, Sydney, Australia, August 10-13 2015a.

Zhou, Y., Liu, L., Lee, K., Pu, C., and Zhang, Q. Fast iterative graph computation with resource aware graph parallel abstractions. In *Proceedings of the 24th ACM Symposium on High-Performance Parallel and Distributed Computing (HPDC'15)*, pp. 179–190, Portland, OR, June 15-19 2015b.

Zhou, Y., Liu, L., Lee, K., and Zhang, Q. Graphtwist: Fast iterative graph computation with two-tier optimizations. *Proceedings of the VLDB Endowment (PVLDB)*, 8(11): 1262–1273, 2015c.

Zhou, Y., Liu, L., Seshadri, S., and Chiu, L. Analyzing enterprise storage workloads with graph modeling and clustering. *IEEE Journal on Selected Areas in Communications (JSAC)*, 34(3):551–574, 2016.

Zhou, Y., Amimeur, A., Jiang, C., Dou, D., Jin, R., and Wang, P. Density-aware local siamese autoencoder network embedding with autoencoder graph clustering. In *Proceedings of the 2018 IEEE International Conference on Big Data (BigData'18)*, pp. 1162–1167, Seattle, WA, December 10-13 2018a.

Zhou, Y., Wu, S., Jiang, C., Zhang, Z., Dou, D., Jin, R., and Wang, P. Density-adaptive local edge representation learning with generative adversarial network multi-label edge classification. In *Proceedings of the 18th IEEE International Conference on Data Mining (ICDM'18)*, pp. 1464–1469, Singapore, November 17-20 2018b.

Zhou, Y., Jiang, C., Zhang, Z., Dou, D., Jin, R., and Wang, P. Integrating local vertex/edge embedding via deep matrix fusion and siamese multi-label classification. In *Proceedings of the 2019 IEEE International Conference on Big Data (BigData'19)*, pp. 1018–1027, Los Angeles, CA, December 9-12 2019a.

Zhou, Y., Ling Liu, Qi Zhang, K. L., and Palanisamy, B. Enhancing collaborative filtering with multi-label classification. In *Proceedings of the 2019 International Conference on Computational Data and Social Networks (CSoNet'19)*, pp. 323–338, Ho Chi Minh City, Vietnam, November 18-20 2019b.

Zhou, Y., Ren, J., Wu, S., Dou, D., Jin, R., Zhang, Z., and Wang, P. Semi-supervised classification-based local vertex ranking via dual generative adversarial nets. In *Proceedings of the 2019 IEEE International Conference on Big Data (BigData'19)*, pp. 1267–1273, Los Angeles, CA, December 9-12 2019c.

Zhou, Y., Liu, L., Lee, K., Palanisamy, B., and Zhang, Q. Improving collaborative filtering with social influence over heterogeneous information networks. *ACM Transactions on Internet Technology (TOIT)*, 20(4):36:1–36:29, 2020a.

Zhou, Y., Ren, J., Dou, D., Jin, R., Zheng, J., and Lee, K. Robust meta network embedding against adversarial attacks. In *Proceedings of the 20th IEEE International Conference on Data Mining (ICDM'20)*, pp. 1448–1453, Sorrento, Italy, November 17-20 2020b.

Zhou, Y., Ren, J., Jin, R., Zhang, Z., Dou, D., and Yan, D. Unsupervised multiple network alignment with multinominal gan and variational inference. In *Proceedings of the 2020 IEEE International Conference on Big Data (BigData'20)*, pp. 868–877, Atlanta, GA, December 10-13 2020c.

Zhou, Y., Zhang, Z., Wu, S., Sheng, V., Han, X., Zhang, Z., and Jin, R. Robust network alignment via attack signal scaling and adversarial perturbation elimination. In *Proceedings of the 30th Web Conference (WWW'21)*, pp. 3884–3895, Virtual Event / Ljubljana, Slovenia, April 19-23 2021.

Zhou, Y., Ren, J., Jin, R., Zhang, Z., Zheng, J., Jiang, Z., Yan, D., and Dou, D. Unsupervised adversarial network alignment with reinforcement learning. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 16(3): 50:1–50:29, 2022b.

A. Appendix

A.1. Theoretical Proof

Theorem 2. *O is Fréchet differentiable and $O(\mathbb{R}^3) \subset L^p$ is a smooth manifold.*

Proof. *According to Definition 4, in order to prove that an operator o is Fréchet differentiable, we first need to find a bounded linear operator $L(x)$ and then demonstrate that the limit condition in the definition is satisfied, i.e., $\lim\limits_{\|\Delta\|_X \to 0} \frac{\|o(x+h)-o(x)-L(x)\Delta\|_Y}{\|\Delta\|_X} \to 0$.*

Without loss of generality, we prove this theorem when $\sigma = 1$. As $\mathcal{N}(0, \sigma^2 I) = \sigma^{-3}\mathcal{N}(0, I)$ in $\mathbb{R}^3$, it is easy to extend the proof process to other σ values by following the same strategy. For ease of presentation, we rewrite $O(q)$ as follows.

$$O(q)(\cdot) = \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) e^{-\frac{\|q-\varepsilon\|_2^2}{2}} d\varepsilon \tag{16}$$

A linear operator $L(q)(\epsilon) : \mathbb{R}^3 \to L^p$ defined as follows.

$$L(q)(\epsilon)(\cdot) = -\frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) e^{-\frac{\|q-\varepsilon\|_2^2}{2}} (q-\varepsilon)\epsilon d\varepsilon \tag{17}$$

where $L(q)$ is the bounded linear operator regarding the Fréchet derivative of $O(q)$.

We compute

$$\frac{d}{ds} e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} = -e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} (q + s\epsilon - \varepsilon)\epsilon \tag{18}$$

By integrating Eq.(18) from 0 to 1, we get

$$e^{-\frac{\|q+\epsilon-\varepsilon\|_2^2}{2}} - e^{-\frac{\|q-\varepsilon\|_2^2}{2}} = -\int_0^1 e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} (q + s\epsilon - \varepsilon)\epsilon ds \tag{19}$$

Next, we compute

$$\begin{aligned} &\|O(q+\epsilon)(\cdot) - O(q)(\cdot) - L(q)\epsilon(\cdot)\|_{L^p(\Omega)} \\ =&\left\| \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) \left(e^{-\frac{\|q+\epsilon-\varepsilon\|_2^2}{2}} - e^{-\frac{\|q-\varepsilon\|_2^2}{2}} + e^{-\frac{\|q-\varepsilon\|_2^2}{2}} (q-\varepsilon)\epsilon \right) d\varepsilon \right\|_{L^p(\Omega)} \end{aligned} \tag{20}$$

By combining Eq.(19) and Eq.(20), we obtain

$$\begin{aligned} &\|O(q+\epsilon)(\cdot) - O(q)(\cdot) - L(q)\epsilon(\cdot)\|_{L^p(\Omega)} \\ =&\left\| \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) \left(\int_0^1 e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} (q+s\epsilon-\varepsilon)\epsilon - e^{-\frac{\|q-\varepsilon\|_2^2}{2}} (q-\varepsilon)\epsilon ds \right) d\varepsilon \right\|_{L^p(\Omega)} \\ =&\left\| \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) \left(\int_0^1 (e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} - e^{-\frac{\|q-\varepsilon\|_2^2}{2}})(q-\varepsilon)\epsilon + e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} s\epsilon^2 ds \right) d\varepsilon \right\|_{L^p(\Omega)} \end{aligned} \tag{21}$$

In terms of the Fubini's theorem, we have

$$\begin{aligned} &\|O(q+\epsilon)(\cdot) - O(q)(\cdot) - L(q)\epsilon(\cdot)\|_{L^p(\Omega)} \\ =&\left\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \left(\int_{\mathbb{R}^3} P(\varepsilon)(\cdot)(e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} - e^{-\frac{\|q-\varepsilon\|_2^2}{2}})(q-\varepsilon)\epsilon + e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} s\epsilon^2 d\varepsilon \right) ds \right\|_{L^p(\Omega)} \end{aligned} \tag{22}$$

Notice that

$$\frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} \epsilon^2 d\varepsilon = \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} e^{-\frac{\|\varepsilon\|_2^2}{2}} \epsilon^2 d\varepsilon = \|\epsilon\|_2^2 \tag{23}$$

It is obvious that $|P(\varepsilon)(x)| \leq 1$ for any ε and x. Thus, we have

$$\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} s\epsilon^2 d\varepsilon ds \Big\|_{L^p(\Omega)} \leq \|1\|_{L^p(\Omega)} \|\epsilon\|_2^2 \tag{24}$$

Similar to Eq.(19), we derive

$$e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} - e^{-\frac{\|q-\varepsilon\|_2^2}{2}} = -\int_0^1 e^{-\frac{\|q+st\epsilon-\varepsilon\|_2^2}{2}} (q+s\epsilon-\varepsilon)\epsilon dt \tag{25}$$

In addition, we calculate

$$\begin{aligned} &\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_{\mathbb{R}^3} P(\varepsilon)(\cdot)(e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} - e^{-\frac{\|q-\varepsilon\|_2^2}{2}})(q-\varepsilon)\epsilon d\varepsilon ds \Big\|_{L^p(\Omega)} \\ =&\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_0^1 \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) e^{-\frac{\|q+st\epsilon-\varepsilon\|_2^2}{2}} (q+s\epsilon-\varepsilon)\epsilon(q-\varepsilon)\epsilon d\varepsilon ds dt \Big\|_{L^p(\Omega)} \\ \leq&\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_0^1 \int_{\mathbb{R}^3} e^{-\frac{\|q+st\epsilon-\varepsilon\|_2^2}{2}} \|q+s\epsilon-\varepsilon\|_2 \|q-\varepsilon\|_2 \|\epsilon\|^2 d\varepsilon ds dt \Big\|_{L^p(\Omega)} \end{aligned} \tag{26}$$

According to the dominated convergence theorem, when $\epsilon \to 0$, we have

$$\int_{\mathbb{R}^3} e^{-\frac{\|q+st\epsilon-\varepsilon\|_2^2}{2}} \|q+s\epsilon-\varepsilon\|_2 \|q-\varepsilon\|_2 \|\epsilon\|^2 d\varepsilon \to \int_{\mathbb{R}^3} e^{-\frac{\|q-\varepsilon\|_2^2}{2}} \|q-\varepsilon\|_2^2 d\varepsilon \tag{27}$$

By plugging Eq.(27) into Eq.(26), when $\epsilon \to 0$, we derive

$$\begin{aligned} &\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_0^1 \int_{\mathbb{R}^3} P(\varepsilon)(\cdot)(e^{-\frac{\|q+s\epsilon-\varepsilon\|_2^2}{2}} - e^{-\frac{\|q-\varepsilon\|_2^2}{2}})(q-\varepsilon)\epsilon d\varepsilon ds dt \Big\|_{L^p(\Omega)} \\ \leq&\Big\| \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} e^{-\frac{\|q'\|_2^2}{2}} \|q'\|_2^2 dq' \Big\|_{L^p(\Omega)} \|\epsilon\|^2 \end{aligned} \tag{28}$$

By integrating Eq.(24) and Eq.(28), we get

$$\|O(q+\epsilon)(\cdot) - O(q)(\cdot) - L(q)\epsilon(\cdot)\|_{L^p(\Omega)} = O(\epsilon^2) \tag{29}$$

When $\epsilon \to 0$, $\|O(q+\epsilon)(\cdot) - O(q)(\cdot) - L(q)\epsilon(\cdot)\|_{L^p(\Omega)} \to 0$ too. This implies that the limit condition in Definition 4 is satisfied and thus $L(q)$ is Fréchet derivative of O. By following the similar strategy, we can derive that $O(q)$ is infinitely Fréchet differentiable. Notice that $O(q)$ can be treated as a global chart for the manifold $O(\mathbb{R}^3)$. Thus, $O(\mathbb{R}^3)$ is a smooth manifold.

Theorem 3. *Let $\bar{G}_k \in \mathbb{R}^T$ be the gradient of the local ML model on edge device k, Q be the gradient quantizatier, C be the Lipschitz constant of Q, $q_k = Q(\bar{G}_k)$, and O be the smooth Nemytskii operator. If $\|\bar{G}_k - \bar{G}_l\|_2 \leq d$ for any $q_k, q_l \in \mathbb{R}^3$, then we have*

$$\begin{aligned} &|\Omega|^{-1} \|O(q_k)(x) - O(q_l)(x)\|_{L^p(\Omega)} \leq \frac{Cd}{\sqrt{2\pi}\sigma} \text{ if } 1 \leq p < \infty, \\ &\|O(q_k)(x) - O(q_l)(x)\|_{L^p(\Omega)} \leq \frac{Cd}{\sqrt{2\pi}\sigma} \text{ if } p = \infty \end{aligned} \tag{30}$$

for any $\Omega \subset \mathbb{R}^n$.

Proof. Again, without loss of generality, we prove this theorem when $\sigma = 1$. By following the same strategy, it is easy to extend the proof process to other σ values. For ease of presentation, we rewrite $O(q)$ as follows.

$$O(q)(\cdot) = \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) e^{-\frac{\|q-\varepsilon\|_2^2}{2}} d\varepsilon \tag{31}$$

We compute

$$\begin{aligned} &\|O(q_k)(\cdot) - O(q_l)(\cdot)\|_{L^p(\Omega)} \\ =&\Big\| \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) \left(e^{-\frac{\|q_k-\varepsilon\|_2^2}{2}} - e^{-\frac{\|q_l-\varepsilon\|_2^2}{2}} \right) d\varepsilon \Big\|_{L^p(\Omega)} \end{aligned} \tag{32}$$

Notice that

$$\frac{d}{ds} e^{-\frac{\|q_l-\varepsilon+s(q_k-q_l)\|_2^2}{2}} = -e^{-\frac{\|q_l-\varepsilon+s(q_k-q_l)\|_2^2}{2}} (q_l - \varepsilon + s(q_k - q_l))(q_k - q_l) \tag{33}$$

By integrating Eq.(33) from 0 to 1, we get

$$e^{-\frac{\|q_k-\varepsilon\|_2^2}{2}} - e^{-\frac{\|q_l-\varepsilon\|_2^2}{2}} = -\int_0^1 e^{-\frac{\|q_l-\varepsilon+s(q_k-q_l)\|_2^2}{2}} (q_l - \varepsilon + s(q_k - q_l))(q_k - q_l) ds \tag{34}$$

By combining Eq.(32) and Eq.(34), we obtain

$$\begin{aligned} &\|O(q_k)(\cdot) - O(q_l)(\cdot)\|_{L^p(\Omega)} \\ =&\Big\| \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) \left(\int_0^1 e^{-\frac{\|q_l-\varepsilon+s(q_k-q_l)\|_2^2}{2}} (q_l - \varepsilon + s(q_k - q_l))(q_k - q_l) ds \right) d\varepsilon \Big\|_{L^p(\Omega)} \\ =&\Big\| \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} \int_0^1 P(\varepsilon)(\cdot) e^{-\frac{\|q_l-\varepsilon+s(q_k-q_l)\|_2^2}{2}} (q_l - \varepsilon + s(q_k - q_l))(q_k - q_l) ds d\varepsilon \Big\|_{L^p(\Omega)} \end{aligned} \tag{35}$$

In terms of the Fubini's theorem, we have

$$\begin{aligned} &\|O(q_k)(\cdot) - O(q_l)(\cdot)\|_{L^p(\Omega)} \\ =&\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) e^{-\frac{\|q_l-\varepsilon+s(q_k-q_l)\|_2^2}{2}} (q_l - \varepsilon + s(q_k - q_l))(q_k - q_l) d\varepsilon ds \Big\|_{L^p(\Omega)} \end{aligned} \tag{36}$$

For ease of presentation, let $U = q_l + s(q_k - q_l)$ and $V = q_k - q_l$. We rewrite Eq.(36) as follows.

$$\begin{aligned} &\|O(q_k)(\cdot) - O(q_l)(\cdot)\|_{L^p(\Omega)} \\ =&\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) e^{\frac{-\|U-\varepsilon\|_2^2}{2}} (U - \varepsilon) V d\varepsilon ds \Big\|_{L^p(\Omega)} \\ =&\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) e^{\frac{-\|U-\varepsilon\|_2^2}{2}} (U - \varepsilon) \frac{V}{\|V\|_2} d\varepsilon ds \Big\|_{L^p(\Omega)} \|V\|_2 \end{aligned} \tag{37}$$

Notice that $0 \le P(q)(x) \le 1$ for any q and x and $\frac{V}{\|V\|_2}$ is a unit vector. By utilizing standard Gaussian integrals, for any unit vector u, we have

$$\begin{aligned} &\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) e^{\frac{-\|U-\varepsilon\|_2^2}{2}} (U - \varepsilon) u d\varepsilon ds \Big\|_{L^p(\Omega)} \\ \le&\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_{\mathbb{R}^3} e^{\frac{-\|U-\varepsilon\|_2^2}{2}} |(U - \varepsilon) u| d\varepsilon ds \Big\|_{L^p(\Omega)} \\ =&\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_{\mathbb{R}^3} e^{\frac{-\|\varepsilon\|_2^2}{2}} |\varepsilon u| d\varepsilon ds \Big\|_{L^p(\Omega)} \end{aligned} \tag{38}$$

We rotate the coordinate system such that the unit vector u coincides with any coordinate axis. Let u, v_1, v_2 be the standard basis of the rotated coordinate system. In other words, for any $\varepsilon \in \mathbb{R}^3$, we decompose ε into

$$\varepsilon = (\varepsilon u)u + \sum_{l=1}^{2} (\varepsilon v_2) v_2 \tag{39}$$

As the Jacobian of rotation is 1, by employing standard Gaussian integrals, we obtain

$$\begin{aligned} \int_{\mathbb{R}^3} e^{\frac{-\|\varepsilon\|_2^2}{2}} |\varepsilon u| d\varepsilon &= \int_{\mathbb{R}^3} e^{\frac{-\sum_{l=1}^{2} |\varepsilon v_l|_2^2}{2}} e^{\frac{-|\varepsilon u|_2^2}{2}} |\varepsilon u| d\varepsilon \\ &= \int_{\mathbb{R}^3} e^{\frac{-\sum_{l=1}^{2} \tau_l^2}{2}} e^{\frac{-|\tau_m|^2}{2}} d\tau_1 \cdots d\tau_m = 2(2\pi)^{\frac{2}{2}} \end{aligned} \tag{40}$$

Accordingly, we derive

$$\Big\| \frac{1}{(2\pi)^{3/2}} \int_0^1 \int_{\mathbb{R}^3} P(\varepsilon)(\cdot) e^{\frac{-\|U-\varepsilon\|_2^2}{2}} (U-\varepsilon) u d\varepsilon ds \Big\|_{L^p(\Omega)} = \| \frac{1}{(2\pi)^{1/2}} \|_{L^p(\Omega)} \tag{41}$$

Thus, when $1 \leq p < \infty$, we obtain

$$\begin{aligned} \|O(q_k)(\cdot) - O(q_l)(\cdot)\|_{L^p(\Omega)} &\leq \frac{1}{(2\pi)^{1/2}} |\Omega| \|q_k - q_l\|_2 = \frac{1}{\sqrt{2\pi}} |\Omega| \|Q(\bar{G}_k) - Q(\bar{G}_l)\|_2 \\ &\leq \frac{C}{\sqrt{2\pi}} |\Omega| \|\bar{G}_k - \bar{G}_l\|_2 \leq \frac{Cd}{\sqrt{2\pi}} |\Omega| \end{aligned} \tag{42}$$

When $p = \infty$, we get

$$\begin{aligned} \|O(q_k)(\cdot) - O(q_l)(\cdot)\|_{L^p(\Omega)} &\leq \frac{1}{(2\pi)^{1/2}} \|q_k - q_l\|_2 = \frac{1}{\sqrt{2\pi}} \|Q(\bar{G}_k) - Q(\bar{G}_l)\|_2 \\ &\leq \frac{C}{\sqrt{2\pi}} \|\bar{G}_k - \bar{G}_l\|_2 \leq \frac{Cd}{\sqrt{2\pi}} \end{aligned} \tag{43}$$

$$\|O(q_k)(\cdot) - O(q_l)(\cdot)\|_{L^\infty(\Omega)} \leq \frac{1}{(2\pi)^{1/2}} \|q_k - q_l\|_2 \tag{44}$$

Therefore, the proof is concluded.

Theorem 4. *Let $\bar{G}_k \in \mathbb{R}^T$ be the gradient of the local ML model on edge device k, Q be the gradient quantizatier, C be the Lipschitz constant of Q, $q_k = Q(\bar{G}_k)$, O be the smooth Nemytskii operator, and $O(q_k)(x)$ be the smooth local model on device k. Let $\bar{G} = \frac{\bar{G}_1 + \cdots + \bar{G}_K}{K}$ be the gradient of the global ML model on the server by averaging the gradients of all the local ML models, $q = Q(\bar{G})$, and $O(q)(x)$ be the smooth global model on the server. If $\max\limits_{1 \leq k, l \leq K} \|\bar{G}_k - \bar{G}_l\|_2 = d$, then we have*

$$\begin{aligned} &|\Omega|^{-1} \|O(q)(x) - O(q_k)(x)\|_{L^p(\Omega)} \leq \frac{(K-1)Cd}{\sqrt{2\pi} K\sigma} \text{ if } 1 \leq p < \infty, \\ &\|O(q)(x) - O(q_k)(x)\|_{L^p(\Omega)} \leq \frac{(K-1)Cd}{\sqrt{2\pi} K\sigma} \text{ if } p = \infty \end{aligned} \tag{45}$$

for any $\Omega \subset \mathbb{R}^n$.

Proof. Similar to the conclusion of Theorem 3, we have

$$\begin{aligned} &|\Omega|^{-1} \|O(q)(x) - O(q_k)(x)\|_{L^p(\Omega)} \leq \frac{C}{\sqrt{2\pi}\sigma} \|\bar{G} - \bar{G}_k\|_2 \text{ if } 1 \leq p < \infty, \\ &\|O(q)(x) - O(q_k)(x)\|_{L^p(\Omega)} \leq \frac{C}{\sqrt{2\pi}\sigma} \|\bar{G} - \bar{G}_k\|_2 \text{ if } p = \infty \end{aligned} \tag{46}$$

According to the definition of $\bar{G} = \frac{\bar{G}_1 + \cdots + \bar{G}_K}{K}$*, for any* $k \in \{1, \cdots K\}$, $\bar{G} - \bar{G}_k = \frac{\sum_{l=1, l\neq k}^{K} \bar{G} - \bar{G}_k}{K}$*.*

Thus, based on the norm's triangle inequality, we obtain

$$\|\bar{G} - \bar{G}_k\|_2 = \frac{\| \sum_{l=1, l\neq k}^{K} \bar{G} - \bar{G}_k\|_2}{K} \leq \frac{\sum_{l=1, l\neq k}^{K} \|\bar{G} - \bar{G}_k\|_2}{K} \leq \frac{(K-1)d}{K} \tag{47}$$

Therefore, the proof is concluded.

Theorem 5. *Let* $\bar{G}$ *be the gradient of the global ML model on the server,* Q *be the gradient quantizatier,* $q = Q(\bar{G})$*, and* O *and* P *be the smooth and non-smooth Nemytskii operators respectively.* $O(q)(x) \rightarrow P(q)(x)$ *for any data sample* $x \in \mathbb{R}^n$ *if* $\sigma \rightarrow 0$*.*

Proof. For ease of presentation, let $\varepsilon = \sigma\tau$*. We rewrite* $O(q)$ *as follows.*

$$O(q)(x) = \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} P(q + \sigma\tau)(x) e^{-\frac{\|\tau|_2^2}{2}} d\tau \tag{48}$$

It is obvious that $|P(q + \sigma\tau)(x) e^{-\frac{\|\tau|_2^2}{2}}| \leq e^{-\frac{\|\tau|_2^2}{2}}$ *and* $e^{-\frac{\|\tau|_2^2}{2}}$ *is integrable. According to the continuity of* $P(q + \sigma\tau)(x)$*, we obtain* $P(q + \sigma\tau)(x) \rightarrow P(q)(x)$ *if* $\sigma \rightarrow 0$*. In terms of the dominated convergence theorem, we have the following conclusion.*

$$\frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} P(q + \sigma\tau)(x) e^{-\frac{\|\tau|_2^2}{2}} d\tau \rightarrow \frac{1}{(2\pi)^{3/2}} \int_{\mathbb{R}^3} P(q)(x) e^{-\frac{\|\tau|_2^2}{2}} d\tau \text{ if } \sigma \rightarrow 0 \tag{49}$$

Therefore, the proof is concluded.

A.2. Additional Experiments

Machine unlearning performance and running time with varying ratios of data removal. Tables 5-15 exhibit the classification accuracy, errors, training time, and unlearning time obtained by twelve federated machine unlearning approaches by varying the ratio of unlearning request / data removal between 2% and 20% on three datasets of Fashion-MNIST, CIFAR-10, and SVHN respectively. Similar trends are observed for the comparison of federated machine unlearning effectiveness and efficiency in these figures: our FFMU method achieves the smallest absolute performance difference with the Retrain model, regarding $Accuracy$ (¡3%), $Error^t$ (¡3%), $Error^r$ (¡3%), and $Error^f$ (¡2%) on three datasets respectively. Our FFMU method achieves better efficiency than most baseline methods in most experiments. Our FFMU method performs one-time operation of simultaneous training and unlearning when addressing a series of federated machine unlearning requests. The above experiment results demonstrate that FFMU is effective as well as efficient for addressing the federated machine unlearning problem. This advantage is very important for entitling data owners to the right to have their private data removed from trained complex models at their requests in a timely and cost-efficient manner in privacy-critical applications that usually require near-zero tolerance of data leaking.

Table 5: Performance with 5% data removal and CNN on Fashion-MNIST

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	84.49	15.51	14.71	16.10	**193**	920	1,113
Knowledge Distillation	81.61	18.39	16.98	21.27	212	1,442	1,654
Rapid Retraining	81.70	18.30	17.99	14.34	230	954	1,184
MacForget	80.52	19.48	17.88	20.37	227	1,025	1,252
FedEraser	82.07	17.93	17.11	20.83	361	1,766	2,127
VeriFi	82.40	17.60	16.76	19.15	212	1,036	1,248
Class-Discriminative Pruning	80.54	19.46	18.27	19.30	215	871	**1,032**
UN	80.35	19.65	19.07	23.19	205	1,053	1,258
RCAD	80.92	19.08	18.77	17.66	331	3,192	3,523
IJ	82.76	17.24	16.65	14.48	232	964	1,196
Noisy-GD	83.00	17.00	15.53	14.36	229	959	1,188
FFMU	**84.55**	**15.45**	**14.87**	**15.37**	1,288	**0**	1,288

Table 6: Performance with 8% data removal and CNN on Fashion-MNIST

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	84.75	15.25	14.45	15.58	**198**	1,416	1,614
Knowledge Distillation	81.65	18.35	17.18	19.24	218	2,201	2,419
Rapid Retraining	82.78	17.22	15.57	17.71	201	1,487	1,688
MacForget	82.09	17.91	17.03	20.09	222	1,592	1,814
FedEraser	83.78	16.22	15.66	16.84	358	2,828	3,186
VeriFi	82.10	17.90	16.85	16.47	205	1,629	1,834
Class-Discriminative Pruning	81.66	18.34	17.42	19.04	240	1,353	1,593
UN	80.38	19.62	18.82	18.00	210	1,618	1,828
RCAD	78.86	21.14	20.50	20.92	341	5,047	5,388
IJ	81.56	18.44	17.95	17.08	228	1,536	1,764
Noisy-GD	80.35	19.65	18.61	19.40	206	1,547	1,753
FFMU	**84.55**	**15.45**	**14.86**	**15.38**	1,288	**0**	**1,288**

Table 7: Performance with 15% data removal and CNN on Fashion-MNIST

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	84.66	15.34	14.84	14.84	**188**	2,378	2,566
Knowledge Distillation	82.35	17.65	16.20	18.67	220	4,098	4,318
Rapid Retraining	83.23	16.77	15.32	18.66	191	2,790	2,981
MacForget	81.83	18.17	16.46	18.26	237	2,722	2,959
FedEraser	83.77	16.23	15.94	15.95	364	5,349	5,713
VeriFi	82.59	17.41	15.94	16.87	232	2,989	3,221
Class-Discriminative Pruning	82.63	17.37	16.23	16.69	236	2,553	2,789
UN	82.06	17.94	16.69	16.67	217	2,963	3,180
RCAD	80.31	19.69	18.65	19.14	344	5,312	5,656
IJ	82.26	17.74	16.99	18.03	228	2,742	2,970
Noisy-GD	82.38	17.62	16.68	16.10	222	2,833	3,055
FFMU	**84.55**	**15.45**	**14.84**	**14.76**	1,288	**0**	**1,288**

Table 8: Performance with 5% data removal and LeNet on CIFAR-10

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	54.77	45.23	41.56	48.04	169	846	1,015
Knowledge Distillation	52.60	47.40	46.35	50.31	144	991	1,135
Rapid Retraining	53.74	46.26	44.65	47.48	141	697	838
MacForget	51.51	48.49	47.88	44.93	**140**	719	859
FedEraser	52.75	47.25	45.85	49.43	489	1,988	2,477
VeriFi	49.36	50.64	49.61	44.30	145	735	880
Class-Discriminative Pruning	51.33	48.67	47.50	53.32	174	565	**739**
UN	53.79	46.21	43.89	42.98	170	721	891
RCAD	51.13	48.87	48.45	42.27	312	3,346	3,658
IJ	51.41	48.59	47.28	47.20	184	697	881
Noisy-GD	52.84	47.16	45.40	44.18	145	740	885
FFMU	**54.84**	**45.16**	**42.99**	**48.60**	986	**0**	986

Table 9: Performance with 8% data removal and LeNet on CIFAR-10

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	53.87	46.13	42.41	48.12	168	1,336	1,504
Knowledge Distillation	51.23	48.77	47.73	49.38	145	1,600	1,745
Rapid Retraining	49.49	50.51	46.59	50.43	**141**	1,086	1,227
MacForget	50.60	49.40	45.75	51.82	145	1,158	1,303
FedEraser	51.97	48.03	46.56	45.34	480	3,088	3,568
VeriFi	49.15	50.85	50.06	46.36	140	1,176	1,316
Class-Discriminative Pruning	51.40	48.60	46.98	44.52	165	905	1,070
UN	51.99	48.01	46.36	51.07	181	1,179	1,360
RCAD	50.27	49.73	49.78	46.06	313	5,427	5,740
IJ	52.59	47.41	47.67	45.97	176	1,132	1,308
Noisy-GD	52.26	47.74	45.82	43.27	148	1,207	1,355
FFMU	**54.84**	**45.16**	**42.86**	**48.02**	986	**0**	**986**

Table 10: Performance with 15% data removal and LeNet on CIFAR-10

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	54.49	45.51	42.57	45.92	169	2,399	2,568
Knowledge Distillation	50.98	49.02	48.59	47.26	144	2,950	3,094
Rapid Retraining	51.34	48.66	44.10	47.20	142	2,044	2,186
MacForget	50.94	49.06	47.65	48.23	142	2,016	2,158
FedEraser	53.38	46.62	44.74	47.80	476	5,660	6,136
VeriFi	52.03	47.97	47.02	47.06	**141**	2,127	2,268
Class-Discriminative Pruning	50.89	49.11	47.28	48.74	165	1,653	1,818
UN	52.53	47.47	44.43	48.88	187	2,156	2,343
RCAD	49.77	50.23	49.52	48.32	334	5,638	5,972
IJ	50.96	49.04	48.65	48.50	178	2,066	2,244
Noisy-GD	52.94	47.06	46.08	46.90	150	2,217	2,367
FFMU	**54.84**	**45.16**	**42.67**	**46.34**	986	**0**	**986**

Table 11: Performance with 5% data removal and ResNet-18 on SVHN

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	90.62	9.38	9.50	10.60	341	1,744	2,085
Knowledge Distillation	87.57	12.43	12.21	12.97	380	1,957	2,337
Rapid Retraining	88.46	11.54	11.75	13.00	325	1,792	2,117
MacForget	88.66	11.34	11.80	13.51	311	1,739	2,050
FedEraser	78.10	21.90	23.36	22.52	5,543	6,701	12,244
VeriFi	88.12	11.88	11.51	12.09	350	1,036	1,386
Class-Discriminative Pruning	86.53	13.47	12.52	14.05	382	1,004	1,386
UN	88.56	11.44	11.42	12.00	354	1,830	2,184
RCAD	78.49	21.51	21.97	21.84	1,008	1,743	2,751
IJ	87.94	12.06	11.99	13.41	324	1,000	1,324
Noisy-GD	89.01	10.99	11.92	12.26	**302**	1,006	**1,308**
FFMU	**90.71**	**9.29**	**9.69**	**10.25**	1,826	**0**	1,826

Table 12: Performance with 8% data removal and ResNet-18 on SVHN

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	90.27	9.73	9.52	10.92	341	2,768	3,109
Knowledge Distillation	86.51	13.49	13.43	14.64	385	3,152	3,537
Rapid Retraining	87.67	12.33	12.29	13.26	320	2,865	3,185
MacForget	88.07	11.93	12.03	13.82	307	2,918	3,225
FedEraser	80.39	19.61	21.85	20.46	5,788	10,494	16,282
VeriFi	87.95	12.05	12.08	13.70	328	1,629	1,957
Class-Discriminative Pruning	86.07	13.93	13.61	12.84	328	1,661	1,989
UN	88.18	11.82	11.48	12.35	342	2,922	3,264
RCAD	78.93	21.07	21.14	21.52	1,011	1,824	2,835
IJ	87.82	12.18	12.05	12.66	324	1,702	2,026
Noisy-GD	87.91	12.09	12.20	13.04	**304**	1,617	1,921
FFMU	**90.71**	**9.29**	**9.67**	**10.55**	1,826	**0**	**1,826**

Table 13: Performance with 10% data removal and ResNet-18 on SVHN

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	90.93	9.07	9.25	10.41	341	3,331	3,672
Knowledge Distillation	86.07	13.93	10.25	13.69	374	4,622	4,996
Rapid Retraining	87.14	12.86	13.50	13.19	317	3,382	3,699
MacForget	87.97	12.03	12.86	13.71	**310**	3,603	3,913
FedEraser	77.16	22.84	12.32	22.93	5,793	13,180	18,973
VeriFi	87.65	12.35	22.82	13.14	328	1,942	2,270
Class-Discriminative Pruning	88.49	11.51	11.77	8.29	386	2,054	2,440
UN	87.42	12.58	11.65	12.80	345	3,578	3,923
RCAD	80.35	19.65	12.22	19.94	1,013	1,860	2,873
IJ	87.72	12.28	19.85	14.35	326	2,224	2,550
Noisy-GD	87.13	12.87	12.67	13.67	**310**	2,081	2,391
FFMU	**90.71**	**9.29**	**9.28**	**11.10**	1,826	**0**	**1,826**

Table 14: Performance with 15% data removal and ResNet-18 on SVHN

	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	90.46	9.54	9.22	10.37	341	4,712	5,053
Knowledge Distillation	86.20	13.80	14.12	15.27	393	6,954	7,347
Rapid Retraining	87.78	12.22	11.70	12.33	316	5,069	5,385
MacForget	86.99	13.01	12.75	13.54	**308**	5,418	5,726
FedEraser	77.81	22.19	22.98	23.02	5,855	18,942	24,797
VeriFi	87.94	12.06	11.96	13.09	324	2,989	3,313
Class-Discriminative Pruning	87.23	12.77	13.19	12.66	324	3,120	3,444
UN	86.42	13.58	12.83	13.36	347	5,393	5,740
RCAD	80.51	19.49	19.40	20.03	1,005	1,784	2,789
IJ	87.53	12.47	12.88	13.72	324	3,698	4,022
Noisy-GD	88.38	11.62	13.27	13.06	**308**	3,163	3,471
FFMU	**90.71**	**9.29**	**9.68**	**10.47**	1,826	**0**	**1,826**

Table 15: Performance with 20% data removal and ResNet-18 on SVHN

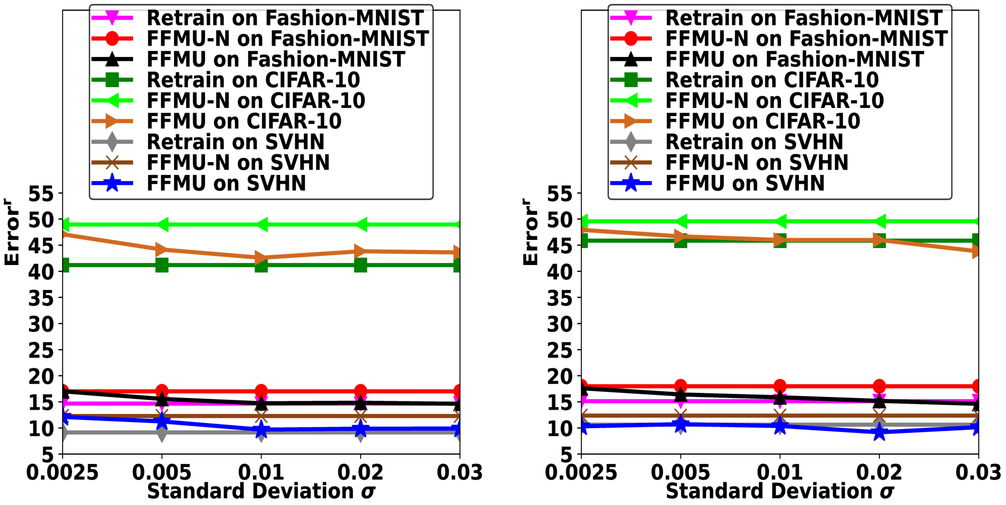
	Performance				Runtime (s)		
Metric	$Accuracy$	$Error^t$	$Error^r$	$Error^f$	Training	Unlearning	Total
Retrain	90.12	9.88	9.16	10.63	341	6,256	6,597
Knowledge Distillation	85.20	14.80	14.89	15.95	379	9,106	9,485
Rapid Retraining	88.05	11.95	12.00	12.81	319	6,608	6,927
MacForget	86.47	13.53	13.26	14.26	310	7,682	7,992
FedEraser	80.63	19.37	20.59	19.52	5,737	24,363	30,100
VeriFi	87.23	12.77	12.48	13.67	322	3,908	4,230
Class-Discriminative Pruning	87.84	12.16	13.03	8.55	334	3,994	4,328
UN	88.22	11.78	11.51	11.97	355	7,356	7,711
RCAD	79.11	20.89	21.04	20.98	1,010	2,033	3,043
IJ	87.11	12.89	12.32	13.82	335	5,293	5,628
Noisy-GD	87.88	12.12	12.60	12.32	**308**	4,153	4,461
FFMU	**90.71**	**9.29**	**9.68**	**10.38**	1,826	**0**	**1,826**

A.3. Parameter Sensitivity

In this section, we conduct more experiments to validate the sensitivity of various parameters in our FFMU method for the federated machine unlearning task.

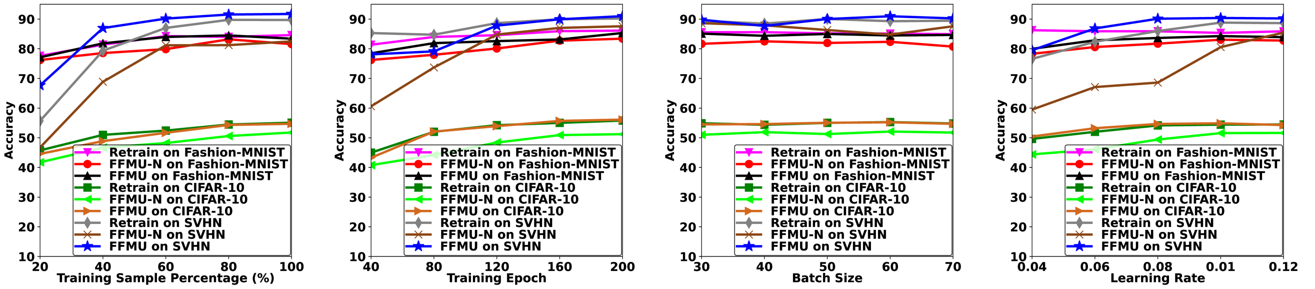
Impact of standard deviation. Figure 3 (a) and (b) measure the effect of standard deviation of the Gaussian distribution in the randomized gradient smoothing for federated machine unlearning on $Error^r$ and $Error^f$ by varying σ from 0.025 to 0.3. The error scores achieved by the Retrain model keep unchanged with varying σ. We have observed similar results in these two figures: The error curves by FFMU initially decrease quickly and then become stable when σ continuously increases. A suitable σ can help utilize the randomized gradient smoothing and quantization for directly training a federated machine unlearning model in advance. A too large σ beyond some thresholds does not affect the performance of machine unlearning any more.

Influence of training sample percentage. Figure 4 (a) shows the influence of training sample percentage in our FFMU model by varying it from 20% to 100%. We make the observations on the quality by three machine unlearning methods. (1) The accuracy by our FFMU model is very close to that of the Retrain method in most experiments. (2) The performance curves keep increasing when the number of training samples increases. (3) FFMU outperforms FFMU-N in most tests with the smallest accuracy difference with the Retrain method. When there are many training samples available ($\geq 60\%$), the quality improvement by FFMU is obvious. A reasonable explanation is more training data makes FFMU be more resilient to machine unlearning under suitable ratios of data removals.



(a) $Error^r$ (b) $Error^f$

Figure 3: Errors with varying standard deviation on three datasets



(a) Training Sample Percentage (b) Training Epoch (c) Batch Size (d) Learning Rate

Figure 4: Performance with varying parameters on three datasets

Impact of training epochs. Figure 4 (b) exhibits the sensitivity of training epochs of our FFMU model by varying them from 40 and 200. As we can see, the performance curves continuously increase with increasing training epochs. This is consistent with the fact that more training epochs makes the image classification models be resilient to machine unlearning under suitable ratios of data removals. It is observed that the accuracy scores oscillate within the range of 8.7% on three datasets.

Sensitivity of batch size. Figure 4 (c) exhibits the sensitivity of batch size of federated machine unlearning models in our FFMU model by varying them from 30 and 70. It is observed that the performance curves keep relatively stable when we continuously change the batch size. This demonstrates that our FFMU method is insensitive to the batch size of machine unlearning. No matter what the batch size is, our FFMU method can always achieve the superior performance in all tests, showing the effectiveness of our FFMU method to the machine unlearning.

Influence of learning rates. Figure 4 (d) shows the influence of learning rate in our FFMU model by varying it from 0.04 to 0.12. We have observed that the accuracy initially raises when the learning rate increases. Intuitively, a large learning rate can help the algorithm quickly find the optimal solution and thus help improve the quality of machine unlearning. Later on, the performance curves decrease quickly when the learning rate continuously increases. A reasonable explanation is that a too large learning rate may miss the optimal solution with large step size in the search process. Thus, it is important to determine the optimal learning rate for the machine unlearning.

A.4. Experimental Details

Environment. The experiments were conducted on a compute server running on Red Hat Enterprise Linux 7.2 with 2 CPUs of Intel Xeon E5-2650 v4 (at 2.66 GHz) and 8 GPUs of NVIDIA GeForce GTX 2080 Ti (with 11GB of GDDR6 on a 352-bit memory bus and memory bandwidth in the neighborhood of 620GB/s), 256GB of RAM, and 1TB of HDD. Overall, the experiments took about 5 days in a shared resource setting. We expect that a consumer-grade single-GPU machine (e.g., with a 2080 Ti GPU) could complete the full set of experiments in around 7-8 days, if its full resources were dedicated. The codes were implemented in Python 3.7.3 and PyTorch 1.0.14. We also employ Numpy 1.16.4 and Scipy 1.3.0 in the implementation. Since the datasets used are all public datasets and our methodologies and the hyperparameter settings are explicitly described in Section 3, 4, and A.4, our codes and experiments can be easily reproduced on top of a GPU server. We promise to release our open-source codes on GitHub and maintain a project website with detailed documentation for long-term access by other researchers and end-users after the paper is accepted.

Training. We study image classification networks on three standard image datasets: Fashion-MNIST [1], CIFAR-10 [2], and SVHN [3]. The above three image datasets are all public datasets, which allow researchers to use for non-commercial research and educational purposes. We use 60,000 examples as training data and 10,000 examples as test data for Fashion-MNIST. We train the machine unlearning model on the CIFAR-10 training set and test it on the CIFAR-10 test set. We use 73,257 digits as training data and 26,032 digits as test data for SVHN. We train a convolutional neural network (CNN) on Fashion-MNIST for clothing classification. We train LeNet over CIFAR-10 for image classification. We apply the ResNet-18 architecture on SVHN for street view house number identification. The neural networks are trained with Kaiming initialization (He et al., 2015) using SGD for 120 epochs with an initial learning rate of 0.05 and batch size 500. The learning rate is decayed by a factor of 0.1 at 1/2 and 3/4 of the total number of epochs. In addition, we run each experiment for 3 trials for obtaining more stable results.

Implementation. To our best knowledge, there are only two federated machine unlearning algorithms with open-source implementation: FedEraser [4] and RCAD [5]. We utilized the same model architecture as the official open-source implementation and default parameter settings provided by the original authors for machine unlearning in all experiments. All hyperparameters are standard values from reference codes or prior works.

For other regular federated learning or federated optimization approaches, including Knowledge Distillation, Rapid Retraining, MacForget, FedEraser, VeriFi, Class-Discriminative Pruning, UN, IJ, and Noisy-GD, to our best knowledge, there are no publicly available open-source implementations on the Internet. We tried our best to implement these approaches in terms of the algorithm description from the original papers. All hyperparameters are standard values from the reference papers. We validate the performance of different federated machine unlearning methods with a range of ratio of data removals $\{5\%, 8\%, 10\%, 15\%, 20\%\}$. The above open-source codes from the GitHub are licensed under the MIT License, which only requires preservation of copyright and license notices and includes the permissions of commercial use, modification, distribution, and private use.

For our FFMU model, we performed hyperparameter selection by performing a parameter sweep on standard deviation $\sigma \in \{0.025, 0.05, 0.1, 0.2, 0.3, 0.5, 1\}$ in the Gaussian distribution, quantization threshold $\lambda \in \{\sigma^2/4, \sigma^2/2, \sigma^2, 2\sigma^2, 4\sigma^2\}$, ratio of data removals $\{5\%, 8\%, 10\%, 15\%, 20\%\}$, local epochs of the machine unlearning model $\in \{1, 2, 3, 4, 5\}$, global epochs of the machine unlearning model $\in \{40, 80, 120, 160, 200\}$, batch size for training the model $\in \{30, 40, 50, 60, 70\}$, and learning rate $\in \{0.04, 0.06, 0.08, 0.1, 0.12\}$. We select the best parameters over 50 epochs of training and evaluate the model at test time.

Hyperparameter settings.

Unless otherwise explicitly stated, we used the following default parameter settings in the experiments.

[1] https://github.com/zalandoresearch/fashion-mnist

[2] https://www.cs.toronto.edu/∼kriz/cifar.html

[3] http://ufldl.stanford.edu/housenumbers/

[4] https://www.dropbox.com/s/1lhx962axovbbom/FedEraser-Code.zip?dl=0

[5] https://github.com/ars22/RCAD-regularizer

Table 16: Hyperparameter Settings

Parameter	Value
Training data on Fashion-MNIST	60,000
Test data ratio on Fashion-MNIST	10,000
Training data on CIFAR-10	50,000
Test data on CIFAR-10	10,000
Training data on SVHN	73,257
Test data on SVHN	26,032
Number of Edge Devices	100
Standard deviation σ in the Gaussian distribution	0.1
Quantization threshold λ	σ^2
Ratio of data removals	20%
Local epochs of the machine unlearning model	2
Global epochs of the machine unlearning model	200
Batch size for training the model	50
Learning rate	0.1

A.5. Potential Negative Societal Impacts and Limitations

In this work, the three image datasets are all open-released datasets (Xiao et al., 2017; Krizhevsky, 2009; Netzer et al., 2011), which allow researchers to use for non-commercial research and educational purposes. These three datasets are widely used in training/evaluating the image classification. All baseline codes are open-accessed resources that are from the GitHub and licensed under the MIT License, which only requires preservation of copyright and license notices and includes the permissions of commercial use, modification, distribution, and private use.

To the best of our knowledge, motivated the idea of PCMU, this work is the first to simultaneously execute the training and unlearning operations for improving the unlearning efficiency with machine unlearning certificate in federated setting, by leveraging the theory of nonlinear functional analysis, including Nemytskii operator and smooth manifold. Many machine learning applications often need to collect massive amount of data from third parties for model training. This raises a legitimate privacy risk: training data can be practically reconstructed from models (Fredrikson et al., 2015; Shokri et al., 2017; Veale et al., 2018; Bourtoule et al., 2021; Mahadevan & Mathioudakis, 2021; Marchant et al., 2022; Chen et al., 2022). In addition, modern privacy regulations, such as the European Union's General Data Protection Regulation (GDPR) (of the EU, 2016) and the California Consumer Privacy Act (CCPA) (Legislature, 2018), enforce the right to be forgotten, i.e., entitle data owners to the right to have their private data removed at their requests (Marchant et al., 2022; Liu et al., 2022c; Chundawat et al., 2022). Our framework is able to resolve the requests of data removal in a timely and cost-efficient manner. Our framework can play an important building block for a wide variety of privacy-critical applications that usually require near-zero tolerance of data leaking, such as financial and health data analyses. This paper is primarily of a theoretical nature. We expect our findings to produce positive impact, i.e, significantly improve the efficiency of federated machine unlearning models by simultaneously training and unlearning in advance. To our best knowledge, we do not envision any immediate negative societal impacts of our results, such as security, privacy, and fairness issues.

An important product of this paper is to explore the possibility of simultaneous training and unlearning in advance as well as one-time federated unlearning. Due to high-dimensional double integrals or non-integrable mapping between samples and labels in the randomized data smoothing and gradient quantization method, the randomized gradient smoothing and quantization approach is designed to produce high confidence certificates for the certified federated machine unlearning. Our theoretical framework can inspire further improved development and implementations on fast federated machine unlearning with better applicability and efficiency from the academic institutions and industrial research labs.