Unraveling the Connections between Privacy and Certified Robustness in Federated Learning Against Poisoning Attacks

Chulin Xie
chulinx2@illinois.edu
University of Illinois at Urbana-Champaign
Urbana, Illinois, USA

Yunhui Long
ylong4@illinois.edu
University of Illinois at Urbana-Champaign
Urbana, Illinois, USA

Pin-Yu Chen
pin-yu.chen@ibm.com
IBM Research
New York, USA

Qinbin Li
qinbin@berkeley.edu
UC Berkeley
Berkeley, California, USA

Arash Nourian
nouriara@amazon.com
Amazon Web Services
Santa Clara, California, USA

Sanmi Koyejo
sanmi@cs.stanford.edu
Stanford University
Stanford, California, USA

Bo Li
lbo@illinois.edu
University of Illinois at Urbana-Champaign
Urbana, Illinois, USA

ABSTRACT

Federated learning (FL) provides an efficient paradigm to jointly train a global model leveraging data from distributed users. As local training data comes from different users who may not be trustworthy, several studies have shown that FL is vulnerable to poisoning attacks. Meanwhile, to protect the privacy of local users, FL is usually trained in a differentially private way (DPFL). Thus, in this paper, we ask: *What are the underlying connections between differential privacy and certified robustness in FL against poisoning attacks? Can we leverage the innate privacy property of DPFL to provide certified robustness for FL? Can we further improve the privacy of FL to improve such robustness certification?* We first investigate both user-level and instance-level privacy of FL and provide formal privacy analysis to achieve improved instance-level privacy. We then provide two robustness certification criteria: *certified prediction* and *certified attack inefficacy* for DPFL on both user and instance levels. Theoretically, we provide the certified robustness of DPFL based on both criteria given a bounded number of adversarial users or instances. Empirically, we conduct extensive experiments to verify our theories under a range of poisoning attacks on different datasets. We find that increasing the level of privacy protection in DPFL results in stronger *certified attack inefficacy*; however, it does not necessarily lead to a stronger *certified prediction*. Thus, achieving the optimal certified prediction requires a proper balance between privacy and utility loss.

CCS '23, November 26–30, 2023, Copenhagen, Denmark

ACM ISBN 979-8-4007-0050-7/23/11…$15.00
https://doi.org/10.1145/3576915.3623193

CCS CONCEPTS

• **Security and privacy → Logic and verification**; **Distributed systems security**; **Privacy-preserving protocols**.

KEYWORDS

Certified Robustness, Federated Learning, Differential Privacy, Poisoning Attacks

ACM Reference Format:
Chulin Xie, Yunhui Long, Pin-Yu Chen, Qinbin Li, Arash Nourian, Sanmi Koyejo, and Bo Li. 2023. Unraveling the Connections between Privacy and Certified Robustness in Federated Learning Against Poisoning Attacks. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security (CCS '23), November 26–30, 2023, Copenhagen, Denmark.* ACM, New York, NY, USA, 26 pages. https://doi.org/10.1145/3576915.3623193

1 INTRODUCTION

Federated Learning (FL), which aims to jointly train a global model with distributed local data, has been widely deployed in different applications, such as finance [81] and medical analysis [14]. However, the fact that the local data and the training process are entirely controlled by the *local users*, who may be adversarial, raises great concerns from both security and privacy perspectives. In particular, recent studies show that FL is vulnerable to different types of training-time attacks, such as model poisoning [8, 24, 67], backdoor attacks [4, 72, 79], and label-flipping attacks [27].

Several defenses have been proposed to defend against poisoning attacks in FL. For instance, various robust aggregation methods [11, 23, 57, 61, 83] identify and down-weight the malicious updates during aggregation, or estimate a true "center" of the received updates instead of taking a weighted average directly. Other defenses include robust FL protocols (e.g., clipping [69], noisy perturbation [69], and additional evaluation during training [80]) and post-training strategies (e.g., fine-tuning and pruning [77]) that repair the poisoned global model. However, as these works mainly

focus on providing empirical robustness on specific types of attacks, they have been shown to be vulnerable to newly proposed strong adaptive attacks [24, 72, 79]. Recently, some *certified* defenses have been proposed against poisoning attacks [38, 39, 43, 65, 76], while they mainly focus on centralized setting.

In the meantime, privacy concerns have motivated FL training, where the sensitive raw data is kept on local devices without sharing. However, sharing other indirect information such as gradients or model updates during the FL training process can also leak sensitive user information [85]. As a result, approaches based on differential privacy (DP) [22], homomorphic encryption [66], and secure multiparty computation [7, 13] have been proposed to protect the privacy of users in FL. In particular, differentially private federated learning (DPFL) [28, 53, 56] provides strong privacy guarantees for user privacy, and has been deployed to real-world FL applications such as Google's Gboard [63] and Apple's Siri [64].

Recent studies observe that differential privacy (DP) is related to the robustness of ML models. Intuitively, DP is designed to protect the privacy of individual data, such that the output of an algorithm should not change much when one individual record is modified. Hence, the prediction of a DP model will be less impacted by a small amount of perturbation. Consequently, several studies have been conducted to provide empirical and certified defenses against evasion attacks [42, 47, 74] and data poisoning attacks [34, 50] based on DP properties in the *centralized* ML setting. Empirical defense against backdoor attacks [32] based on DP has also been studied in *federated learning* without theoretical guarantees [4, 56, 69]. To the best of our knowledge, despite the widespread use of DP in FL, there is no study exploring the underlying connections between DP and *certified* robustness in FL against poisoning attacks, or providing certified robustness for DPFL leveraging its privacy properties.

Hence, in this paper, we aim to bridge this gap and answer the research questions: Can we quantitatively uncover the underlying connections between differential privacy and the certified robustness of FL against poisoning attacks? Can we improve the privacy of FL to improve its certified robustness?

To explore and exploit the inherent privacy properties of DPFL for robustness certifications of FL, we mainly focus on two goals: (1) conducting thorough privacy analysis of DPFL algorithms over multiple rounds of training; (2) providing certified robustness of DPFL as a function of its privacy parameters (ϵ, δ) under different robustness criteria. In terms of privacy analysis, we revisit existing DPFL algorithms and provide improved privacy analysis. We investigate user-level DP, which is commonly guaranteed in cross-device FL to protect the sensitive information of each user [2, 3, 28, 46, 53], as well as instance-level DP which is more suitable for cross-silo FL to protect sensitive information in each data instance [49, 51, 86]. Moreover, we carry out privacy analysis for instance-level DPFL algorithms, and provide an improved guarantee for FedSGD [52]-based algorithm with privacy amplification of user and batch subsampling. We also provide a formal privacy guarantee for FedAvg [52]-based algorithm with parallel composition [54] considering local privacy budget accumulation and global privacy budget aggregation over training rounds. In terms of certified robustness of FL, we introduce two robustness criteria: *certified prediction* and *certified attack inefficacy*, which can be adapted to different threat models in DPFL. We prove that user-level (instance-level) DPFL is certifiably robust against a bounded number of adversarial users (instances). We also show that our analysis on certified robustness is *agnostic* to the type of poisoning attack strategies as long as the number of adversarial users or instances is bounded. Empirically, we quantitatively measure the relationship between privacy guarantee and the certified robustness of FL based on different robustness criteria. We present the first set of certified robustness for DPFL on image datasets MNIST, CIFAR and text dataset Tweets against various FL poisoning attacks, including backdoor attacks [4, 69], distributed backdoor attacks [79], label-flipping attacks [27], model replacement attacks [4, 8], and optimization-based model poisoning attacks [67]. From our theoretical and empirical results, we provide the following insights:

(*1*) Certified robustness in terms of *certified prediction* is influenced by both the privacy guarantee and model utility. Moderately strong privacy protection enhances certified prediction, while overly strong privacy protection can harm. This is potentially caused by the significant loss of model utility. Thus, optimal certified prediction is achieved by balancing privacy protection and utility.

(*2*) *Certified attack inefficacy* is always enhanced by stronger privacy protection. The certified lower bounds of attack inefficacy are generally tight when the number of poisoned users or instances is small, or the attack strategy is strong.

(*3*) Different DPFL algorithms yield varying certification robustness under the same privacy guarantee due to distinct training mechanisms (e.g., per-layer clipping or flat clipping).

(*4*) Larger FL data heterogeneity leads to a smaller number of tolerable adversaries for certified prediction, due to degraded utility.

Contributions. In this paper, we take the first step to characterize the underlying connections between privacy guarantees and certified robustness in FL. We hope our work can pave the way for more private and robust FL applications.

- We provide two criteria for certified robustness of FL against poisoning attacks (Section 4.2).
- Given an FL model satisfying user-level DP, we prove that it is certifiably robust against arbitrary poisoning attacks with a bounded number of adversarial users (Section 4.2).
- We revisit two instance-level DPFL algorithms and provide the improved privacy analysis (Section 5.1). We further prove that instance-level DPFL is certifiably robust against a bounded number of poisoning instances during training (Section 5.2).
- We systematically evaluate the *certified* robustness for user-level and instance-level DPFL based on two robustness criteria on both image and text datasets against five types of poisoning attacks. We provide a series of ablation studies to further analyze the factors that affect the certified robustness, such as different DPFL algorithms and data heterogeneity. Our results also indicate that our certification approach offers strong *empirical* robustness when compared to six empirical FL defenses (Section 6).

2 RELATED WORK

2.1 Differentially Private Federated Learning

To guarantee *user-level privacy* for FL, McMahan et al. [53] introduce user-level DP-FedAvg and DP-FedSGD to train language

Table 1: Comparison between our work and existing studies on privacy and robustness in the context of poisoning attacks.

	FL	DP	Empirical Robustness	Certifed Prediction	Certifed Attack Inefficacy
[43, 65, 73, 76]	×	×	✓	✓	×
[34]	×	✓	✓	×	×
[50]	×	✓	✓	×	✓
[15, 78]	✓	×	✓	✓	×
[4, 56, 69, 72]	✓	✓	✓	×	×
Our work	✓	✓	✓	✓	✓

models with millions of users, where the server clips the norm of each local update, then adds Gaussian noise on the summed update. User-level DP-FedAvg is also proposed independently by Geyer et al. [28]. Both of these works calculate the privacy budget via the moment accountant [1]. In CpSGD [2], each user clips and quantizes the model update, and adds noise drawn from Binomial distribution, achieving both communication efficiency and DP. Bhowmick et al. [9] derive DP for FL via Rényi divergence [55] and study its resilience against data reconstruction attacks. Liang et al. [46] utilize Laplacian smoothing for each local update to enhance model utility. Asoodeh and Calmon [3] propose a different way to calculate the privacy budget by interpreting each round as a Markov kernel and quantifying its impact on privacy parameters. Recent studies propose different regularization and sparsification techniques to improve utility [17] and leverage sharpness-aware optimizer [25] to make the model less sensitive to weight perturbation [68].

In terms of *instance-level privacy* for FL, Dopamine [51] provides instance-level privacy guarantee for FedSGD [52] where each user only performs one step of DP-SGD [1] at each FL round. Girgis et al. [29] introduce variants of instance-level DP-FedSGD with a trusted shuffler between the server and users to randomly permutes user gradients for privacy amplification through anonymization. Nonetheless, both works cannot be applied to the more general setting (e.g., FedAvg [52]) where each user performs multiple steps of SGD. Zhu et al. [86] privately aggregate the label predictions from users in a voting scheme and provide DP guarantees on both user and instance levels. However, it does not allow aggregating the gradients or updates and is thus not applicable to standard FL. Recent works combine local DP-SGD training of clients with personalized FL algorithms [48, 49, 58, 82] to address the user heterogeneity issue in FL and improve privacy-utility tradeoff.

In summary, the above works focus on privacy in FL while leaving its robustness unexplored. Our goal is to uncover the underlying connections between privacy guarantees with certified robustness.

2.2 Certified Robustness against Evasion Attacks

Machine learning models are susceptible to test-time evasion attacks [31], and different defenses have been proposed to enhance the robustness of models and provide certifications to guarantee consistent predictions under a specified perturbation radius [44]. Pixel-DP [42] first connects DP to certified robustness against adversarial examples by adding noise on the test sample O times and taking the expectation over the corresponding outputs. Later on, *randomized smoothing* [18] is proposed to provide a tight robustness certification. Wang et al. [74] extends Pixel-DP [42] to NLP tasks, and Liu et al. [47] improves the certification based on Rényi DP [55]. However, such an approach of adding noise to test samples does not guarantee that the training algorithm itself satisfies DP. In contrast, our certification against poisoning attacks focuses on DPFL, which requires the *training algorithm* to satisfy DP. Such analysis requires careful privacy budget analysis of DPFL models across multiple training rounds and aggregation.

2.3 Certified Robustness against Poisoning Attacks

Compared to test-time certifications against evasion attacks, training-time certifications against poisoning attacks have been less explored due to the notably different threat models and the complexity of analyzing model training dynamics, even in a centralized setting.

In **centralized setting**, current approaches primarily utilize *randomized smoothing* to certify the model robustness under a bounded number of poisoned instances. Weber et al. [76] and Rosenfeld et al. [65] propose to add noise directly to the training dataset, train multiple models on the randomized datasets, and take majority vote for the final prediction for certification. Levine and Feizi [43] and Wang et al. [73] propose to partition a centralized dataset into disjoint subsets, train an independent model on each partition, and make majority predictions among all models. However, these certifications do not apply to FL, where each local model can influence other users' local models through periodic global model aggregation, so the malicious effect of one poisoned local model could spread to all local models, making the certified robustness in FL a far more challenging task. To achieve certified robustness in **FL**, CRFL [78] clips the aggregated FL model parameters and adds noise, but it does not consider the properties provided by DPFL. Emsemble [15] trains numerous FL global models (e.g., 500) on different subsets of users and takes majority prediction. Similarly, it only leverages the randomness in user-subsampling and does not consider data privacy property during training. Our goal is to explore the underlying connections between DP properties of DPFL algorithms and their certified robustness, as well as provide recipes for achieving higher certified robustness.

Several studies have explored the robustness against poisoning attacks induced by **DP**, either in centralized learning or only empirically in FL. Ma et al. [50] first demonstrate that private learners are resistant to data poisoning for centralized regression models and analyze the lower bound of attack inefficacy. Here we extend such a lower bound of attack inefficacy from DP in centralized setting [50] to user-level DP in FL, and further derive the upper bound of the attack inefficacy. We also provide certified prediction guarantees as another robustness certification criterion for general classification tasks in FL based on the privacy properties. Meanwhile, some *empirical* studies [4, 34, 56, 69] show that DP property can mitigate backdoor attacks. For instance, in the **centralized setting**, Hong et al. [34] show that the off-the-shelf mechanism DP-SGD [1] can serve as a defense against poisoning attacks; in **FL**, [4, 69, 72] show that bounding the norm and adding Gaussian noise on model updates can mitigate backdoor attacks. Recently, Naseri et al. [56] revealed that both user-level DP and instance-level DP can defend against backdoor attacks empirically with varying levels of privacy protection. However, none of these studies provides certified robustness guarantees for DPFL or characterizes the quantitative relationships between privacy guarantees and certified robustness

in FL. In contrast, our work offers robustness certifications, which can be represented as a function of DP parameters (ϵ, δ) based on different robustness criteria. We provide an overall comparison between our work and existing studies in Table 1.

3 PRELIMINARIES

We start by providing some background on Differential Privacy (DP) and Federated Learning (FL).

Differential Privacy. DP provides a mathematically rigorous guarantee for privacy, which ensures that the output of a random algorithm is close no matter whether an individual data record is included in the input.

Definition 1 ((ϵ, δ)-DP [21]). *A randomized mechanism $\mathcal{M}: \mathcal{D} \to \Theta$ with domain $\mathcal{D}$ and output set Θ satisfies (ϵ, δ)-DP if for any pair of two adjacent datasets $d, d' \in \mathcal{D}$, and for any possible (measurable) output set $E \subseteq \Theta$, it holds that*

$$\Pr[\mathcal{M}(d) \in E] \leq e^{\epsilon} \Pr\left[\mathcal{M}\left(d'\right) \in E\right] + \delta. \tag{1}$$

Group DP follows immediately Definition 1, where the privacy guarantee decreases with the size of the group.

Definition 2 (Group DP). *For mechanism $\mathcal{M}$ that satisfies (ϵ, δ)-DP, it satisfies $(k\epsilon, \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta)$-DP for groups of size k. That is, for any $d, d' \in \mathcal{D}$ that differ by k individuals and any $E \subseteq \Theta$, it holds that*

$$\Pr[\mathcal{M}(d) \in E] \leq e^{k\epsilon} \Pr\left[\mathcal{M}\left(d'\right) \in E\right] + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta. \tag{2}$$

Federated Learning. The standard instantiation of FL is FedAvg [52], which trains a shared global model in FL without directly accessing the local training data of users. We consider an FL system consisting of N users, with B representing the set of all users (i.e., $B := [N]$) and $D := \{D_1, \ldots, D_N\}$ denoting the union of local datasets across all users. At round t, the server sends the current global model w_{t-1} to users in the selected user set U_t, where $|U_t| = m = qN$ and q is the user sampling probability. Each selected user $i \in U_t$ then locally updates the model for E local epochs with its dataset D_i and learning rate η to obtain a new local model. Then, the user sends the local model updates Δw_t^i to the server. Finally, the server aggregates over the updates from all selected users into the new global model: $w_t = w_{t-1} + \frac{1}{m}\sum_{i \in U_t} \Delta w_t^i$.

4 USER-LEVEL DP AND CERTIFIED ROBUSTNESS

4.1 User-level DP and Background

Definition 1 leaves the definition of adjacent datasets flexible, which is application-dependent. When DP is used for the privacy protection of individual users, the adjacency relation is defined as that differing by data from one user [53].

Definition 3 (User-level (ϵ, δ)-DP). *Let B, B' be two user sets. Let D and D' be the datasets that are the union of local training examples from all users in B and B', respectively. Then, D and D' are adjacent if B and B' differ by one user. The mechanism $\mathcal{M}$ satisfies user-level (ϵ, δ)-DP if it meets Definition 1 with D and D' as adjacent datasets.*

Following the standard user-level DPFL [28, 53], we introduce `UserDP-FedAvg` (Algorithm 1 in Appendix A). Specifically, at each round, the server first clips the model update from each user with a threshold S such that its ℓ_2-sensitivity is upper bounded by S. Next, the server sums up the updates, adds Gaussian noise sampled from $\mathcal{N}(0, \sigma^2 S^2)$, and takes the average:

$$w_t \leftarrow w_{t-1} + \frac{1}{m}\left(\sum_{i \in U_t} \text{Clip}(\Delta w_t^i, S) + \mathcal{N}\left(0, \sigma^2 S^2\right)\right). \tag{3}$$

During FL training, the users repeatedly query private datasets over training rounds; thus, the privacy guarantee composes. We use the existing accountant [75] based on Rényi Differential Privacy (RDP) [55] for a tight privacy budget accumulation over T rounds.

4.2 Certified Robustness of User-level DPFL

4.2.1 Threat Model. We consider there are k adversarial users (attackers) out of N users.

- **Attack Goal**: The goal of attackers is to fool the trained FL global model on the server side with specific attack objectives (e.g., misclassification).
- **Attack Capability**: In line with prior works [56, 69], for attacker capability, we consider the attacker with full control of its local training data/model. The attacker can arbitrarily manipulate the features and labels of the local data and modify the weights of the local model before submitting it to the server. However, the attacker has no control over the server operations nor over the local training process of other users. The trusted server conducts DP-related operations [28], including model update clipping and noise perturbing, so that the trained FL global model satisfies user-level DP even in the presence of attackers.
- **Attack Strategy**: The attacker strategies include backdoor attacks [16, 32], which alter local data to embed a backdoor trigger with a targeted adversarial label during local training, causing the FL global model to misclassify any test data with the backdoor trigger as the target label [4, 69, 72, 79]; label flipping attacks [10, 36] which switch the labels of local training data from one source class to a target class while keeping the data features unchanged, causing the FL global model to misclassify any test data from source class to target class [26]; and model poisoning attacks that directly manipulate local model weights to tamper global model convergence [24] or amplify the malicious effects of the attacker's model updates derived from poisoning data by scaing the updates by a factor of γ [4, 8]. Note that by providing certified robustness for FL, which is agnostic to the actual attack strategies, our work is able to explore the worst-case robustness of FL and its relationship to privacy properties.

We denote B' as the set of all users, among which k users are adversarial, and $D' := \{D'_1, \ldots, D'_{k-1}, D'_k, D_{k+1}, \ldots, D_N\}$ as the corresponding union of local datasets.

Next, we introduce two criteria for robustness certification in FL: *certified prediction* and *certified attack inefficacy*.

4.2.2 Certified Prediction. Consider the classification task with C classes. We define the classification *scoring function* $f: (\Theta, \mathbb{R}^d) \to \Upsilon^C$ which maps model parameters $\theta \in \Theta$ and an input data $x \in \mathbb{R}^d$ to a confidence vector $f(\theta, x)$, and $f_c(\theta, x) \in [0, 1]$ represents the confidence of class c. We mainly focus on the confidence after normalization, i.e., $f(\theta, x) \in \Upsilon^C = \{p \in \mathbb{R}^C_{\geq 0} : \|p\|_1 = 1\}$ in the probability simplex. Since the DP mechanism $\mathcal{M}$ is randomized and produces a *stochastic* FL global model $\theta = \mathcal{M}(D)$, it is natural

to resort to a probabilistic expression as a bridge for quantitative robustness certifications. In particular, we will use the expectation of the model's predictions to provide a quantitative guarantee on the robustness of $\mathcal{M}$. Concretely, we define the *expected scoring function* $F : (\theta, \mathbb{R}^d) \to \Upsilon^C$ where $F_c(\mathcal{M}(D), x) = \mathbb{E}[f_c(\mathcal{M}(D), x)]$ is the expected confidence for class c. The expectation is taken over DP training randomness, e.g., random Gaussian noise and random user subsampling. The corresponding *prediction* $H : (\theta, \mathbb{R}^d) \to [C]$ is defined by

$$H(\mathcal{M}(D), x) := \arg\max_{c \in [C]} F_c(\mathcal{M}(D), x), \tag{4}$$

which is the top-one class based on expected prediction confidence. We prove that such prediction allows robustness certification.

Certified Prediction under One Adversarial User. Following our threat model above and the DPFL training mechanism in Algorithm 1, we denote the trained global model exposed to a poisoned dataset D' as $\mathcal{M}(D')$. When the number of adversarial users $k = 1$, D and D' are user-level adjacent datasets according to Definition 3. Given that mechanism $\mathcal{M}$ satisfies user-level (ϵ, δ)-DP, based on the DP property, the distribution of the stochastic model $\mathcal{M}(D')$ is "close" to the distribution of $\mathcal{M}(D)$. Intuitively, according to the *post-processing property* of DP [21], during testing, given a test sample x, we would expect the values of the expected confidence for each class c, i.e., $F_c(\mathcal{M}(D'), x)$ and $F_c(\mathcal{M}(D), x)$, to be close, and hence the returned most likely class to be the *same*, i.e., $H(\mathcal{M}(D), x) = H(\mathcal{M}(D'), x)$, indicating *robust* prediction.

Theorem 1 (Certified Prediction under One Adversarial User)**.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level (ϵ, δ)-DP. For two user sets B and B' that differ by one user, let D and D' be the corresponding training datasets. For a test input x, suppose $\mathbb{A}, \mathbb{B} \in [C]$ satisfy $\mathbb{A} = \arg\max_{c \in [C]} F_c(\mathcal{M}(D), x)$ and $\mathbb{B} = \arg\max_{c \in [C]: c \neq \mathbb{A}} F_c(\mathcal{M}(D), x)$. Then, it is guaranteed that $H(\mathcal{M}(D'), x) = H(\mathcal{M}(D), x) = \mathbb{A}$ if:*

$$F_{\mathbb{A}}(\mathcal{M}(D), x) > e^{2\epsilon} F_{\mathbb{B}}(\mathcal{M}(D), x) + (1 + e^{\epsilon})\delta, \tag{5}$$

Proof sketch. The proof generalizes the analysis of pixel-level DP in test-time [42]. Specifically, with DP property for two FL neighboring datasets, we can lower bound $F_{\mathbb{A}}(\mathcal{M}(D'), x)$ based on $F_{\mathbb{A}}(\mathcal{M}(D), x)$, and upper bound $F_{\mathbb{B}}(\mathcal{M}(D'), x)$ based on $F_{\mathbb{B}}(\mathcal{M}(D), x)$. When the lower-bound of $F_{\mathbb{A}}(\mathcal{M}(D'), x)$ is strictly higher than the upper-bound of $F_{\mathbb{B}}(\mathcal{M}(D'), x)$, the predicted class will be provably $\mathbb{A}$ even under poisoning attack. Equation (5) states the condition for achieving such robustness. Full proofs are in Appendix C. □

Remark. In Theorem 1, if ϵ is large, i.e., weak privacy guarantee, such that the RHS of Equation (5) > 1, the robustness condition cannot hold since the expected confidence $F_{\mathbb{A}}(\mathcal{M}(D), x) \in [0, 1]$. On the other hand, to achieve small ϵ, i.e., strong privacy guarantee, large noise is required during training, which would hurt model utility and thus result in a small confidence margin between the top two classes (e.g., $F_{\mathbb{A}}(\mathcal{M}(D), x)$ and $F_{\mathbb{B}}(\mathcal{M}(D), x)$), making it hard to meet the robustness condition. This indicates that achieving certified prediction requires a reasonable privacy level ϵ.

Certified Prediction under k Adversarial Users. When the number of adversarial users $k > 1$, we resort to group DP. According to Definition 2, given mechanism $\mathcal{M}$ satisfying user-level (ϵ, δ)-DP, it also satisfies user-level $(k\epsilon, \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta)$-DP for groups of size k. When k is smaller than a certain threshold, leveraging the group DP property, we would expect that the distribution of the stochastic model $\mathcal{M}(D')$ is not too far away from the distribution of $\mathcal{M}(D)$ such that they would make the close prediction for a test sample with high probability. Next, we present the corresponding robustness certificate by studying the sufficient condition of k, such that the prediction for a test sample is consistent between the stochastic FL models trained from D and D' separately.

Theorem 2 (Upper Bound of k for Certified Prediction)**.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level (ϵ, δ)-DP. For two user sets B and B' that differ by k users, let D and D' be the corresponding training datasets. For a test input x, suppose $\mathbb{A}, \mathbb{B} \in [C]$ satisfy $\mathbb{A} = \arg\max_{c \in [C]} F_c(\mathcal{M}(D), x)$ and $\mathbb{B} = \arg\max_{c \in [C]: c \neq \mathbb{A}} F_c(\mathcal{M}(D), x)$, then $H(\mathcal{M}(D'), x) = H(\mathcal{M}(D), x) = \mathbb{A}, \forall k < \mathsf{K}$ where K is the certified number of adversarial users:*

$$\mathsf{K} = \frac{1}{2\epsilon} \log \frac{F_{\mathbb{A}}(\mathcal{M}(D), x)(e^{\epsilon} - 1) + \delta}{F_{\mathbb{B}}(\mathcal{M}(D), x)(e^{\epsilon} - 1) + \delta} \tag{6}$$

Proof sketch. By solving Theorem 1 combined with Group DP definition, we derive the above robustness condition. Full proofs are in Appendix C. □

Remark. **(1)** In Theorem 2, if we fix $F_{\mathbb{A}}(\mathcal{M}(D), x)$ and $F_{\mathbb{B}}(\mathcal{M}(D), x)$, the smaller ϵ of FL can certify larger K. However, smaller ϵ also induces lower confidence due to the model performance drop, thus reducing the tolerable K instead. As a result, properly choosing ϵ would help to improve the certified robustness and tolerate more adversaries during training (e.g. certify against a large K). **(2)** Theorem 2 provide a *unified* certification against k adversarial users built upon ϵ, which remains valid regardless of how ϵ is achieved. It thus offers the flexibility of choosing various types of noise, clipping, subsampling strategies, and FL training algorithms to achieve user-level ϵ. DPFL mechanisms that can retain a larger prediction confidence margin under the same ϵ can certify a larger K. **(3)** Theorem 2 is distinct from the maximum adversarial perturbation magnitude against test-time attacks provided by Pixel-DP [42] in three important aspects. First, we employ group DP to provide certifications against a discrete k number of adversarial users under the threat model of FL poisoning attacks, while Pixel-DP measures maximum perturbation magnitude using the ℓ_p-norm due to the continuous nature of pixels. Second, the certification from Pixel-DP is based on the one-time noise in the direct input perturbation during test time, leading to different closed-form solutions for different types of noise distributions such as Laplace and Gaussian. In contrast, Theorem 2 based on ϵ is a unified certification applicable to *any* user-level DP FL mechanisms. Third, the analysis of ϵ in DPFL takes into account more factors than sorely the noise, such as user subsampling and the privacy accountant techniques for DP composition over training rounds.

Certified Prediction via Rényi DP. In addition to the theoretical guarantees of DP-based certified prediction, we also derive the certified prediction based on RDP [55] with the randomized smoothing technique via Rényi Divergence [20] in Appendix D. Yet, compared to DP-based certifications, RDP-based certifications are more intricate, due to the additional parameter, RDP order α, and its foundational Rényi Divergence-based definition, which makes it

more challenging to derive a straightforward upper bound K as in Theorem 2. In our main paper, we focus on DP-based certifications for the convenience of illustration.

4.2.3 Certified Attack Inefficacy. In addition to the certified prediction, we define a bounded *attack inefficacy* for attacker $C : \Theta \to \mathbb{R}$, which quantifies the difference between the attack performance of the poisoned model and the *attack goal*, following [50]. In general, the attacker aims to minimize the *expected* attack inefficacy $J(D') := \mathbb{E}[C(\mathcal{M}(D'))]$ where $\mathcal{M}(D')$ is the global model trained from poisoned dataset D', and the expectation is taken over the randomness of DP training. The inefficacy function can be instantiated according to the concrete attack goal in different types of poisoning attacks, and we provide some examples below. For instance, in Example 1 of backdoor attack, the attack inefficacy is defined as the loss of the poisoned FL model $\theta' = \mathcal{M}(D')$ evaluated on a backdoor testset. During the FL training stage, the attacker optimizes the poisoned FL model θ' with poisoned training data, so as to minimize the attack inefficacy $C(\theta')$ during the test phase. The lower the attack inefficacy, the stronger the attack is.

Given a global FL model $\mathcal{M}(D')$ satisfying user-level (ϵ, δ)-DP, we prove the lower bound of the attack inefficacy $J(D')$ when there are at most k users. The existence of the lower bound implies that $J(D')$ can not be arbitrarily low under the constraint of k adversarial users, i.e., the attack can not be arbitrarily successful, which reflects the robustness of the trained global model. A higher lower bound of the attack inefficacy (i.e., less effective attack) indicates a more *certifiably robust* global model.

Example 1. *(Backdoor attack [32])* $C(\theta') = \frac{1}{n}\sum_{i=1}^{n} l(\theta', z_i^*)$, *where* $z_i^* = (x_i + \delta_x, y^*)$, δ_x *is the backdoor pattern,* y^* *is the target adversarial label. Minimizing* $J(D')$ *over model parameters* θ' *drives the prediction on test data with backdoor pattern* δ_x *to* y^*.

Example 2. *(Label Flipping attack [10])* $C(\theta') = \frac{1}{n}\sum_{i=1}^{n} l(\theta', z_i^*)$, *where* $z_i^* = (x_i, y^*)$ *and* y^* *is the target adversarial label. Minimizing* $J(D')$ *over model* θ' *drives the prediction on test data to* y^*.

Certified Attack Inefficacy under k Adversarial Users. We discuss our main results on certified attack inefficacy below.

Theorem 3 (Attack Inefficacy with k Attackers)**.** *Suppose a randomized mechanism* $\mathcal{M}$ *satisfies user-level* (ϵ, δ)*-DP. For two user sets* B *and* B' *that differ* k *users,* D *and* D' *are the corresponding training datasets. Let* $J(D)$ *be the expected attack inefficacy where* $|C(\theta)| \le \bar{C}, \forall \theta$*. Then,*

$$\begin{aligned}
&\min\{e^{k\epsilon}J(D) + \frac{e^{k\epsilon}-1}{e^{\epsilon}-1}\delta\bar{C}, \bar{C}\} \ge J(D') \\
&\ge \max\{e^{-k\epsilon}J(D) - \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}, 0\}, \quad \textit{if} \quad C(\cdot) \ge 0 \\
&\min\{e^{-k\epsilon}J(D) + \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}, 0\} \ge J(D') \\
&\ge \max\{e^{k\epsilon}J(D) - \frac{e^{k\epsilon}-1}{e^{\epsilon}-1}\delta\bar{C}, -\bar{C}\}, \quad \textit{if} \quad C(\cdot) \le 0
\end{aligned} \tag{7}$$

Proof sketch. Theorem 3 contains the lower bound and upper bound for attack inefficacy. For the lower bound, we generalize the proof from *DP in centralized learning* [50] to the *user-level DP in FL*. Concretely, we derive the lower bound of $J(D')$ based on $J(D)$ according to the satisfied condition in the Group DP definition for the neighboring datasets differing k users. In addition, we prove the upper bound by leveraging the symmetric property of DP neighboring datasets. The full proofs are omitted to Appendix C. □

Remark. In Theorem 3, **(1)** the lower bounds show to what extent the attack can reduce $J(D')$ by manipulating up to k users, i.e., how successful the attack can be. The lower bounds depend on $J(D)$, k, and ϵ. Here $J(D)$ is the attack inefficacy evaluated on the global model trained from clean dataset D, which is unrelated to the adversarial users and is only influenced by DPFL mechanism $\mathcal{M}$. When $J(D)$ is higher (i.e., the clean model $\mathcal{M}(D)$ is more robust), the DPFL model under poisoning attacks $\mathcal{M}(D')$ is more robust because the lower bounds are accordingly higher; a tighter privacy guarantee, i.e., smaller ϵ, can also lead to higher robustness certification as it increases the lower bounds. On the other hand, with larger k, the attacker ability grows and thus leads to lower $J(D')$. **(2)** The upper bounds indicate the minimal adversarial impact caused by k attackers, demonstrating the vulnerability of DPFL models in the most optimistic case (e.g., the backdoor pattern is less distinguishable). **(3)** Leveraging the above lower bounds, we can lower bound the minimum number of attackers required to reduce attack inefficacy to a certain level associated with hyperparameter τ in Corollary 1.

Corollary 1 (Lower Bound of k Given τ, extended from [50])**.** *Suppose a randomized mechanism* $\mathcal{M}$ *satisfies user-level* (ϵ, δ)*-DP. Let attack inefficacy function be* $C(\cdot)$*, the expected attack inefficacy be* $J(\cdot)$*. In order to achieve* $J(D') \le \frac{1}{\tau}J(D)$ *for* $\tau \ge 1$ *when* $0 \le C(\cdot) \le \bar{C}$*, or achieve* $J(D') \le \tau J(D)$ *for* $1 \le \tau \le -\frac{\bar{C}}{J(D)}$ *when* $-\bar{C} \le C(\cdot) \le 0$*, the number of adversarial users should satisfy the following:*

$$k \ge \frac{1}{\epsilon}\log\frac{(e^{\epsilon}-1)\,J(D)\tau + \bar{C}\delta\tau}{(e^{\epsilon}-1)\,J(D) + \bar{C}\delta\tau} \quad \textit{or } k \ge \frac{1}{\epsilon}\log\frac{(e^{\epsilon}-1)\,J(D)\tau - \bar{C}\delta}{(e^{\epsilon}-1)\,J(D) - \bar{C}\delta},$$

Proof sketch. The proof generalizes the proof of DP in centralized learning [50] to the user-level DP in FL. Consider the case $0 \le C(\cdot) \le \bar{C}$, when the lower bound of $J(D')$ in Theorem 3 is smaller than the desired attack inefficacy $\frac{1}{\tau}J(D)$, the current attack inefficacy $J(D')$ will be smaller than the desired attack inefficacy, i.e., $J(D') \le \frac{1}{\tau}J(D)$, indicating the desired attack effectiveness under k adversarial users. Corollary 1 states the aforementioned condition. The full proofs are omitted to Appendix C. □

Remark. Corollary 1 shows that stronger privacy guarantee (i.e., smaller ϵ) requires more attackers to achieve the same effect of the attack, indicating higher robustness.

5 INSTANCE-LEVEL DP AND CERTIFIED ROBUSTNESS

5.1 Instance-level Privacy

We start by introducing instance-level DP definition that protects privacy of individual instances, and guarantees that the trained stochastic FL model should not differ much if one instance is modified.

Definition 4 (Instance-level (ϵ, δ)-DP)**.** *Let* D *be the dataset that is the union of local training examples from all users. Then,* D *and* D' *are adjacent if they differ by one instance. The randomized mechanism* $\mathcal{M}$ *is instance-level* (ϵ, δ)*-DP if it meets Definition 1 with* D *and* D' *as adjacent datasets.*

Next, we revisit InsDP-FedSGD [51] and InsDP-FedAvg, where each user adds noise in each training step using DP-SGD [1] when training its local model based on Fed-SGD and Fed-Avg, respectively. Then, we formally provide the corresponding privacy analysis.

5.1.1 Instance-level DP for FedSGD. Dopamine [51] provides the first instance-level DP guarantee for the DP-SGD [1] training of FedSGD [52]. Although FedSGD performs the user sampling on the server and the batch sampling in each user during training, Dopamine neglects the privacy gain provided by random user sampling, unlike the privacy analysis in user-level DP. Therefore, we improve the privacy guarantee via privacy amplification [1, 6] with user sampling. In addition, we use the Rényi DP (RDP) accountant [75], instead of the moment accountant [1] used in Dopamine [51], for a tighter privacy budget analysis, given its tighter compositions rules based on Rényi divergence [55].

Specifically, in InsDP-FedSGD (Algorithm 2 in Appendix A), each user updates its local model by one step of DP-SGD [1] to protect the privacy of each training instance, the randomized mechanism $\mathcal{M}$ that outputs the global model preserves the instance-level DP. The one-step update for the global model can be described as follows:

$$w_t \leftarrow w_{t-1} - \frac{1}{m} \sum_{i \in U_t} \frac{\eta}{L} \left(\sum_{x_j \in b_t^i} \text{Clip}(\nabla l_i(w_{t-1}; x_j), S) + \mathcal{N}\left(0, \sigma^2 S^2\right) \right), \tag{8}$$

where b_t^i is a local batch randomly sampled from the local dataset of user i, L is the batch size, $\nabla l_i(w_{t-1}; x_j)$ is the gradient for local sample $x_j \in b_t^i$ calculated upon the current FL model w_{t-1}, and $\mathcal{N}\left(0, \sigma^2 S^2\right)$ is the Gaussian noise added to the per-sample gradient.

Proposition 1 (InsDP-FedSGD Privacy Guarantee)**.** *Given batch sampling probability p without replacement, and user sampling probability $q = \frac{m}{N}$ without replacement, FL rounds T, the clipping threshold S, the noise parameter σ, the randomized mechanism $\mathcal{M}$ in Algorithm 2 satisfies $(T\epsilon'(\alpha) + \log \frac{\alpha-1}{\alpha} - \frac{\log \delta + \log \alpha}{\alpha-1}, \delta)$-DP with $\epsilon(\alpha) = \alpha/(2m\sigma^2)$ where α is the RDP order and*

$$\epsilon'(\alpha) \leq \frac{1}{\alpha - 1} \cdot \log \left(1 + (pq)^2 \binom{\alpha}{2} \min \left\{ 4\left(e^{\epsilon(2)} - 1\right), e^{\epsilon(2)} \cdot \min\left\{2, \left(e^{\epsilon(\infty)} - 1\right)^2\right\}\right\} + \sum_{j=3}^{\alpha} (pq)^j \binom{\alpha}{j} e^{(j-1)\epsilon(j)} \min\left\{2, \left(e^{\epsilon(\infty)} - 1\right)^j\right\} \right)$$

Proof sketch. We use pq to represent *instance-level* sampling probability, T to represent FL training rounds, $\sigma\sqrt{m}$ to represent the *equivalent global noise* level. The rest of the proof follows **(1)** RDP subsampling amplification [75], **(2)** RDP composition for privacy budget accumulation over T rounds based on the RDP composition [55] and **(3)** transferring RDP guarantee to DP guarantee based on the conversion theorem [5]. □

5.1.2 Instance-level DP for FedAvg. Dopamine only allows users to perform *one* step of DP-SGD [1] during each FL round, while in practice, users are typically allowed to update their local models for many steps before submitting updates to reduce communication costs. To solve this problem, we introduce InsDP-FedAvg (Algorithm 3 in Appendix A), where each user i performs local DP-SGD for multiple steps so that the local training mechanism $\mathcal{M}^i$ satisfies instance-level DP. Then, the server aggregates the updates by FedAvg. We prove that the global mechanism $\mathcal{M}$ preserves instance-level DP using DP parallel composition theorem [54].

In InsDP-FedAvg, before FL training, local privacy costs $\{\epsilon_0^i\}_{i \in [N]}$ are initialized as 0. At round t, if user i is not selected, its local privacy cost is kept unchanged $\epsilon_t^i \leftarrow \epsilon_{t-1}^i$ since local dataset D_i is not accessed. Otherwise, user i updates the local model by running DP-SGD for V local steps with batch sampling probability p, noise level σ and clipping threshold S, and ϵ_t^i is accumulated upon ϵ_{t-1}^i via its local RDP accountant. Next, the server aggregates the updates from selected users and leverages the parallel composition in Proposition 2 to calculate the global privacy cost $\epsilon_t = \max_{i \in [N]} \epsilon_t^i$. After T rounds, the mechanism $\mathcal{M}$ that outputs the final FL global model satisfies instance-level (ϵ_T, δ)-DP.

To derive the privacy guarantee for InsDP-FedAvg, we analyze the privacy cost *accumulation* for each local model across FL training rounds, as well as the privacy cost *aggregation* during model aggregation on the server side at each round.

Proposition 2 (InsDP-FedAvg Privacy Guarantee)**.** *In Algorithm 3, during round t, the local mechanism $\mathcal{M}^i$ satisfies (ϵ_t^i, δ^i)-DP, and the global mechanism $\mathcal{M}$ satisfies $\left(\max_{i \in [N]} \epsilon_t^i, \delta^i\right)$-DP.*

Proof sketch. When D' and D differ in one instance, the modified instance only falls into one user's local dataset for any t training round, and thus parallel composability of DP [54] applies. Moreover, server aggregation does not increase privacy costs due to DP post-processing property. The local cost ϵ_i is only accumulated via the local RDP accountant. Finally, the privacy guarantee corresponds to the worst case and is obtained by taking the maximum local privacy cost across all the users. Proof is in Appendix A. □

Remark. Proposition 2 provides the privacy guarantee for trained FL global model when users perform local DP-SGD training. To achieve that, we examine the outcomes from FL local and global randomized mechanisms and analyze the accumulation of local privacy costs and subsequent aggregation of global privacy costs over different training rounds. In the centralized setting, Yu et al. [84] analyzes disjoint data batching and presents similar results. Recent studies [48, 49, 82] directly apply the results from [84] for instance-level DPFL. However, these studies lack a thorough privacy analysis in the context of FL, and our analysis fills this gap.

5.2 Certified Robustness of Instance-level DPFL

5.2.1 Threat Model. We consider there are in total k poisoned instances that the same or multiple users could control.

- **Attack Goal.** The goal of attackers is to mislead the trained global model to make mispredictions by injecting poisoning data during local training.
- **Attack Capability.** In accordance with prior work [56], for attack capability, we consider that local users, including adversaries, follow the DP training protocol to protect data privacy. That means the adversaries need to follow the training protocol to sample local data randomly during training. This scenario is realistic for instance-level DPFL because FL users often run pre-defined programs [12, 40] that implement DP mechanisms. For example, according to Bonawitz et al. [12], "If the device has been selected, the FL runtime receives the FL plan, queries the app's example store for data requested by the plan, and computes plan-determined model updates and metrics." On the other hand,

the users have full control over their training data, so they can arbitrarily manipulate the local training data. Under this setting, the trained FL model is guaranteed to satisfy instance-level DP.

- **Attack Strategy.** It includes data poisoning attacks, e.g., backdoor [16, 32] or label-flipping [10, 36]. Our analysis of certified robustness is agnostic to the specific attack strategy employed.

5.2.2 Certified Robustness. According to the group DP property and the post-processing property for the FL model with instance-level (ϵ, δ)-DP, we prove that our robust certification results for user-level DP are also applicable to instance-level DP. Below is the formal theorem (proof is given in Appendix C).

Theorem 4. *Suppose D and D' differ by k instances, and mechanism $\mathcal{M}$ satisfies instance-level (ϵ, δ)-DP. The results on user-level DPFL in Theorem 1, Theorem 2, Theorem 3, and Corollary 1 still hold for the instance-level DPFL $\mathcal{M}$, D, and D'.*

Remark. We analyze the underlying relationship between privacy and certified robustness under both user-level DPFL and instance-level DPFL, as well as the relationship between these two levels of privacy in FL. From the privacy perspective, the same ϵ for these two different privacy levels signifies different privacy scopes. One straightforward way to convert instance-level DP to user-level DP is to use Group DP [22] to incorporate all instances of a user, which could lead to a loose privacy bound. On the other hand, a randomized mechanism that satisfies (ϵ, δ) user-level DP also satisfies (ϵ, δ) instance-level DP based on their definitions. From the certified robustness perspective, the same ϵ on two different privacy levels implies different levels of robustness. When considering the ability to tolerate adversarial poisoning instances, instance-level DPFL provides rigorous certified robustness as a function of the number of poisoning instances, while user-level DPFL may indicate stronger robustness if we consider injecting all poisoning instances with one user. The user-level DPFL, however, might compromise the model utility when controlling per-user sensitivity during DP training. Thus, different types of DPFL mechanisms and algorithms may be chosen to protect both privacy and robustness considering several factors such as adversarial strategies, data types, and trained model sensitivity. Our evaluation on diverse datasets and different DPFL algorithms in Section 6 will validate our analysis and findings on both user-level and instance-level DP, as well as provide more observational insights.

6 EXPERIMENTS

In this section, we conduct the evaluation on three datasets (both image and text data) for the certified robustness of different DPFL algorithms against various poisoning attacks to verify the insights from our theorems. We highlight our main results and present some interesting findings and ablation studies.

6.1 Experimental Setup

6.1.1 Datasets and Models. We consider three datasets: image classification on MNIST, CIFAR and text sentiment analysis on tweets from Sentiment140 [30] (Sent140), which involves classifying Twitter posts as positive or negative. For MNIST, we use a CNN model with two Conv-ReLu-MaxPooling layers and two linear layers; for CIFAR, we use the CNN architecture from PyTorch differential privacy library [62] which consists of four Conv-ReLu-AveragePooling layers and one linear layer. In line with previous work on DP ML [37, 50] and backdoor attacks [70, 76], we mainly discuss the binary classification for MNIST (digit 0 and 1) and CIFAR (airplane and bird) in the main text, and defer their 10-class results to Appendix B. For Sent140, we use a two-layer LSTM classifier containing 256 hidden units with pretrained 300D GloVe embedding [60] following [45].

Table 2: Dataset description and parameters.

Algorithm	Dataset	N	m	E	V	batch size	η	S	δ	$\bar{C}$
`UserDP-FedAvg`	MNIST	200	20	10	/	60	0.02	0.7	0.0029	0.5
`UserDP-FedAvg`	CIFAR	200	40	5	/	50	0.05	1	0.0029	0.2
`UserDP-FedAvg`	Sent140	805	10	1	/	10	0.3	0.5	0.000001	1.4
`InsDP-FedAvg`	MNIST	10	10	/	25	50	0.02	0.7	0.00001	0.5
`InsDP-FedAvg`	CIFAR	10	10	/	100	50	0.05	1	0.00001	2

6.1.2 Training Setups. Unless otherwise specified, we split the training datasets for N FL users in an i.i.d manner for MNIST and CIFAR. For Sent140, the local datasets are naturally non-i.i.d, where each Twitter account corresponds to an FL user. We also study the effect of data heterogeneity degrees on certified robustness by simulating FL non-i.i.d setting based on Dirichlet distribution [35] in Section 6.2.3. FL users run SGD with learning rate η, momentum 0.9, and weight decay 0.0005 to update the local models. The training parameter setups, including the number of total users N, the number of selected users per round m, local epochs E, the number of local SGD steps V, local learning rate η, batch size, etc., are summarized in Table 2.

To simulate cross-device settings for `UserDP-FedAvg`, we follow the FL settings in previous studies and use Sent140 data with ~ 800 clients [45], and CIFAR/MNIST with 200 clients [52]. To simulate cross-silo FL settings for `InsDP-FedAvg`, we train DPFL models on MNIST and CIFAR with 10 users. Following [53] that use $\delta \approx \frac{1}{N^{1.1}}$ as privacy parameter, for `UserDP-FedAvg` we set $\delta = 0.0029$ for MNIST and CIFAR, and $\delta = 0.000001$ for Sent140 according to the total number of users; for `InsDP-FedAvg` we set $\delta = 0.00001$ according the total number of training samples. When training on CIFAR10, we follow the standard practice for differential privacy [1, 37] that fine-tunes a whole model pre-trained non-privately on CIFAR100 [41]. We refer the readers to Appendix B for more details about detailed hyperparameters for differential privacy.

6.1.3 Poisoning Attacks. We evaluate four poisoning attacks against our DPFL mechanisms, which represent the common threats in FL research. We consider *backdoor attacks (BKD)* on image datasets [4] and *label flipping attacks (LF)* on image and text datasets [27] against both levels of DPFL. For `InsDP-FedAvg`, we evaluate the worst-case where k backdoored or label-flipped instances are injected into the dataset of one user. For `UserDP-FedAvg`, we additionally evaluate the *static optimization attacks (STAT-OPT)* [67], which solve the adversarial optimization problem to find poisoning model updates, as well as *distributed backdoor attack (DBA)* [79], which decomposes the backdoor pattern into several smaller ones and embeds them into different local training sets for different adversarial users. Moreover, we also consider BKD, LF, and DBA via *model replacement* attack [4, 8] where k attackers train the local models using local datasets with α fraction of poisoned instances, and scale the malicious updates directly with hyperparameter γ, i.e., $\Delta w_t^i \leftarrow \gamma \Delta w_t^i$, before sending them to the

server. This way, the malicious updates would have a stronger impact on the FL model. Note that even when attackers perform scaling after server clipping, the sensitivity of each model update is still upper-bounded by the clipping threshold S, so the privacy guarantee of user-level DPFL still holds under poisoning attacks via model replacement.

Specifically, for the attacks against `UserDP-FedAvg`, by default, the local poison fraction is $\alpha = 100\%$, and the scale factor is $\gamma = 50$. We use the same parameters setups for all k attackers. In terms of label flipping attacks, the attackers swap the label of images in the source class (digit 1 for MNIST; bird for CIFAR; positive for Sent140) into the target label (digit 0 for MNIST; airplane for CIFAR; negative for Sent140). In terms of backdoor attacks in MNIST and CIFAR, the attackers add a triangle pattern in the right lower corner of the image as the backdoor pattern and swap the label of any sample with such pattern into the target label (digit 0 for MNIST; airplane for CIFAR). In terms of distributed backdoor attacks, the triangle pattern is evenly decomposed and injected by the k attackers. For the attacks against `InsDP-FedAvg`, the same target classes and backdoor patterns are used as `UserDP-FedAvg`.

6.1.4 Evaluation Metrics. We consider two evaluation metrics based on our robustness certification criteria.

- **Certified Accuracy.** To evaluate the *certified prediction*, we calculate certified accuracy, which is the fraction of the test set for which the poisoned DPFL model makes correct and the same prediction compared with that of the clean model. The test set can be either poisoned or clean based on Theorem 2. Given that the certifications are agnostic to the actual attack strategy, and certain attacks, such as model poisoning and label flipping, do not produce poisoned test input samples x, we use the clean test samples to calculate the certification following the standard setting of certified robustness in centralized systems [19]. Given a test set of size n, for the i-th test sample x_i, the ground truth label is y_i, the output prediction is c_i , and the number of adversarial users/instances that can be certifiably tolerated is K_i based on Equation 6. We calculate the certified accuracy given k adversarial users/instances as $\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}\{c_i = y_i \text{ and } \mathsf{K}_i \geq k\}$.
- **Lower bound of attack inefficacy.** To evaluate the *certified attack inefficacy*, we calculate the lower bound of attack inefficacy in Theorem 3: $\underline{J(D')} = \max\{e^{-k\epsilon}J(D) - \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}, 0\}$. This lower bound represents the cost of the attacker for performing poisoning attacks. The lower the certified attack inefficacy is, the less robust the model is. We evaluate the tightness of $\underline{J(D')}$ by comparing it with the empirical attack inefficacy $J(D')$ under different attacks.

6.1.5 Robustness Certification with Monte Carlo Approximation. The robustness certifications presented in our theorems depend on the expected confidence $F_c(\mathcal{M}(D), x)$ for class c or expected attack inefficacy $J(D)$. We take $F_c(\mathcal{M}(D), x)$ as an example here, and denote $F_c(\mathcal{M}(D), x)$ as $F(\mathcal{M})$ for simplicity. In practice, $F(\mathcal{M})$ is not directly used for prediction because the true expectation cannot be analytically computed for deep neural networks. *To empirically verify the insights provided by our theorems*, we follow the convention in prior work on certified robustness [15, 18, 42, 50, 65, 76] to use $\widetilde{F}(\mathcal{M})$, which is a Monte Carlo approximation of $F(\mathcal{M})$ by taking the average over O models outputs for utility evaluation in our experiments. Note that **(1)** from the *DP perspective*, increasing O increases the overall privacy budget as the sampling process re-accesses the sensitive data and consumes the privacy budget. Based on standard DP composition theory [21], calculating $\widetilde{F}(\mathcal{M})$ costs $O\epsilon$ privacy budget, where ϵ is the privacy budget consumed by training one model; **(2)** From the *robustness certification perspective*, the estimation of $\widetilde{F}(\mathcal{M})$ will be more accurate with higher confidence when we use larger O; **(3)** Using a single model for prediction is equivalent to computing $\widetilde{F}(\mathcal{M})$ with $O = 1$, leading to strong privacy protection but low confidence for the robustness certification.

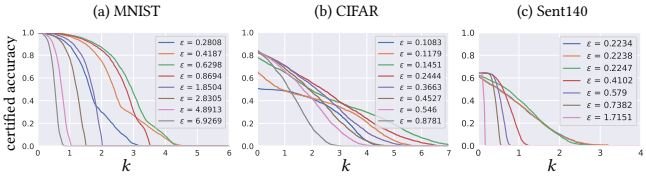

Figure 1: Certified accuracy of `UserDP-FedAvg` under different privacy budgets ϵ.

Specifically, we estimate the expected class confidence by $F_c(\mathcal{M}(D), x) \approx \frac{1}{O}\sum_{s=1}^{O} f_c^s$ to evalute Theorem 2, where each $f_c^s = f_c(\mathcal{M}(D), x)$ is obtained from one DPFL model. Similarly, we approximate the attack inefficacy to evaluate Theorem 3 and Corollary 1. We use a relatively large $O = 1000$ for certified accuracy and $O = 100$ for certified attack inefficacy in experiments so as to obtain an accurate approximation of the expectation following [50] and precisely reveal the connections between the privacy parameters (ϵ,δ) and certified robustness under different criteria. In Section 6.3.3, we use Hoeffding's inequality [33] to calibrate the empirical estimation with confidence level parameter ψ.

6.2 Evaluation Results of User-level DPFL

Here we present our main results on user-level DPFL based on the **certified accuracy** under different (1) privacy budget ϵ, (2) DPFL algorithms, and (3) data heterogeneity degrees; **empirical accuracy** under (1) different poisoning attacks and (2) comparison to empirical FL defenses; **certified and empircal attack inefficacy** under (1) different k and poisoning attacks, and (2) different ϵ.

6.2.1 Certified Accuracy under Different ϵ. Figure 1 presents the user-level certified accuracy under different ϵ by training `UserDP-FedAvg` with different noise scale σ. (The uncertified accuracy of `UserDP-FedAvg` under non-DP training and DP training is deferred to Appendix B.1.2.) Since each test sample x_i has its own certified K_i, the largest k that an FL model can reach is a threshold that none of the test samples have a larger K_i than it, i.e., $\mathsf{K}_i < k, \forall i$, which can be observed as the largest value on the x-axis of Figure 1. Note that here we calculate the certified K_i as the numerical upper bound in Theorem 2, which could be fractional.

We observe that **(1)** the largest number of adversaries k can be certified when ϵ is around 0.6298 (0.1451, 0.2238) on MNIST (CIFAR, Sent140), which verifies the relationship between ϵ and certified accuracy as discussed in Section 4.2. In particular, when ϵ is too large, K_i decreases since ϵ is in the denominator of Equation 6; when ϵ is too small, large noise is added during training, which hurts the model utility, and the model is not confident in predicting the top-1 class, thus decreasing the margin between $F_{\mathbb{A}}$ and $F_{\mathbb{B}}$ and

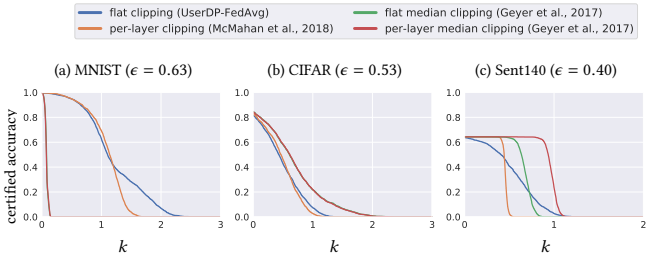

Figure 2: Certified accuracy of `UserDP-FedAvg` under different user-level DPFL algorithms with the same ϵ.

decreasing K_i. **(2)** Additionally, for each fixed k, there is an optimal ϵ that yields the maximum certified accuracy due to similar reasons. For example, to certify $k = 2$ adversaries, the ϵ with highest certified accuracy is around 0.6298 (0.2444, 0.2234) on MNIST (CIFAR, Sent140). **(3)** Given that there is a ϵ achieving maximal certified number of adversaries k or yielding the maximum certified accuracy under a fixed k, properly choosing ϵ would be important for certified accuracy. As the optimal ϵ is data/task-dependent, one can find it automatically as hyperparameter tuning. Our evaluation can serve as a guide for similar data/tasks to narrow down the search space of ϵ. **(4)** We also notice that for certain datasets like CIFAR, the ideal ϵ for certified accuracy can be small, primarily because the datasets are inherently difficult to learn. Nevertheless, on simpler datasets like MNIST, using ϵ= 0.6298 to train DPFL models remains feasible (with 97% clean accuracy) and yields the maximal certified $k \approx 4$. When DPFL algorithms offer improved utility and a larger confidence margin, a larger ϵ can be used to certify the same k, as indicated in Theorem 2. Moreover, enhanced privacy accountants that produce a tighter DP bound naturally result in a smaller ϵ without impacting model utility. As our paper focuses on deciphering the privacy-robustness interplay, our findings — both theoretical and empirical — imply opportunities to further improve the utility of current DPFL algorithms or the tightness of privacy accountants in order to achieve higher certified robustness for FL.

6.2.2 Certified Accuracy under Different DPFL Algorithms. Given that our certifications are agnostic to DPFL algorithms (i.e., the certifications hold no matter how (ϵ, δ) is achieved), we are able to compare the certified results of different DPFL algorithms given the same privacy budget ϵ. Specifically, we consider the following four DPFL algorithms with different clipping mechanisms:

- *flat clipping* (`UserDP-FedAvg`) clips the concatenation of all the layers of model update with the L2 norm threshold S.
- *per-layer clipping* [53] clips each layer of model update with the L2 norm threshold S.
- *flat median clipping* [28] uses the median[1] of the norms of clients' model updates as the threshold S for flat clipping.
- *per-layer median clipping* [28] uses the median of each layer's norms of model updates as threshold S for per-layer clipping.

We defer the detailed experimental parameters to Appendix B.1.3.

As shown in Figure 2, the models trained by different DPFL algorithms satisfying the same ϵ can have different certified robustness results. The flat clipping is able to certify the largest number of adversaries k on MNIST; while on CIFAR and Sent140, the median clipping certifies the largest k instead. Moreover, flat clipping and per-layer clipping with the same S lead to different certification results on all datasets, while the results of flat median clipping and per-layer median clipping are nearly identical on MNIST and CIFAR. We observe that no clipping mechanism is strictly better than others on all datasets. This is likely due to the significant difference in the norm of model updates when training on different datasets, which consequently affects the effectiveness of different clipping mechanisms, and thus the DP utility is dataset-dependent. Under the same DP guarantee ϵ, if one DPFL algorithm has higher utility and is more confident in predicting the ground-truth class, then it can increase the margin between the class confidences $F_{\mathbb{A}}$ and $F_{\mathbb{B}}$ in Theorem 2 and lead to a larger certified number of adversaries. Therefore, advanced DPFL protocols that have fewer clipping constraints or require less noise while achieving the same level of privacy are favored to improve both utility and certified robustness. The practitioner can use our certifications to conduct offline comparisons of different DPFL algorithms under the same ϵ, and better understand which DPFL algorithm provides better protection against poisoning attacks before real-world deployment.

[1]Strictly speaking, the median norm information can leak privacy and this slight looseness would extend to robustness certifications which leverage the DP guarantee. Nevertheless, the information leakage through the median is small, so median-clipping-based methods claimed to be DPFL in [28].

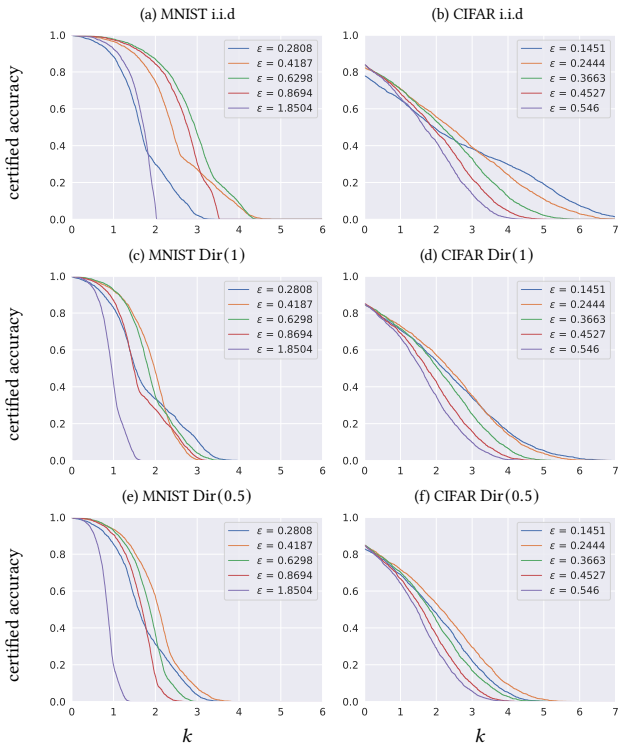

Figure 3: Certified accuracy of `UserDP-FedAvg` under varying levels of data heterogeneity. We use Dirichlet distribution $\mathrm{Dir}(\alpha)$ to create FL heterogeneous data distributions, where smaller α indicates greater heterogeneity.

6.2.3 Certified Accuracy under Different Data Heterogeneity Degrees. Recent studies [58, 82] show that DP makes the utility of the FL global model degraded more under heterogeneous data distributions among users, compared to the i.i.d data setting. Motivated by those findings, we study the impact of heterogeneity on the

certified accuracy of DPFL models. We simulate varying levels of data heterogeneity on MNIST and CIFAR using the Dirichlet distribution Dir(α), which create FL heterogeneous data partitions with different local data sizes and label distributions for users, and smaller α indicates greater heterogeneity (more non-i.i.d).

From the results in Figure 3, we find that **(1)** different non-i.i.d degrees have different optimal ϵ and the largest number of adversaries can be certified when ϵ is around 0.62, 0.28, 0.41 under the i.i.d, Dir(1), Dir(0.5) settings on MNIST, respectively. The optimal ϵ for CIFAR is around 0.14, 0.14, 0.24 under the i.i.d, Dir(1), Dir(0.5) settings, respectively. **(2)** Moreover, when FL data is more non-i.i.d, the largest number of adversaries that can be certified is smaller. This is mainly because the utility of the global model trained from the FedAvg-based DPFL degrades when FL data is more non-i.i.d, leading to a smaller confidence gap between $F_{\mathbb{A}}$ and $F_{\mathbb{B}}$ in Theorem 2. This suggests that advanced FL algorithms designed for training more accurate FL models that tackle data heterogeneity issues can be applied to DPFL settings [49, 58, 82]. By doing so, it is possible to amplify the class confidences margin between $F_{\mathbb{A}}, F_{\mathbb{B}}$ under non-i.i.d data and certify a larger k, subsequently improving both privacy-utility tradeoff and certified robustness.

6.2.4 Empircal Robust Accuracy against State-of-the-Art Poisoning Attacks. In addition to the robustness certification, our DPFL certification process that produces prediction based on Equation 4, exhibits effective robustness *empirically* against state-of-the-art poisoning attacks, even without theoretical guarantees. Table 9 in Appendix B.2.3 show that DPFL certification achieves high empirical robust accuracy on CIFAR when $k = 2, 3, 5, 10$ against different attack strategies including STAT-OPT attacks [67], BKD and LF attacks boosted by the model replacement strategy [4, 8]. Moreover, we see that the certified accuracy serves as the lower bound for the empirical robust accuracy. Details are deferred to Appendix B.2.3.

6.2.5 Comparison to Empirical FL Defenses. Another line of research is to develop empirical defenses such as robust aggregation mechanisms [11, 23, 27, 57] to detect and remove malicious users. Compared to empirical FL defenses, our work provides robustness *certifications*, while existing studies only offer *empirical* robustness. One key advantage of our analysis is that our robustness certifications provide lower bounds for model accuracy or attack inefficiency against constrained attacks, and such certification is agnostic to actual attack strategies, which means there are no future attacks that can break the certification as long as the k is within the certified range. Conversely, empirical countermeasures are typically designed against specific types of attacks, leaving them potentially vulnerable to stronger or adaptive attacks in unknown environments [24, 72]. Moreover, our certifications are general and uncover the inherent relations between DPFL and certified robustness, and DPFL algorithms with better utility or tighter privacy accountants can further enhance the certification results.

As existing FL defenses do not provide robustness guarantees and hence cannot be directly compared under our certified criteria, we compare the *empirical* robust accuracy of our certification method with *six* FL robust aggregations, including Krum [11], Multi-krum [11], Trimmed-mean [83], Median [83], Bulyan [23], RFA [61]. Table 9 in Appendix B.2.4 shows that our certification method achieves similar and even higher robust accuracy than empirical defenses under the state-of-the-art poisoning attacks on CIFAR, while our approach can further provide robustness guarantees under different criteria. We defer detailed results and discussion to Appendix B.2.4.

Moreover, it is worth noting that our certifications still hold when DPFL is combined with other empirical defense strategies. Theoretically, in the presence of such defensive mechanisms, the (ϵ, δ) privacy guarantee holds due to the post-processing property of DP, and therefore certified robustness guarantee given (ϵ, δ)-DP still holds. Combining DPFL with other robust aggregations would further enhance the empirical robustness, which remains an interesting future direction.

6.2.6 Computational Overhead and Overall Privacy Costs of Robustness Certifications. Our robustness certifications are based on DPFL, and we do not impose additional operations for DPFL, so the certifications are applicable for practical FL scenarios where the DPFL algorithm is implemented [63]. The major overhead of our certifications comes from re-training the DPFL algorithm O times for Monte-Carlo approximation (see Section 6.1.5). Notably, retraining is a common requirement when providing certifications against poisoning attacks [65, 76]. In addition, the multiple runs of re-training are parallelizable and can be speeded up with multiple GPUs. We report the running time for certifications on Sent140 in Appendix B.2.1. The re-training for Monte-Carlo approximation also increases the overall privacy costs, as discussed in Section 6.1.5. In practice, one can adjust O to prioritize robustness (i.e., a larger O for higher certification confidence), or privacy (i.e., a smaller O for fewer times of re-training). As a result, certified robustness

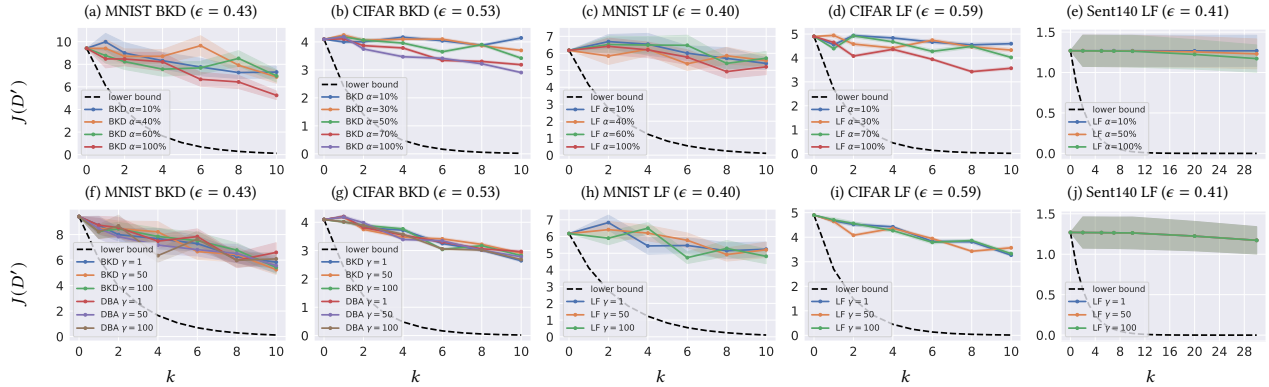

Figure 4: Certified attack inefficacy of `UserDP-FedAvg` given different k, under various attacks with different α or γ.

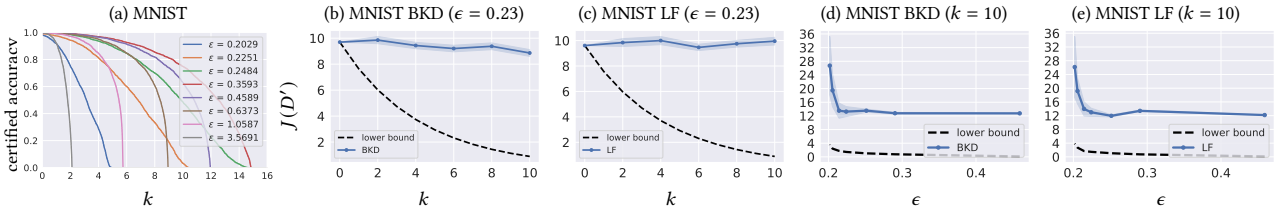

Figure 5: Certified accuracy (a) and certified attack inefficacy of `InsDP-FedAvg` on MNIST under different attacks given different k (b-c) and different ϵ (d-e).

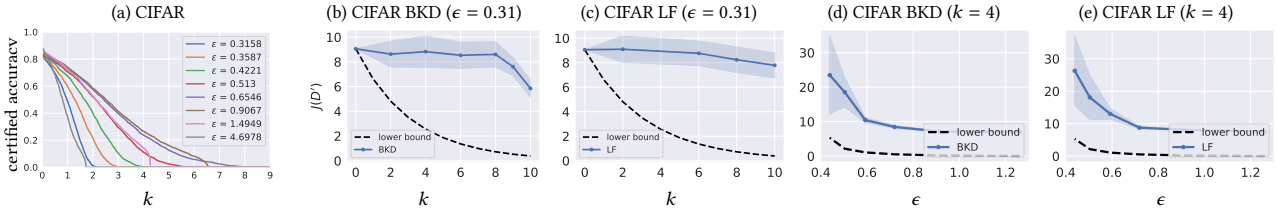

Figure 6: Certified accuracy (a) and certified attack inefficacy of `InsDP-FedAvg` on CIFAR given different k (b-c) and different ϵ (d-e).

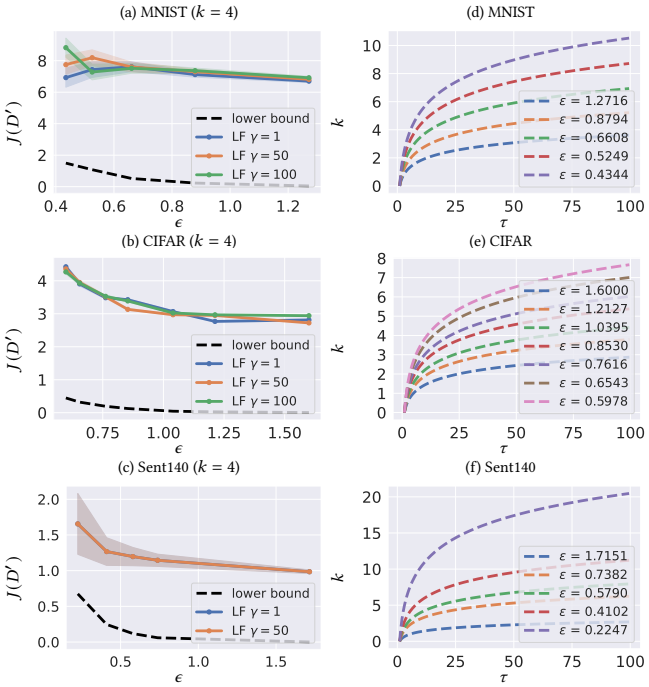

Figure 7: Certified attack inefficacy of `UserDP-FedAvg` with different ϵ (a-c), and the lower bound of k given different ϵ under different attack effectiveness τ (d-f).

can be achieved by balancing the privacy budget and robustness confidence. For example, as shown in Appendix B.2.2, the maximal certified number of adversaries on CIFAR is $k = 4$ with the overall privacy cost 10.15 (calculated by ϵO) under a confidence level of 80% (details about confidence level are deferred to Section 6.3.3).

6.2.7 Certified Attack Inefficacy under Different k and Different Poisoning Attacks. To evaluate Theorem 3 and characterize the tightness of our theoretical lower bound $\underline{J(D')}$, we compare it with the empirical attack inefficacy $J(D')$ under different local poison fraction α, attack methods and scale factor γ in Figure 4. Note that when $k = 0$, the model is benign, so the empirical attack inefficacy equals the certified one.

(1) When k increases, the attack ability grows, and both the empirical attack inefficacy and theoretical lower bound decrease.
(2) In Figure 4 row 1, given the same k, higher α, i.e., poisoning more local instances for each attacker, achieves a stronger attack, under which the empirical $J(D)$ can be achieved and is closer to the certified lower bound. This indicates that the lower bound appears tighter when the poisoning attack is stronger.
(3) In Figure 4 row 2, we fix $\alpha = 100\%$ and evaluate `UserDP-FedAvg` under different γ and attack methods. It turns out that DP serves as a strong defense empirically for FL, given that $J(D)$ did not vary much under different γ (1,50,100) and different attack methods (BKD, DBA, LF). This is because the clipping operation restricts the magnitude of malicious updates, rendering the model replacement ineffective; the Gaussian noise perturbs the malicious updates and makes the DPFL model stable, and thus the FL model is less affected by poisoning instances.
(4) In both rows, the lower bounds are tight when k is small. When k is large, there remains a gap between our lower bounds and empirical attack inefficacy under different attacks, suggesting that there is room for improvement in either devising more effective poisoning attacks or developing tighter robustness certification techniques.

6.2.8 Certified Attack Inefficacy under Different ϵ. We further explore the impacts of different factors on the certified attack inefficacy. Figure 7 (a-c) present the empirical attack inefficacy and the certified attack inefficacy lower bound given different ϵ of user-level DP. As the privacy guarantee becomes stronger (smaller ϵ), the model is more robust, achieving higher $J(D')$ and $\underline{J(D')}$. The results under the BKD attack are omitted to Appendix B.2.5.

In Figure 7 (d-f), we train user-level (ϵ, δ) DPFL models, calculate corresponding $J(D)$, and plot the lower bound of k given different

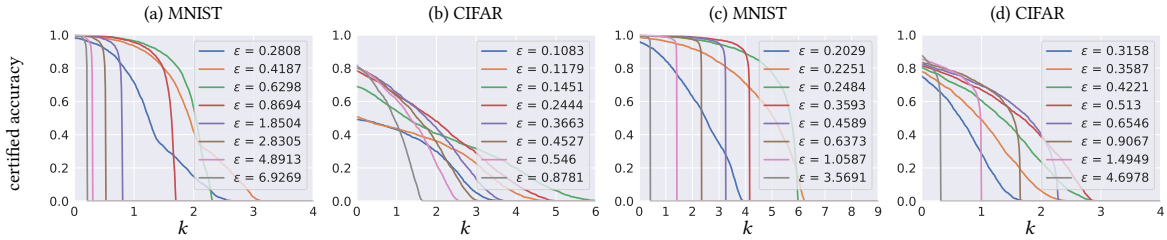

Figure 8: Certified accuracy under 99% confidence of FL satisfying user-level DP (a,b), and instance-level DP (c,d).

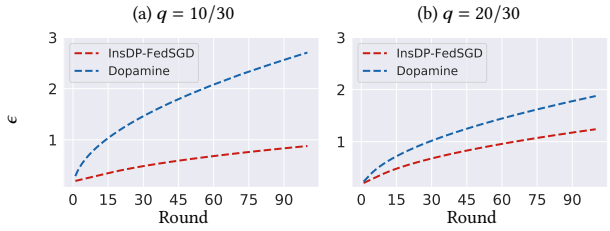

Figure 9: Privacy budget ϵ under differnt user subsampling probability q. `InsDP-FedSGD` achieves a tighter bound than Dopamine.

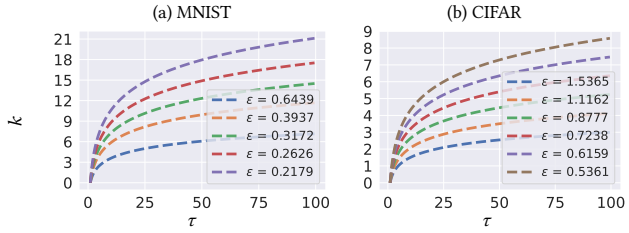

Figure 10: Lower bound of k under instance-level ϵ given attack effectiveness τ.

attack effectiveness hyperparameter τ according to Corollary 1. It shows that **(1)** when the required attack effectiveness is higher (larger τ), more attackers are required. **(2)** To achieve the same effectiveness of the attack, a fewer number of attackers is needed for larger ϵ, which means a DPFL model with weaker privacy is more vulnerable to poisoning attacks.

6.3 Evaluation Results of Instance-level DPFL

Here, we start by comparing the privacy protection between our `InsDP-FedSGD` and Dopamine, and then present certified robustness for `InsDP-FedAvg` based on **certified accuracy** under (1) different ϵ, (2) given confidence level; and **certified attack inefficacy** under (1) different k and attacks, and (2) different ϵ.

6.3.1 Privacy Bound Comparison. We compare `InsDP-FedSGD` with Dopamine, both under RDP accountant [55] for convenience of comparison, to validate the privacy amplification of `InsDP-FedSGD` provided by user subsampling. With the same noise level ($\sigma = 3.0$), clipping threshold ($S = 1.5$), and batch sampling probability ($p = 0.4$), we calculate the privacy budget under different user sampling probability $q = m/N$. Figure 9 shows that `InsDP-FedSGD` achieves tighter privacy bound over training rounds. For instance, at round 200, with $q = 10/30$, our method ($\epsilon = 0.87$) achieves a much tighter privacy guarantee than Dopamine ($\epsilon = 2.70$), which comes from user subsampling $q < 1$, while Dopamine neglects it.

6.3.2 Certified Accuracy under Different ϵ. We report the certified accuracy of `InsDP-FedAvg` under different ϵ on MNIST and CIFAR in Figure 5 (a) and Figure 6 (a). We note that the optimal ϵ that is able to certify the largest number of poisoned instances k is around 0.3593 for MNIST and 0.6546 for CIFAR. Despite the different FL setups (e.g., the total number of users) under user/instance DP, we can approximately compare the *certified robustness* in terms of the number of tolerable poisoned *instances* for the two DP levels under the same ϵ. When $\epsilon \approx 0.4$ on MNIST, `UserDP-FedAvg` can certify a maximum of $k \approx 5$ attackers, translating to a total of roughly 1250 poisoned instances, while `InsDP-FedAvg` can certify up to $k \approx 12$ poisoned instances. Therefore, `UserDP-FedAvg` can certify many more poisoned instances under the same ϵ than `InsDP-FedAvg`, though with a different privacy scope. We report the (uncertified) accuracy of `InsDP-FedAvg` in Appendix B.

6.3.3 Certified Accuracy with a Confidence Level. Here, we present the certified accuracy with the confidence level for both user and instance-level DPFL. We use Hoeffding's inequality [33] to calibrate the empirical estimation with one-sided error tolerance ψ, i.e., one-sided confidence level $1 - \psi$. We denote the empirical estimation of the class confidence for class c as $\widetilde{F}_c(\mathcal{M}(D), x) = \frac{1}{O}\sum_{o=1}^{O} f_c^s$. For a test input x, suppose $\mathbb{A}, \mathbb{B} \in [C]$ satisfy $\mathbb{A} = \arg\max_{c\in[C]} \widetilde{F}_c(\mathcal{M}(D), x)$ and $\mathbb{B} = \arg\max_{c\in[C]:c\neq\mathbb{A}} \widetilde{F}_c(\mathcal{M}(D), x)$. For a given error tolerance ψ, we use Hoeffding's inequality to compute a lower bound $\underline{F_{\mathbb{A}}(\mathcal{M}(D), x)} = \widetilde{F}_{\mathbb{A}}(\mathcal{M}(D), x) - \sqrt{\frac{\log(1/\psi)}{2O}}$ for $\mathbb{A}$, as well as a upper bound $\overline{F_{\mathbb{B}}(\mathcal{M}(D), x)} = \widetilde{F}_{\mathbb{B}}(\mathcal{M}(D), x) + \sqrt{\frac{\log(1/\psi)}{2O}}$ for $\mathbb{B}$. We use $\psi = 0.01$ (i.e., 99% confidence).

From the results in Figure 8, we observe similar trends between ϵ and certified accuracy as in Figure 1, Figure 5 (a) and Figure 6 (a). In general, the largest number of certified adversarial users in Figure 8 is smaller than the previous results because we calibrate the empirical estimation, leading to the narrowed class confidence gap between classes $\mathbb{A}$ and $\mathbb{B}$.

6.3.4 Certified Attack Inefficacy under Different k. We report the certified attack inefficacy of `InsDP-FedAvg` on MNIST and CIFAR in Figure 5 and Figure 6. We see that from Figure 5 (b)(c) and Figure 6 (b)(c), poisoning more instances (i.e., a larger k) induces lower theoretical and empirical attack inefficacy lower bounds.

6.3.5 Certified Attack Inefficacy under Different ϵ. From Figure 5 (d)(e) and Figure 6 (d)(e), it is clear that instance-level DPFL with a stronger privacy guarantee ensures higher attack inefficacy both empirically and theoretically, meaning that it is more robust against poisoning attacks. In Figure 10, we train instance-level (ϵ, δ) DPFL models, calculate corresponding $J(D)$, and plot the lower bound of k given different attack effectiveness hyperparameter τ according to Corollary 1. We can observe that fewer poisoned instances are

required to reduce the $J(D')$ to a similar level for a less private DPFL model, indicating that the model is easier to be attacked.

7 DISCUSSION & CONCLUSION

In this work, we take the first step to characterize the connections between *certified* robustness against poisoning attacks and DP in FL. We introduce two certification criteria, based on which we prove that an FL model satisfying user-level (instance-level) DP is certifiably robust against a bounded number of adversarial users (instances). We also provide formal privacy analysis to achieve improved instance-level privacy. Through comprehensive evaluations, we validate our theories and establish a general measurement framework to assess the certified robustness yielded by DPFL.

Limitations & Future Work. One limitation of our work is that we focus on the "central" DP with a trusted server for user-level DPFL, where the FL server clips and adds noise, as opposed to a "local" DP setting, where each client clips and adds their noise locally [56]. While we follow [28, 53] to consider a trusted server in the central DP regime, it offers weaker privacy protection than local DP, since the privacy guarantee does not hold against the server who can see raw client updates. It would be interesting to further extend the analysis to FL with local DP guarantees. Another limitation is that our certifications could add computational overhead. Certifying training-time robustness necessitates training multiple models, demonstrated in prior certification studies [65, 76], though this can be accelerated using parallelization and multiple GPUs.

The future directions include (1) extending our analysis to more complicated DP settings, such as scenarios where only non-attackers apply local DP in FL while attackers do not [56]; (2) combining DPFL with robust FL aggregations to further boost robustness; (3) investigating the certified robustness of advanced FL algorithms [17, 58, 68] that would maintain higher utility under DP in non-IID data settings; (4) developing tighter privacy accountant techniques over FL training to improve the certified robustness from the DP theory perspective; (5) investigating advanced model architectures and pretraining techniques to further improve the certified robustness of DPFL. We hope our work will help provide more insights into the relationships between privacy and certified robustness in the context of FL, paving the way for more secure and privacy-preserving FL applications in practice.

8 ACKNOWLEDGEMENTS

The authors thank Linyi Li and the anonymous reviewers for their valuable feedback and suggestions. This work is partially supported by the NSF grant No.1910100, No.2046726, No. 2046795, No. 2205329, NSF ACTION Institute, DARPA GARD (HR00112020007), C3AI, Alfred P. Sloan Foundation, Amazon Research Award, DARPA contract #N66001-15-C-4066, and the National Research Foundation Singapore and DSO National Laboratories under the AI Singapore Programme (AISG Award No: AISG2-RP-2020-018).

REFERENCES

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. 308–318.

[2] Naman Agarwal, Ananda Theertha Suresh, Felix Yu, Sanjiv Kumar, and H Brendan McMahan. 2018. cpSGD: communication-efficient and differentially-private distributed SGD. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. 7575–7586.

[3] Shahab Asoodeh and F Calmon. 2020. Differentially private federated learning: An information-theoretic perspective. In *ICML Workshop on Federated Learning for User Privacy and Data Confidentiality*.

[4] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. 2020. How to backdoor federated learning. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2938–2948.

[5] Borja Balle, Gilles Barthe, Marco Gaboardi, Justin Hsu, and Tetsuya Sato. 2020. Hypothesis testing interpretations and renyi differential privacy. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2496–2506.

[6] Raef Bassily, Adam Smith, and Abhradeep Thakurta. 2014. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. IEEE, 464–473.

[7] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. 1988. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*. 1–10.

[8] Arjun Nitin Bhagoji, Supriyo Chakraborty, Prateek Mittal, and Seraphin Calo. 2019. Analyzing Federated Learning through an Adversarial Lens. In *International Conference on Machine Learning*. 634–643.

[9] Abhishek Bhowmick, John Duchi, Julien Freudiger, Gaurav Kapoor, and Ryan Rogers. 2018. Protection against reconstruction and its applications in private federated learning. *arXiv preprint arXiv:1812.00984* (2018).

[10] Battista Biggio, Blaine Nelson, and Pavel Laskov. 2012. Poisoning attacks against support vector machines. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*. 1467–1474.

[11] Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. 2017. Machine learning with adversaries: Byzantine tolerant gradient descent. In *NeurIPS*. 118–128.

[12] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konečnỳ, Stefano Mazzocchi, Brendan McMahan, et al. 2019. Towards federated learning at scale: System design. *Proceedings of Machine Learning and Systems* 1 (2019), 374–388.

[13] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2017. Practical secure aggregation for privacy-preserving machine learning. In *CCS*.

[14] Theodora S Brisimi, Ruidi Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch Paschalidis, and Wei Shi. 2018. Federated learning of predictive models from federated electronic health records. *International journal of medical informatics* 112 (2018), 59–67.

[15] Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. 2021. Provably secure federated learning against malicious clients. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 6885–6893.

[16] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. 2017. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526* (2017).

[17] Anda Cheng, Peisong Wang, Xi Sheryl Zhang, and Jian Cheng. 2022. Differentially Private Federated Learning with Local Regularization and Sparsification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10122–10131.

[18] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. 2019. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*. PMLR, 1310–1320.

[19] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. 2019. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*. PMLR, 1310–1320.

[20] Krishnamurthy (Dj) Dvijotham, Jamie Hayes, Borja Balle, Zico Kolter, Chongli Qin, Andras Gyorgy, Kai Xiao, Sven Gowal, and Pushmeet Kohli. 2020. A framework for robustness certification of smoothed classifiers using f-divergences. In *International Conference on Learning Representations*.

[21] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. 2006. Our data, ourselves: Privacy via distributed noise generation. In *Advances in Cryptology – EUROCRYPT*.

[22] Cynthia Dwork and Aaron Roth. 2014. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends in Theoretical Computer Science* 9, 3-4 (2014), 211–407.

[23] El Mahdi El Mhamdi, Rachid Guerraoui, and Sébastien Louis Alexandre Rouault. 2018. The Hidden Vulnerability of Distributed Learning in Byzantium. In *International Conference on Machine Learning*.

[24] Minghong Fang, Xiaoyu Cao, Jinyuan Jia, and Neil Gong. 2020. Local model poisoning attacks to Byzantine-robust federated learning. In *USENIX Security Symposium*. 1605–1622.

[25] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. 2021. Sharpness-aware Minimization for Efficiently Improving Generalization. In *International Conference on Learning Representations*.

[26] Shuhao Fu, Chulin Xie, Bo Li, and Qifeng Chen. 2019. Attack-resistant federated learning with residual-based reweighting. *arXiv preprint arXiv:1912.11464* (2019).

[27] Clement Fung, Chris JM Yoon, and Ivan Beschastnikh. 2020. The Limitations of Federated Learning in Sybil Settings. In *23rd International Symposium on Research in Attacks, Intrusions and Defenses ({RAID} 2020)*. 301–316.

[28] Robin C Geyer, Tassilo Klein, and Moin Nabi. 2017. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557* (2017).

[29] Antonious Girgis, Deepesh Data, Suhas Diggavi, Peter Kairouz, and Ananda Theertha Suresh. 2021. Shuffled model of differential privacy in federated learning. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2521–2529.

[30] Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. (2009).

[31] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

[32] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. 2019. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access* 7 (2019), 47230–47244.

[33] Wassily Hoeffding. 1994. Probability inequalities for sums of bounded random variables. In *The Collected Works of Wassily Hoeffding*. Springer, 409–426.

[34] Sanghyun Hong, Varun Chandrasekaran, Yiğitcan Kaya, Tudor Dumitraş, and Nicolas Papernot. 2020. On the effectiveness of mitigating data poisoning attacks with gradient shaping. *arXiv preprint arXiv:2002.11497* (2020).

[35] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. 2019. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335* (2019).

[36] Ling Huang, Anthony D Joseph, Blaine Nelson, Benjamin IP Rubinstein, and J Doug Tygar. 2011. Adversarial machine learning. In *Proceedings of the 4th ACM workshop on Security and artificial intelligence*. 43–58.

[37] Matthew Jagielski, Jonathan Ullman, and Alina Oprea. 2020. Auditing Differentially Private Machine Learning: How Private is Private SGD? *Advances in Neural Information Processing Systems* 33 (2020).

[38] Jinyuan Jia, Xiaoyu Cao, and Neil Zhenqiang Gong. 2021. Intrinsic certified robustness of bagging against data poisoning attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 7961–7969.

[39] Jinyuan Jia, Yupei Liu, Xiaoyu Cao, and Neil Zhenqiang Gong. 2022. Certified Robustness of Nearest Neighbors against Data Poisoning and Backdoor Attacks. AAAI.

[40] Peter Kairouz, H Brendan McMahan, Brendan Avent, et al. 2021. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning* 14, 1–2 (2021), 1–210.

[41] Alex Krizhevsky. 2009. *Learning multiple layers of features from tiny images*. Technical Report.

[42] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. 2019. Certified Robustness to Adversarial Examples with Differential Privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*. 656–672. https://doi.org/10.1109/SP.2019.00044

[43] Alexander Levine and Soheil Feizi. 2021. Deep partition aggregation: Provable defense against general poisoning attacks. *ICLR* (2021).

[44] Linyi Li, Tao Xie, and Bo Li. 2022. SoK: Certified Robustness for Deep Neural Networks. In *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society, 94–115.

[45] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2018. Federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127* (2018).

[46] Zhicong Liang, Bao Wang, Quanquan Gu, Stanley Osher, and Yuan Yao. 2020. Exploring private federated learning with laplacian smoothing. *arXiv preprint arXiv:2005.00218* (2020).

[47] Ao Liu, Xiaoyu Chen, Sijia Liu, Lirong Xia, and Chuang Gan. 2022. Certifiably Robust Interpretation via Rényi Differential Privacy. *Artif. Intell.* 313, C (dec 2022), 14.

[48] Junxu Liu, Jian Lou, Li Xiong, Jinfei Liu, and Xiaofeng Meng. 2021. Projected federated averaging with heterogeneous differential privacy. *Proceedings of the VLDB Endowment* 15, 4 (2021), 828–840.

[49] Ken Liu, Shengyuan Hu, Steven Z Wu, and Virginia Smith. 2022. On privacy and personalization in cross-silo federated learning. *Advances in Neural Information Processing Systems* 35 (2022), 5925–5940.

[50] Yuzhe Ma, Xiaojin Zhu Zhu, and Justin Hsu. 2019. Data Poisoning against Differentially-Private Learners: Attacks and Defenses. In *International Joint Conference on Artificial Intelligence*.

[51] Mohammad Malekzadeh, Burak Hasircioglu, Nitish Mital, Kunal Katarya, Mehmet Emre Ozfatura, and Deniz Gunduz. 2021. Dopamine: Differentially Private Federated Learning on Medical Data. *The Second AAAI Workshop on Privacy-Preserving Artificial Intelligence (PPAI-21)* (2021).

[52] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, Vol. 54. PMLR, 1273–1282.

[53] H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2018. Learning Differentially Private Recurrent Language Models. In *International Conference on Learning Representations*.

[54] Frank D McSherry. 2009. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*. 19–30.

[55] Ilya Mironov. 2017. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*. IEEE, 263–275.

[56] Mohammad Naseri, Jamie Hayes, and Emiliano De Cristofaro. 2022. Local and Central Differential Privacy for Robustness and Privacy in Federated Learning. *NDSS* (2022).

[57] Thien Duc Nguyen, Phillip Rieger, Roberta De Viti, Huili Chen, et al. 2022. {FLAME}: Taming backdoors in federated learning. In *USENIX Security Symposium*.

[58] Maxence Noble, Aurélien Bellet, and Aymeric Dieuleveut. 2022. Differentially private federated learning on heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 10110–10145.

[59] Adam Paszke, Sam Gross, Francisco Massa, et al. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *NeurIPS*. 8024–8035.

[60] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[61] Krishna Pillutla, Sham M Kakade, and Zaid Harchaoui. 2019. Robust aggregation for federated learning. *arXiv preprint arXiv:1912.13445* (2019).

[62] PyTorch. 2021. Opacus – Train PyTorch models with Differential Privacy. (2021). https://opacus.ai/

[63] Google Research. 2023. Distributed differential privacy for federated learning. https://ai.googleblog.com/2023/03/distributed-differential-privacy-for.html. (2023). Accessed: 2023-08-16.

[64] MIT Technology Review. 2019. How Apple personalizes Siri without hoovering up your data. https://www.technologyreview.com/2019/12/11/131629/apple-ai-personalizes-siri-federated-learning/. (2019). Accessed: 2023-08-16.

[65] Elan Rosenfeld, Ezra Winston, Pradeep Ravikumar, and Zico Kolter. 2020. Certified robustness to label-flipping attacks via randomized smoothing. In *International Conference on Machine Learning*. PMLR, 8230–8241.

[66] Bita Darvish Rouhani, M Sadegh Riazi, and Farinaz Koushanfar. 2018. DeepSecure: Scalable provably-secure deep learning. In *Proceedings of the 55th Annual Design Automation Conference*. 1–6.

[67] Virat Shejwalkar and Amir Houmansadr. 2021. Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning. In *NDSS*.

[68] Yifan Shi, Yingqi Liu, Kang Wei, Li Shen, Xueqian Wang, and Dacheng Tao. 2023. Make Landscape Flatter in Differentially Private Federated Learning. *CVPR* (2023).

[69] Ziteng Sun, Peter Kairouz, Ananda Theertha Suresh, and H Brendan McMahan. 2019. Can you really backdoor federated learning? *arXiv preprint arXiv:1911.07963* (2019).

[70] Brandon Tran, Jerry Li, and Aleksander Madry. 2018. Spectral Signatures in Backdoor Attacks. In *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.), Vol. 31.

[71] Stephen Tu. 2013. Lecture 20: Introduction to Differential Privacy. (2013). https://stephentu.github.io/writeups/6885-lec20-b.pdf

[72] Hongyi Wang, Kartik Sreenivasan, Shashank Rajput, Harit Vishwakarma, Saurabh Agarwal, Jy-yong Sohn, Kangwook Lee, and Dimitris Papailiopoulos. 2020. Attack of the tails: Yes, you really can backdoor federated learning. *NeurIPS* (2020).

[73] Wenxiao Wang, Alexander J Levine, and Soheil Feizi. 2022. Improved certified defenses against data poisoning with (deterministic) finite aggregation. In *International Conference on Machine Learning*. PMLR, 22769–22783.

[74] Wenjie Wang, Pengfei Tang, Jian Lou, and Li Xiong. 2021. Certified robustness to word substitution attack with differential privacy. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 1102–1112.

[75] Yu-Xiang Wang, Borja Balle, and Shiva Prasad Kasiviswanathan. 2019. Subsampled rényi differential privacy and analytical moments accountant. In *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, 1226–1235.

[76] Maurice Weber, Xiaojun Xu, Bojan Karlaš, Ce Zhang, and Bo Li. 2023. Rab: Provable robustness against backdoor attacks. *IEEE Symposium on Security and Privacy (SP)* (2023).

[77] Chen Wu, Xian Yang, Sencun Zhu, and Prasenjit Mitra. 2020. Mitigating Backdoor Attacks in Federated Learning. *arXiv preprint arXiv:2011.01767* (2020).

[78] Chulin Xie, Minghao Chen, Pin-Yu Chen, and Bo Li. 2021. Crfl: Certifiably robust federated learning against backdoor attacks. In *International Conference on Machine Learning*. PMLR, 11372–11382.

[79] Chulin Xie, Keli Huang, Pin-Yu Chen, and Bo Li. 2020. Dba: Distributed backdoor attacks against federated learning. In *International Conference on Learning Representations*.

[80] Cong Xie, Oluwasanmi Koyejo, and Indranil Gupta. 2022. ZenoPS: A Distributed Learning System Integrating Communication Efficiency and Security. *Algorithms* 15, 7 (2022), 233.

[81] Wensi Yang, Yuhang Zhang, Kejiang Ye, Li Li, and Cheng-Zhong Xu. 2019. FFD: a federated learning based method for credit card fraud detection. In *International Conference on Big Data*. Springer, 18–32.
[82] Yuchen Yang, Bo Hui, Haolin Yuan, Neil Gong, and Yinzhi Cao. 2023. PRIVATEFL: Accurate, Differentially Private Federated Learning via Personalized Data Transformation. In *USENIX Security Symposium*.
[83] Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. 2018. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International Conference on Machine Learning*. PMLR, 5650–5659.
[84] Lei Yu, Ling Liu, Calton Pu, Mehmet Emre Gursoy, and Stacey Truex. 2019. Differentially private model publishing for deep learning. In *2019 IEEE symposium on security and privacy (SP)*. IEEE, 332–349.
[85] Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep Leakage from Gradients. In *NeurIPS*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.), Vol. 32. Curran Associates, Inc.
[86] Yuqing Zhu, Xiang Yu, Yi-Hsuan Tsai, Francesco Pittaluga, Masoud Faraki, Manmohan Chandraker, and Yu-Xiang Wang. 2021. Voting-based Approaches For Differentially Private Federated Learning. (2021).

The Appendix is organized as follows:

- Appendix A provides the proofs for the privacy guarantees of our DPFL algorithms.
- Appendix B provides more details on experimental setups and the additional experimental results on robustness certifications.
- Appendix C provides the proofs for the certified robustness-related analysis, including Definition 2, Theorem 1, Theorem 2, Theorem 3, Theorem 4 and Corollary 1.
- Appendix D provides the theoretical results and corresponding proofs for certified robustness against FL poisoning attacks derived from Rényi DP and Randomized Smoothing via Rényi Divergence.

A DIFFERENTIALLY PRIVATE FEDERATED LEARNING

We first present all the notations used in our paper in Table 3.

A.1 UserDP-FedAvg

To calculate the privacy costs for Algorithm 1, existing works utilize moments accountant [1] for privacy analysis [28, 53]. We note that Rényi Differential Privacy (RDP) [55] supports a tighter composition of privacy budget than the moments accounting technique for DP [55]. Therefore, we utilize RDP [55] to perform the privacy analysis in Algorithm 1. Specifically, $\mathcal{M}.accum\_priv\_spending()$ is the call on RDP accountant [75], and $\mathcal{M}.get\_privacy\_spent()$ transfers RDP guarantee to DP guarantee based on the RDP to DP conversion theorem of [5].

A.2 InsDP-FedSGD

Here, we present the algorithm `InsDP-FedSGD`.

Next, we recall Proposition 1 and present its proof.

Proposition 1 (`InsDP-FedSGD` Privacy Guarantee)**.** *Given batch sampling probability p without replacement, and user sampling probability $q = \frac{m}{N}$ without replacement, FL rounds T, the clipping threshold S, the noise parameter σ, the randomized mechanism $\mathcal{M}$ in Algorithm 2 satisfies $(T\epsilon'(\alpha) + \log\frac{\alpha-1}{\alpha} - \frac{\log\delta+\log\alpha}{\alpha-1}, \delta)$-DP with*

Algorithm 1: `UserDP-FedAvg`.

Input: Initial model w_0, user sampling probability q, privacy parameter δ, clipping threshold S, noise level σ, local datasets $D_1, ..., D_N$, local epochs E, learning rate η.
Output: FL model w_T and privacy cost ϵ

Server executes:
  for *each round* $t = 1$ **to** T **do**
    $m \leftarrow \max(q \cdot N, 1)$;
    $U_t \leftarrow$ (random subset of m users);
    for *each user* $i \in U_t$ *in parallel* **do**
      $\Delta w_t^i \leftarrow$ `UserUpdate`(i, w_{t-1}) ;
    $w_t \leftarrow w_{t-1} + \frac{1}{m}\left(\sum_{i\in U_t} \texttt{Clip}(\Delta w_t^i, S) + \mathcal{N}\left(0, \sigma^2 S^2\right)\right)$;
    $\mathcal{M}.accum\_priv\_spending(\sigma, q, \delta)$;
  $\epsilon = \mathcal{M}.get\_privacy\_spent()$;
  return w_T, ϵ
Procedure `UserUpdate`(i, w_{t-1})
  $w \leftarrow w_{t-1}$;
  for *local epoch* $e = 1$ **to** E **do**
    for *batch* $b \in$ *local dataset* D_i **do**
      $w \leftarrow w - \eta\nabla l(w; b)$
  $\Delta w_t^i \leftarrow w - w_{t-1}$;
  return Δw_t^i
Procedure `Clip`(Δ, S)
  return $\Delta / \max\left(1, \frac{\|\Delta\|_2}{S}\right)$

$\epsilon(\alpha) = \alpha/(2m\sigma^2)$ where α is the RDP order and

$$\epsilon'(\alpha) \le \frac{1}{\alpha-1}\cdot\log\left(1 + (pq)^2\binom{\alpha}{2}\min\left\{4\left(e^{\epsilon(2)}-1\right), e^{\epsilon(2)}\cdot \min\left\{2, \left(e^{\epsilon(\infty)}-1\right)^2\right\}\right\} + \sum_{j=3}^{\alpha}(pq)^j\binom{\alpha}{j}e^{(j-1)\epsilon(j)}\min\left\{2, \left(e^{\epsilon(\infty)}-1\right)^j\right\}\right)$$

Proof. **(1)** In instance-level DP, we consider the sampling probability of each instance under the combination of user-level sampling and batch-level sampling. Since the user-level sampling probability is q and the batch-level sampling probability is p, each instance is sampled with probability pq. **(2)** Additionally, since the sensitivity of instance-wise gradient w.r.t one instance is S, after local gradient descent and server FL aggregation, the equivalent sensitivity of global model w.r.t one instance is $S' = \frac{\eta S}{Lm}$ according to Eq (8). **(3)** Moreover, since the local noise is $n_i \sim \mathcal{N}(0, \sigma^2 S^2)$, the "virtual" global noise is $n = \frac{\eta}{mL}\sum_{i\in U_t} n_i$ according to Eq (8), so $n \sim \mathcal{N}(0, \frac{\eta^2\sigma^2 S^2}{mL^2})$. Let $\frac{\eta^2\sigma^2 S^2}{mL^2} = \sigma'^2 S'^2$ such that $n \sim \mathcal{N}(0, \sigma'^2 S'^2)$. Since $S' = \frac{\eta S}{Lm}$, the equivalent global noise level is $\sigma'^2 = \sigma^2 m$, i.e., $\sigma' = \sigma\sqrt{m}$. Then, we use pq to represent *instance-level* sampling probability, T to represent FL training rounds, $\sigma\sqrt{m}$ to represent the equivalent global noise level. The rest of the proof follows **(1)** RDP subsampling amplification [75], **(2)** RDP composition for privacy

Table 3: Table of notations.

Notation	Description
N	number of FL users
$D_1, \ldots, D_N$	local datasets of N users
D	$\{D_1, \ldots, D_N\}$ clean FL dataset
T	total number of communication rounds
η	learning rate
E	local epochs
q	user sampling probability
m	number of selected users at each round
U_t	selected user set at round t
w_t	global model at round t
Δw_t^i	local update of client i at round t
D'	poisoned FL dataset
k	number of adversarial users or adversarial instances
S	clipping threshold
σ	noise level
δ	DP privacy parameter
ϵ	DP privacy budget
$\mathcal{M}$	DPFL training protocol
$\mathcal{M}(D)$	clean DPFL model at round T
$\mathcal{M}(D')$	poisoned DPFL model at round T
$f_c(\mathcal{M}(D), x)$	confidence for class c on test sample x
$F_c(\mathcal{M}(D), x)$	expected confidence for class c on test sample x
$H(\mathcal{M}(D), x)$	prediction, i.e., top-1 class based on the expected confidence
$C(\mathcal{M}(D'))$	attack cost on the poisoned model $\mathcal{M}(D')$
$J(D')$	expected attack cost on the poisoned model $\mathcal{M}(D')$
$\bar{C}$	bound on attack cost $C(\cdot)$
$\bar{g}(x_j)$	clipped gradient for sample x_j in `InsDP-FedSGD`
$\widetilde{g}$	noise-perturbed and clipped gradient for sample x_j in `InsDP-FedAvg`
γ	scale factor in model replacement attack
O	number of Monte Carlo samples
ψ	one-sided error tolerance in Monte-Carlo sampling
K	theoretical upper bound for the number of adversarial users/instances that can satisfy the certified prediction
$\underline{J(D')}$	theoretical lower bound of the attack cost for poisoned DPFL model based on the certified cost
$\epsilon(\alpha)$	RDP parameter
α	RDP order

budget accumulation over T rounds based on the RDP composition [55] and **(3)** transferring RDP guarantee to DP guarantee based on the conversion theorem [5]. □

A.3 `InsDP-FedAvg`

Next, we will first consider the special case of one FL training round (i.e., $T = 1$) to showcase the privacy cost *aggregation*. Then, we will combine *local privacy cost accumulation* in each user and the privacy cost aggregation in the server for the general case with any t FL rounds. When $T = 1$, the relationship between the privacy cost of the local model $\epsilon^i, i \in [N]$ and the privacy cost of global model ϵ for one FL training round is characterized in Lemma 1. For the general case of any t FL rounds, we provide the privacy guarantee by combing the RDP accountant for the local model and the parallel composition for the global model in Proposition 2.

Lemma 1 (`InsDP-FedAvg` Privacy Guarantee when $T = 1$)**.** *In Algorithm 3, when $T = 1$, suppose local mechanism $\mathcal{M}^i$ satisfies (ϵ^i, δ^i)-DP, then global mechanism $\mathcal{M}$ satisfies $(\max_{i\in[N]} \epsilon^i, \delta^i)$-DP.*

Proof. We can regard FL as partitioning a dataset D into N disjoint subsets $\{D_1, D_2, \ldots, D_N\}$. N local randomized mechanisms $\{\mathcal{M}^1, \ldots, \mathcal{M}^N\}$ are operated on these N parts separately and each $\mathcal{M}^i$ satisfies its own ϵ^i-DP for $i \in [1, N]$. Without loss of generality, we assume the modified data sample x' ($x \to x'$ causes $D \to D'$) is in the local dataset of k-th client D_k. Then D, D' are two neighboring datasets, and D_k, D'_k are also two neighboring datasets. Consider a sequence of outcomes (i.e., local model updates) from local mechanisms $\{z_1 = \mathcal{M}^1(D_1), z_2 = \mathcal{M}^2(D_2; z_1), z_3 = \mathcal{M}^3(D_3; z_1, z_2), \ldots\}$. The global mechanism consists of a series of linear operators on the sequence $z = \mathcal{M}(D) = w_0 + \frac{1}{m}\sum_{i=1}^{N} z_i$. Note that if i-th user is

Algorithm 2: InsDP-FedSGD.

Input: Initial model w_0, user sampling probability q, privacy parameter δ, local clipping threshold S, local noise level σ, local datasets $D_1, ..., D_N$, learning rate η, batch sampling probability p.
Output: FL model w_T and privacy cost ϵ

Server executes:
for *each round* $t = 1$ **to** T **do**
  $m \leftarrow \max(q \cdot N, 1)$;
  $U_t \leftarrow$ (random subset of m clients);
  for *each user* $i \in U_t$ *in parallel* **do**
    $\Delta w_t^i \leftarrow$ UserUpdate(i, w_{t-1}) ;
  $w_t \leftarrow w_{t-1} + \frac{1}{m} \sum_{i \in U_t} \Delta w_t^i$;
  $\mathcal{M}.accum\_priv\_spending(\sqrt{m}\sigma, pq, \delta)$
$\epsilon = \mathcal{M}.get\_privacy\_spent()$;
return w_T, ϵ
Procedure UserUpdate(i, w_{t-1})
  $w \leftarrow w_{t-1}$;
  $b_t^i \leftarrow$(uniformly sample a batch from D_i with probability $p = L/|D_i|$);
  for *each* $x_j \in b_t^i$ **do**
    $g(x_j) \leftarrow \nabla l(w; x_j)$;
    $\bar{g}(x_j) \leftarrow$ Clip$(g(x_j), S)$;
  $\widetilde{g} \leftarrow \frac{1}{L}\left(\sum_j \bar{g}(x_j) + \mathcal{N}\left(0, \sigma^2 S^2\right)\right)$;
  $w \leftarrow w - \eta \widetilde{g}$;
  $\Delta w_t^i \leftarrow w - w_{t-1}$;
  return Δw_t^i
Procedure Clip(Δ, S)
  return $\Delta / \max\left(1, \frac{\|\Delta\|_2}{S}\right)$

not selected, $z_i = 0$. According to the parallel composition [71], we have

$$
\begin{aligned}
&\Pr[\mathcal{M}(D) = z] \\
&= \Pr[\mathcal{M}^1(D_1) = z_1] \cdot \Pr[\mathcal{M}^2(D_2; z_1) = z_2] \cdots \\
&\qquad \cdot \Pr[\mathcal{M}^N(D_N; z_1, \ldots, z_{N-1}) = z_N] \\
&\le \left(\exp(\epsilon^k) \Pr[\mathcal{M}^k(D'_k; z_1, \ldots, z_{k-1}) = z_k] + \delta^k\right) \\
&\qquad \cdot \prod_{i \neq k} \Pr[\mathcal{M}^i(D_i; z_1, \ldots, z_{i-1}) = z_i] \\
&= \exp(\epsilon^k) \Pr[\mathcal{M}^k(D'_k; z_1, \ldots, z_{k-1}) = z_k] \prod_{i \neq k} \Pr[\mathcal{M}^i(D_i; z_1, \ldots, z_{i-1}) = z_i] \\
&\qquad + \prod_{i \neq k} \Pr[\mathcal{M}^i(D_i; z_1, \ldots, z_{i-1}) = z_i] \delta^k \\
&= \exp(\epsilon^k) \Pr[\mathcal{M}(D') = z] + \prod_{i \neq k} \Pr[\mathcal{M}^i(D_i; z_1, \ldots, z_{i-1}) = z_i] \delta^k \\
&\le \exp(\epsilon^k) \Pr[\mathcal{M}(D') = z] + \delta^k
\end{aligned}
$$

So $\mathcal{M}$ satisfies ϵ^k-DP when the modified data sample lies in the subset D_k. Considering the worst case where the modified data samples are sampled, we derive that $\mathcal{M}$ satisfies $(\max_{i \in [N]} \epsilon^i)$-DP. □

Next, we recall Proposition 2 and present its proof.

Algorithm 3: InsDP-FedAvg.

Input: Initial model w_0, user sampling probability q, privacy parameter δ, local clipping threshold S, local noise level σ, local datasets $D_1, ..., D_N$, local steps V, learning rate η, batch sampling probability p.
Output: FL model w_T and privacy cost ϵ

Server executes:
for *each round* $t = 1$ **to** T **do**
  $m \leftarrow \max(q \cdot N, 1)$;
  $U_t \leftarrow$ (random subset of m users);
  for *each user* $i \in U_t$ *in parallel* **do**
    $\Delta w_t^i, \epsilon_t^i \leftarrow$ UserUpdate(i, w_{t-1}) ;
  for *each user* $i \notin U_t$ **do**
    $\epsilon_t^i \leftarrow \epsilon_{t-1}^i$;
  $w_t \leftarrow w_{t-1} + \frac{1}{m} \sum_{i \in U_t} \Delta w_t^i$;
  $\epsilon_t = \mathcal{M}.parallel\_composition(\{\epsilon_t^i\}_{i \in [N]})$
$\epsilon = \epsilon_T$;
return w_T, ϵ
Procedure UserUpdate(i, w_{t-1})
  $w \leftarrow w_{t-1}$;
  for *each local step* $v = 1$ **to** V **do**
    $b \leftarrow$(uniformly sample a batch from D_i with probability $p = L/|D_i|$);
    for *each* $x_j \in b$ **do**
      $g(x_j) \leftarrow \nabla l(w; x_j)$;
      $\bar{g}(x_j) \leftarrow$ Clip$(g(x_j), S)$;
    $\widetilde{g} \leftarrow \frac{1}{L}(\sum_j \bar{g}(x_j) + \mathcal{N}\left(0, \sigma^2 S^2\right))$;
    $w \leftarrow w - \eta \widetilde{g}$;
    $\mathcal{M}^i.accum\_priv\_spending(\sigma, p, \delta)$;
  $\epsilon_t^i = \mathcal{M}^i.get\_privacy\_spent()$;
  $\Delta w_t^i \leftarrow w - w_{t-1}$;
  return $\Delta w_t^i, \epsilon_t^i$
Procedure Clip(Δ, S)
  return $\Delta / \max\left(1, \frac{\|\Delta\|_2}{S}\right)$

Proposition 2 (InsDP-FedAvg Privacy Guarantee)**.** *In Algorithm 3, during round t, the local mechanism $\mathcal{M}^i$ satisfies (ϵ_t^i, δ^i)-DP, and the global mechanism $\mathcal{M}$ satisfies $\left(\max_{i \in [N]} \epsilon_t^i, \delta^i\right)$-DP.*

Proof. Again, without loss of generality, we assume the modified data sample x' ($x \to x'$ causes $D \to D'$) is in the local dataset of k-th user D_k. We first consider the case when all users are selected. At each round t, N mechanisms are operated on N disjoint parts, and each $\mathcal{M}_t^i$ satisfies its own ϵ^i-DP where ϵ^i is the privacy cost for accessing the local dataset D_i *for one round* (not accumulating over previous rounds). Let D, D' be two neighboring datasets (D_k, D'_k are also two neighboring datasets). Suppose $z_0 = \mathcal{M}_{t-1}(D)$ is the aggregated randomized global model at round $t-1$, and $\{z_1, \ldots, z_N\}$ are the randomized local updates at round t, we have a sequence of computations $\{z_1 = \mathcal{M}_t^1(D_1; z_0), z_2 = \mathcal{M}_t^2(D_2; z_0, z_1), z_3 = \mathcal{M}_t^3(D_3; z_0, z_1, z_2) \ldots\}$ and $z = \mathcal{M}_t(D) = z_0 + \frac{1}{m} \sum_i^N z_i$. We first consider the sequential composition [22] to accumulate the privacy cost over FL rounds to gain intuition. According to parallel composition, we have

$$
\begin{aligned}
&\Pr[\mathcal{M}_t(D) = z] \\
&= \Pr[\mathcal{M}_{t-1}(D) = z_0] \cdot \prod_{i=1}^{N} \Pr[\mathcal{M}_t^i(D_i; z_0, z_1, \ldots, z_{i-1}) = z_i] \\
&= \Pr[\mathcal{M}_{t-1}(D) = z_0] \cdot \Pr[\mathcal{M}_t^k(D_k; z_0, z_1, \ldots, z_{k-1}) = z_k] \\
&\quad \cdot \prod_{i \neq k} \Pr[\mathcal{M}_t^i(D_i; z_0, z_1, \ldots, z_{i-1}) = z_i] \\
&\leq \exp(\epsilon_{t-1}) \Pr[\mathcal{M}_{t-1}(D') = z_0] \\
&\quad \cdot \exp(\epsilon^k) \cdot \Pr[\mathcal{M}_t^k(D'_k; z_0, z_1, \ldots, z_{k-1}) = z_k] \\
&\quad \cdot \prod_{i \neq k} \Pr[\mathcal{M}_t^i(D_i; z_0, z_1, \ldots, z_{i-1}) = z_i] \\
&= \exp(\epsilon_{t-1} + \epsilon^k) \cdot \Pr[\mathcal{M}_t(D') = z]
\end{aligned}
$$

Therefore, $\mathcal{M}_t$ satisfies ϵ_t-DP, where $\epsilon_t = \epsilon_{t-1} + \epsilon^k$. Because the modified data sample always lies in D_k over t rounds and $\epsilon_0 = 0$, we can have $\epsilon_t = t\epsilon^k$, which means that the privacy guarantee of global mechanism $\mathcal{M}_t$ is only determined by the local mechanism of k-th user over t rounds.

Moreover, RDP accountant [75] is known to reduce the privacy cost from $\mathcal{O}(t)$ to $\mathcal{O}(\sqrt{t})$. We can use this advanced composition, instead of the sequential composition, to accumulate the privacy cost for local mechanism $\mathcal{M}^k$ over t FL rounds. In addition, we consider user selection. As described in Algorithm 3, if the user i is not selected at round t, then its local privacy cost is kept unchanged at this round.

Take the worst case of where x' could lie in, at round t, $\mathcal{M}$ satisfies ϵ_t-DP, where $\epsilon_t = \max_{i \in [N]} \epsilon_t^i$, local mechanism $\mathcal{M}^i$ satisfies ϵ_t^i-DP, and the local privacy cost ϵ_t^i is accumulated via local RDP accountant in i-th user over t rounds.

□

B EXPERIMENTAL DETAILS AND ADDITIONAL RESULTS

B.1 Experimental Details

B.1.1 Additional Implementation Details. We simulate the federated learning setup (1 server and N users) on a Linux machine with Intel® Xeon® Gold 6132 CPUs and 8 NVidia® 1080Ti GPUs. All code is implemented in Pytorch [59].

B.1.2 Training Details. Next, we summarize the privacy guarantees and clean accuracy offered when we study the certified prediction and certified attack inefficacy, which are also the training parameters setups when $k = 0$ in Figure 1, 4, 7, 6, 12, 10, 5.

User-level DPFL. In order to study the user-level certified prediction under different privacy guarantees, for MNIST, we set ϵ to be 0.2808, 0.4187, 0.6298, 0.8694, 1.8504, 2.8305, 4.8913, 6.9269, which are obtained by training `UserDP-FedAvg` FL model for 3 rounds with noise level $\sigma = 3.0, 2.3, 1.8, 1.5, 1.0, 0.8, 0.6, 0.5$, respectively (Figure 1(a)). For CIFAR, we set ϵ to be 0.1083, 0.1179, 0.1451, 0.2444, 0.3663, 0.4527, 0.5460, 0.8781, which are obtained by training `UserDP-FedAvg` FL model for one round with noise level $\sigma = 10.0, 8.0, 6.0, 4.0, 3.0, 2.6, 2.3, 1.7$, respectively (Figure 1(b)). For Sent140, we set ϵ to be 0.2234, 0.2238, 0.2247, 0.4102, 0.579, 0.7382, 1.7151, which are obtained by training `UserDP-FedAvg` FL model for three rounds with noise level $\sigma = 5, 4, 3, 2, 1.7, 1.5, 1$, respectively (Figure 1(c)).

The clean accuracy (average over 1000 runs) of `UserDP-FedAvg` under non-DP training ($\epsilon = \infty$) and DP training (varying ϵ) on MNIST, CIFAR, and Sent140 are reported in Table. 4, Table. 5 and Table. 6 respectively. We note that smaller ϵ results in lower accuracy, but we evaluate small ϵ only to study the relationship between privacy and certified accuracy in Figure 1, so as to show the trade-off. Such extreme cases are not required for certification. For other evaluations on our paper (such as Figure 4, Figure 7), we use normal ϵ with reasonable clean accuracy.

To certify the attack inefficacy under the different number of adversarial users k (Figure 4), for MNIST, we set the noise level σ to be 2.5. When $k = 0$, after training `UserDP-FedAvg` for $T = 3, 4, 5$ rounds, we obtain FL models with privacy guarantee $\epsilon = 0.3672, 0.4025, 0.4344$ and clean accuracy (average over O runs) 86.69%, 88.76%, 88.99%. For CIFAR, we set the noise level σ to be 3.0. After training `UserDP-FedAvg` for $T = 3, 4$ rounds under $k = 0$, we obtain FL models with privacy guarantee $\epsilon = 0.5346, 0.5978$ and clean accuracy 78.63%, 78.46%. For Sent140, we set the noise level σ to be 2.0. After training `UserDP-FedAvg` for $T = 3$ rounds under $k = 0$, we obtain FL models with privacy guarantee $\epsilon = 0.4102$ and clean accuracy 58.00%.

With the interest of certifying attack inefficacy under different user-level DP guarantees (Figure 7, Figure 12), we explore the empirical attack inefficacy, and the certified attack inefficacy lower bound given different ϵ. For MNIST, we set the privacy guarantee ϵ to be 1.2716, 0.8794, 0.6608, 0.5249, 0.4344, which are obtained by training `UserDP-FedAvg` FL models for five rounds under noise level $\sigma = 1.3, 1.6, 1.9, 2.2, 2.5$, respectively, and the clean accuracy for the corresponding models are 99.50%, 99.06%, 96.52%, 93.39%, 88.99%. For CIFAR, we set the privacy guarantee ϵ to be 1.600, 1.2127, 1.0395.0.8530, 0.7616, 0.6543, 0.5978, which are obtained by training `UserDP-FedAvg` FL models for four rounds under noise level $\sigma = 1.5, 1.8, 2.0, 2.3, 2.5, 2.8, 3.0$, respectively, and the clean accuracy for the corresponding models are 85.59%, 84.52%, 83.23%, 81.90%, 81.27%, 79.23%, 78.46%. For Sent140, we use the same set of ϵ as in certified prediction.

Instance-level DPFL. To certify the prediction for instance-level DPFL under different privacy guarantees, for MNIST, we set privacy cost ϵ to be 0.2029, 0.2251, 0.2484, 0.3593, 0.4589, 0.6373, 1.0587, 3.5691, which are obtained by training `InsDP-FedAvg` FL models for 3 rounds with noise level $\sigma = 15, 10, 8, 5, 4, 3, 2, 1$, respectively (Figure 5(a)). For CIFAR, we set privacy cost ϵ to be 0.3158, 0.3587, 0.4221, 0.5130, 0.6546, 0.9067, 1.4949, 4.6978, which are obtained by training `InsDP-FedAvg` FL models for one round with noise level $\sigma = 8, 7, 6, 5, 4, 3, 2, 1$, respectively (Figure 6(a)). The clean accuracy (average over 1000 runs) of `InsDP-FedAvg` under non-DP training ($\epsilon = \infty$) and DP training (varying ϵ) on MNIST and CIFAR are reported in Table 7 and Table 8 respectively.

With the aim to study certified attack inefficacy under the different number of adversarial instances k, for MNIST, we set the noise level σ to be 10. When $k = 0$, after training `InsDP-FedAvg` for $T = 4$ rounds, we obtain FL models with privacy guarantee

Table 4: Clean accuracy of `UserDP-FedAvg` on MNIST

σ	0	0.5	0.6	0.8	1.0	1.5	1.8	2.3	3.0
ϵ	∞	6.9269	4.8913	2.8305	1.8504	0.8694	0.6298	0.4187	0.2808
Clean Acc.	99.66%	99.72%	99.69%	99.71%	99.59%	98.86%	97.42%	89.15%	72.79%

Table 5: Clean accuracy of `UserDP-FedAvg` on CIFAR

σ	0	1.7	2.3	2.6	3.0	4.0	6.0
ϵ	∞	0.8781	0.546	0.4527	0.3663	0.2444	0.1451
Clean Acc.	81.90%	81.82%	80.09%	79.27%	77.89%	73.07%	64.36%

Table 6: Clean accuracy of `UserDP-FedAvg` on Sent140

σ	0	1	1.5	1.7	2.0	3.0
ϵ	∞	1.7151	0.7382	0.579	0.4102	0.2247
Clean Acc.	64.33%	62.64 %	60.76 %	59.57%	58.00%	55.28%

Table 7: Clean accuracy of `InsDP-FedAvg` on MNIST

σ	0	1	2	3	4	5	8	10	15
ϵ	∞	3.5691	1.0587	0.6373	0.4589	0.3593	0.2484	0.2251	0.2029
Clean Acc.	99.85%	99.73%	99.73%	99.70%	99.65%	99.57%	97.99%	93.30%	77.12%

Table 8: Clean accuracy of `InsDP-FedAvg` on CIFAR

σ	0	1	2	3	4	5	6	7	8
ϵ	∞	4.6978	1.4949	0.9067	0.6546	0.513	0.4221	0.3587	0.3158
Clean Acc.	91.15%	87.91%	86.02%	83.85%	81.43%	77.59%	72.69%	66.47%	62.26%

$\epsilon = 0.2383$ and clean accuracy (average over O runs) 96.40% (Figure 5(b)(c)). For CIFAR, we set the noise level σ to be 8.0. After training `InsDP-FedAvg` for one round under $k = 0$, we obtain FL models with privacy guarantee $\epsilon = 0.3158$ and clean accuracy 61.78% (Figure 6(b)(c)).

In order to study the empirical attack inefficacy and certified attack inefficacy lower bound under different instance-level DP guarantees, we set the privacy guarantee ϵ to be 0.5016, 0.311, 0.2646, 0.2318, 0.2202, 0.2096, 0.205 for MNIST, which are obtained by training `InsDP-FedAvg` FL models for six rounds under noise level $\sigma = 5, 8, 10, 13, 15, 18, 20$, respectively, and the clean accuracy for the corresponding models are 99.60%, 98.81%, 97.34%, 92.29%, 88.01%, 80.94%, 79.60% (Figure 5 (d)(e)). For CIFAR, we set the privacy guarantee ϵ to be 1.261, 0.9146, 0.7187, 0.5923, 0.5038, 0.4385, which are obtained by training `InsDP-FedAvg` FL models for two rounds under noise level $\sigma = 3, 4, 5, 6, 7, 8$, respectively, and the clean accuracy for the corresponding models are 84.47%, 80.99%, 76.01%, 68.65%, 63.07%, 60.65% (Figure 6 (d)(e)).

With the intention of exploring the upper bound for k given τ under different instance-level DP guarantee, for MNIST, we set noise level σ to be $5, 8, 10, 13, 20$, respectively, to obtain instance-DP FL models after ten rounds with privacy guarantee $\epsilon = 0.6439, 0.3937, 0.3172, 0.2626, 0.2179$ and clean accuracy 99.58%, 98.83%, 97.58%, 95.23%, 85.72% (Figure 10(a)). For CIFAR, we set noise level σ to be $3, 4, 5, 6, 7, 8$ and train `InsDP-FedAvg` for $T = 3$ rounds to obtain FL models with privacy guarantee $\epsilon = 1.5365, 1.1162, 0.8777, 0.7238, 0.6159, 0.5361$ and clean accuracy 84.34%, 80.27%, 74.62%, 66.94%, 62.14%, 59.75% (Figure 10(b)).

B.1.3 Detailed Setup for Different User-level DPFL Algorihtms. For MNIST (CIFAR, Sent140), we set ϵ to be 0.6319 (0.5346, 0.4089), which is obtained by training all DPFL algorithms with the same noise level $\sigma = 2.3$ ($\sigma = 3.0$, $\sigma = 2.0$) for same number of rounds. For flat clipping and per-layer clipping, we set $S = 0.7$ ($S = 1$, $S = 0.5$) on MNIST (CIFAR, Sent140). Except for local epoch $E = 1$, other FL parameter setups are the same as in Table 2. We set $E = 1$ because we find that the FL model in our experiments can be trained with

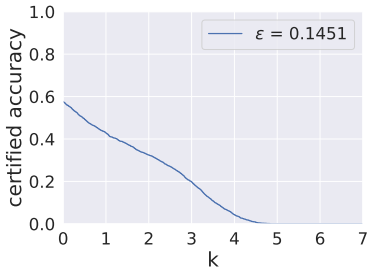

Figure 11: Certified accuracy of `UserDP-FedAvg` on CIFAR under 80% confidence with $\epsilon O = 10.15$.

median norm clipping approaches [28] only when the number of the local epoch is small. Recall that in the server aggregation step, the noise is sampled from $\mathcal{N}(0, \sigma^2 S^2)$, so S cannot be too large in order to keep the amount of noise reasonable and preserve a good model utility. As more local epoch leads to a larger norm of model updates, we set the local epoch as 1 to keep the median norm small.

B.2 Additional Experimental Results

B.2.1 Running Time Analysis for the Certifications. Compared to non-DP FL, the mechanisms introduced by DPFL, i.e., clipping and noise addition, are low-cost and easy to implement. In our experiments, the averaged running time for each communication round on Sent140 dataset is 6.06s for FedAvg and 6.11s for `UserDP-FedAvg` (averaged over 1000 times), based on a Linux machine with Intel 8 Core i7-7820X CPU and 4 NVidia 2080Ti GPUs. The major overhead of our certifications comes from re-training the DPFL algorithm O times for Monte-Carlo approximation (see Section 6.1.5). Notably, re-training is a common requirement when providing certifications against poisoning attacks [65, 76]. Also, multiple runs of training are parallelizable and can be speeded up with multiple GPUs. Given all trained models and the inference results from each model, running the certifications (e.g., averaging class confidence, and making predictions) has negligible costs, which is 0.04s on the Sent140 dataset.

B.2.2 Certifications with Moderate Overall Privacy Budget. Certified robustness can be achieved under a moderate overall privacy budget and robustness confidence. As shown in the Figure 11, on CIFAR, when $\epsilon = 0.1451$ and $O = 70$, the overall privacy cost is about $\epsilon O = 10.15$. Under the confidence level of 80%, the maximal number of adversaries that can be certified is about $k = 4$.

B.2.3 Empirical Robust Accuracy against State-of-the-art Poisoning Attacks. In this section, we evaluate our certification method against state-of-the-art poisoning attacks and report the empirical accuracy and certified accuracy. Specifically, we consider the following attacks. Static Optimization (STAT-OPT) attack [67] solves adversarial optimization problems to find optimal poisoned local model updates. We consider the "agnostic" setting of STAT-OPT attack, where the gradients of benign devices and the server's aggregation algorithm are unknown to the attacker, based on the attacker's knowledge of our settings. We evaluate two variants of STAT-OPT attack: *STAT-OPT (Min-Max)* and *STAT-OPT (Min-Sum)*; for details please refer to [67]. We also consider backdoor attack (*BKD*) and label flipping attack (*LF*) under model replacement strategy with a scale factor γ to boost malicious local update [4, 8]. For our `UserDP-FedAvg` certification approach, denoted as `UserDP-FedAvg`-*cert*, the prediction for each test sample is calculated based on Equation 4, and we train `UserDP-FedAvg` algorithms $O = 100$ times for Marto-Carlo approximation of the expected class confidence in Equation 4.

From Table 9, we see that the empirical robust accuracy of our certification method on CIFAR is high and remains stable in the presence of $k = 2, 3, 5, 10$ attackers under various attacks (i.e., less than 1%∼2% accuracy drop compared with the no-attacker setting). It shows that our DPFL certification is empirically robust against poisoning attacks.

Table 9 also shows that the certified accuracy of `UserDP-FedAvg`-cert serves as the lower bound for its empirical robust accuracy. We notice that under relatively strong attack settings such as $k = 5, 10$, our DPFL certification cannot provide non-trivial certified accuracy. Nevertheless, *our DPFL certification approach still exhibits strong empirical effective robustness*, even without theoretical guarantees. The gap between certified robust accuracy and empirical robust accuracy indicates potential advancements either in crafting stronger poisoning attacks to further reduce empirical robust accuracy, or in developing tighter robustness certification techniques to improve theoretical lower bound.

B.2.4 Comparison to Empirical FL Defenses. Here, we compare the *empirical* robust accuracy of our certification method with *six* FL robust aggregations, including *Krum* [11], *Multi-krum* [11], *Trimmed-mean* [83], *Median* [83], *Bulyan* [23], *RFA* [61].

show that our certification method `UserDP-FedAvg`-cert achieves similar and even higher accuracy than empirical defenses under state-of-the-art poisoning attacks, while providing privacy and robustness guarantees. Specifically, under the optimization-based attacks STAT-OPT (Min-Max) and STAT-OPT (Min-Sum), `UserDP-FedAvg`-cert consistently achieves higher empirical robust accuracy than other FL robust aggregation methods when $k = 2, 3, 5, 10$; under BKD and LF attacks, `UserDP-FedAvg`-cert exhibits similar robustness as FL robust aggregation methods. Note that Multi-Krum, Trimmed-mean, and Bulyan require specifying the number of attackers in their aggregation rules to detect the outliers, while our approach does not require such knowledge about attackers during DPFL training.

We also notice that when ϵ is too small (e.g., $\epsilon = 0.3205$), `UserDP-FedAvg`-cert has lower empirical robust accuracy than robust aggregation defenses. This is mainly because of the noise level being large during `UserDP-FedAvg` training to achieve a strong privacy guarantee, which hurts the utility of the DPFL model, as we can see in the no-attack setting. Therefore, we recommend adopting a reasonable ϵ with good utility to achieve robustness, as elaborated in Section 6.2.1.

B.2.5 Additional Robustness Evaluation of User-level DPFL. Here we further explore the impacts of ϵ on the certified attack inefficacy. Similar to the results of label flipping attacks in Figure 7 (a-c), the results of backdoor attacks in Figure 12 show that as the privacy guarantee becomes stronger, i.e. smaller ϵ, the model is more robust, achieving higher $J(D')$ and $\underline{J(D')}$.

Table 9: Comparison of empirical robust accuracy between our certification approach and empirical FL defenses against state-of-the-art poisoning attacks on CIFAR. "UserDP-FedAvg-cert" denotes our certification approach based on UserDP-FedAvg. UserDP-FedAvg-cert provides similar or even higher empirical robust accuracy than empirical defenses. The certified accuracy of UserDP-FedAvg-cert serves as the lower bound for its empirical robust accuracy.

		k=2					k=3				
		Empirical Robust Acc.				Certified Robust Acc.	Empirical Robust Acc.				Certified Robust Acc.
	No Attack	STAT-OPT [67] (Min-Max)	STAT-OPT [67] (Min-Sum)	BKD [4] (γ = 100)	LF [8] (γ = 100)		STAT-OPT [67] (Min-Max)	STAT-OPT [67] (Min-Sum)	BKD [4] (γ = 100)	LF [8] (γ = 100)	
FedAvg [52]	**88.08**%	87.29%	87.35%	65.73%	65.47%	/	86.36%	86.55%	58.39%	58.07%	/
Median [83]	87.76%	87.09%	87.16%	87.73%	87.74%	/	86.22%	86.42%	87.74%	87.75%	/
Trimmed-mean [83]	**88.08**%	87.28%	87.35%	87.98%	87.98%	/	86.36%	86.55%	87.94%	87.94%	/
Krum [11]	85.97%	85.84%	85.96%	85.87%	85.87%	/	85.12%	85.4%	85.85%	85.85%	/
Multi-Krum [11]%	88.02%	87.23%	87.29%	87.99%	**87.99**%	/	86.31%	86.51%	**87.98**%	**87.98**%	/
Bulyan [23]	88.02%	87.24%	87.3%	87.93%	87.94%	/	86.31%	86.52%	87.89%	87.89%	/
RFA [61]	87.97%	87.21%	87.28%	87.94%	87.94%	/	86.29%	86.49%	87.96%	87.95%	/
UserDP-FedAvg-cert (ϵ = 0.7693)	88.05%	87.65%	**88**%	**88.05**%	87.8%	17.65%	**87.15**%	87.5%	87.8%	87.85%	1.4%
UserDP-FedAvg-cert (ϵ = 0.648)	87.35%	**87.8**%	87.6%	87.9%	87.5%	28.15%	86.45%	**87.6**%	87.2%	87.6%	4.3%
UserDP-FedAvg-cert (ϵ = 0.5346)	86.45%	86.5%	87%	87.15%	86.8%	37.75%	87.05%	86.65%	87.15%	87.15%	11.45%
UserDP-FedAvg-cert (ϵ = 0.3205)	85.2%	85.15%	86.05%	85.1%	85.7%	**48.5**%	83.9%	85.85%	85.8%	84.95%	**21.85**%

		k=5					k=10				
		Empirical Robust Acc.				Certified Robust Acc.	Empirical Robust Acc.				Certified Robust Acc.
	No Attack	STAT-OPT [67] (Min-Max)	STAT-OPT [67] (Min-Sum)	BKD [4] (γ = 100)	LF [8] (γ = 100)		STAT-OPT [67] (Min-Max)	STAT-OPT [67] (Min-Sum)	BKD [4] (γ = 100)	LF [8] (γ = 100)	
FedAvg [52]	**88.08**%	84.58%	85.75%	54.69%	54.35%	/	80.89%	84.52%	51.17%	51.21%	/
Median [83]	87.76%	84.5%	85.67%	87.69%	87.69%	/	80.86%	84.5%	87.56%	87.56%	/
Trimmed-mean [83]	**88.08**%	84.58%	85.75%	87.8%	87.8%	/	80.89%	84.52%	87.44%	87.43%	/
Krum [11]	85.97%	83.78%	85%	85.85%	85.85%	/	80.62%	84.29%	85.89%	85.88%	/
Multi-Krum [11]	88.02%	84.54%	85.72%	**87.94**%	**87.95**%	/	80.88%	84.52%	**87.92**%	**87.92**%	/
Bulyan [23]	88.02%	84.54%	85.72%	87.79%	87.79%	/	80.88%	84.52%	87.66%	87.66%	/
RFA [61]	87.97%	84.54%	85.71%	87.93%	87.93%	/	80.87%	84.51%	87.82%	87.82%	/
UserDP-FedAvg-cert (ϵ = 0.7693)	88.05%	**86.2**%	**86.35**%	87.4%	87.3%	0%	**85.25**%	**86.5**%	86.95%	86.75%	0%
UserDP-FedAvg-cert (ϵ = 0.648)	87.35%	**86.2**%	86.3%	87.15%	87.4%	0%	85.1%	85.75%	86.75%	85.85%	0%
UserDP-FedAvg-cert (ϵ = 0.5346)	86.45%	85.6%	86.1%	87.05%	87.1%	0%	84.65%	85.2%	86.65%	85.1%	0%
UserDP-FedAvg-cert (ϵ = 0.3205)	85.2%	83.4%	85.25%	84.5%	85.35%	0.35%	82.35%	84.95%	84.2%	85.6%	0%

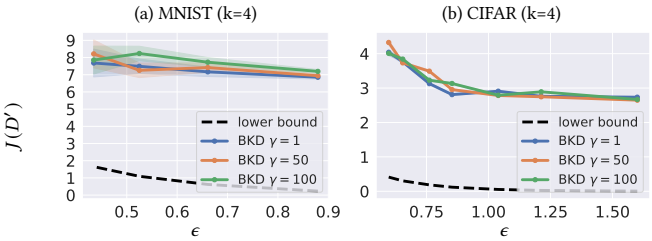

Figure 12: Certified attack inefficacy of UserDP-FedAvg with different ϵ under backdoor attack.

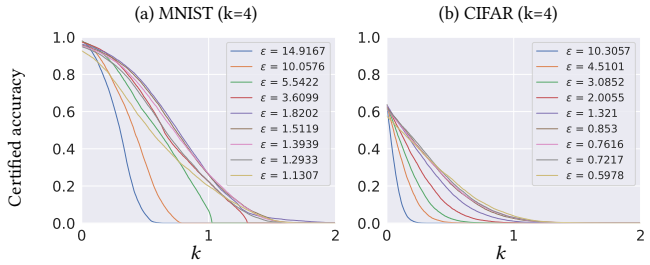

Figure 13: Certified accuracy of FL UserDP-FedAvg on 10-class classification.

B.2.6 Robustness Evaluation on 10-class Classification. Here we report the robustness evaluation of user-level DPFL under backdoor attacks on 10-class classification problems. Figure 13 presents the certified accuracy under different ϵ. We observe the interplay between ϵ and certified accuracy on MNIST. On CIFAR, larger k can be certified with smaller ϵ. The certified K is relatively small because we set large ϵ to preserve a reasonable accuracy for 10-class classification. Our results suggest that advanced DP mechanisms would be preferred to provide tighter privacy guarantees (i.e., smaller ϵ) while achieving a similar level of accuracy. In terms of certified attack inefficacy, as shown in Figure 14 and Figure 15, the trends are similar to the 2-class results in Figure 7 and Figure 4,

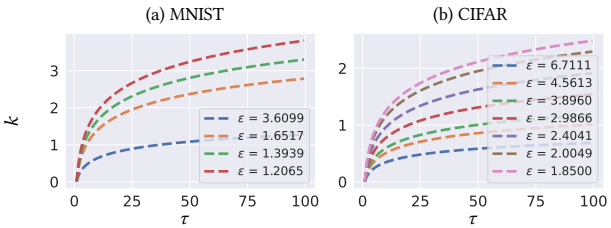

Figure 14: Lower bound of k on 10-class classification under user-level ϵ given attack effectiveness τ.

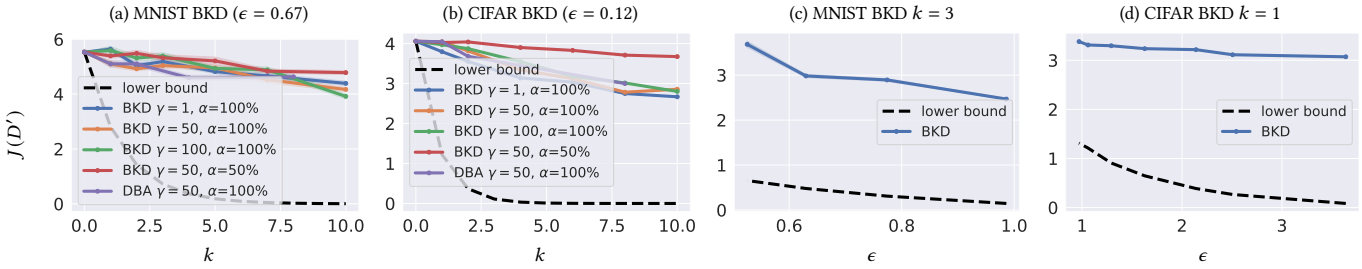

Figure 15: Certified attack inefficacy of `UserDP-FedAvg` on 10-class classification given the different number of malicious instances k (a)(b) and different ϵ (c)(d).

C PROOFS OF CERTIFIED ROBUSTNESS ANALYSIS

We restate our Definition 2 here.

Definition 2 (Group DP). *For mechanism $\mathcal{M}$ that satisfies (ϵ, δ)-DP, it satisfies $(k\epsilon, \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta)$-DP for groups of size k. That is, for any $d, d' \in \mathcal{D}$ that differ by k individuals and any $E \subseteq \Theta$, it holds that*

$$\Pr[\mathcal{M}(d) \in E] \leq e^{k\epsilon} \Pr\left[\mathcal{M}\left(d'\right) \in E\right] + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta. \tag{2}$$

Proof. We denote d as d_0, d' as d_k. d_i differ i individuals with d_0. For any $i \in [1, k]$, d_i and d_{i-1} differ by one individual, thus

$$\Pr[M(d_{i-1})] \leq e^{\epsilon} \Pr[M(d_i)] + \delta. \tag{9}$$

By iteratively applying Eq. (9) k times, we have

$$\begin{aligned}\Pr[M(d_0)] &\leq e^{k\epsilon} \Pr[M(d_k)] + (1 + e^{\epsilon} + e^{2\epsilon} + \ldots + e^{(k-1)\epsilon})\delta \\ &= e^{k\epsilon} \Pr[M(d_k)] + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\end{aligned}$$

□

Before we prove Theorem 1, we introduce the following lemma:

Lemma 2. *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level (ϵ, δ)-DP. For two user sets B and B' that differ by one user, D and D' are the corresponding training datasets. For a test input x, for any $c \in [C]$, $f_c(\mathcal{M}(D), x) \in [0, 1]$ is the class confidence, then the expected class confidence $F_c(\mathcal{M}(D), x) := \mathbb{E}[f_c(\mathcal{M}(D), x)]$ meets the following property:*

$$F_c(\mathcal{M}(D), x) \leq e^{\epsilon} F_c(\mathcal{M}(D'), x) + \delta \tag{10}$$

Proof. Define $\Theta(a) := \{\theta : f_c(\theta, x) > a\}$. Then

$$\begin{aligned}F_c(\mathcal{M}(D), x) &= \mathbb{E}[f_c(\mathcal{M}(D), x)] \\ &= \int_0^1 \mathbb{P}\left[f_c(\mathcal{M}(D), x) > a\right] da \\ &= \int_0^1 \mathbb{P}\left[\mathcal{M}(D) \in \Theta(a)\right] da \\ &\leq \int_0^1 \left(e^{\epsilon}\mathbb{P}\left[\mathcal{M}(D') \in \Theta(a)\right] + \delta\right) da \\ &= \int_0^1 e^{\epsilon}\mathbb{P}\left[f_c(\mathcal{M}(D'), x) > a\right] da + \int_0^1 \delta da \\ &= e^{\epsilon} F_c(\mathcal{M}(D'), x) + \delta\end{aligned}$$

□

We recall Theorem 1.

Theorem 1 (Certified Prediction under One Adversarial User)**.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level (ϵ, δ)-DP. For two user sets B and B' that differ by one user, let D and D' be the corresponding training datasets. For a test input x, suppose $\mathbb{A}, \mathbb{B} \in [C]$ satisfy $\mathbb{A} = \arg\max_{c \in [C]} F_c(\mathcal{M}(D), x)$ and $\mathbb{B} = \arg\max_{c \in [C]: c \neq \mathbb{A}} F_c(\mathcal{M}(D), x)$. Then, it is guaranteed that $H(\mathcal{M}(D'), x) = H(\mathcal{M}(D), x) = \mathbb{A}$ if:*

$$F_{\mathbb{A}}(\mathcal{M}(D), x) > e^{2\epsilon} F_{\mathbb{B}}(\mathcal{M}(D), x) + (1 + e^{\epsilon})\delta, \tag{5}$$

Proof. According to Lemma 2,

$$F_{\mathbb{A}}(\mathcal{M}(D), x) \leq e^{\epsilon} F_{\mathbb{A}}(\mathcal{M}(D'), x) + \delta \tag{11}$$

$$F_{\mathbb{B}}(\mathcal{M}(D'), x) \leq e^{\epsilon} F_{\mathbb{B}}(\mathcal{M}(D), x) + \delta. \tag{12}$$

Then

$$\begin{aligned}F_{\mathbb{A}}(\mathcal{M}(D'), x) &\geq \frac{F_{\mathbb{A}}(\mathcal{M}(D), x) - \delta}{e^{\epsilon}} && \text{(Because of Eq. 11)} \\ &\geq \frac{e^{2\epsilon} F_{\mathbb{B}}(\mathcal{M}(D), x) + (1 + e^{\epsilon})\delta - \delta}{e^{\epsilon}} \\ &&& \text{(Because of the given condition Eq. 5)} \\ &= e^{\epsilon} F_{\mathbb{B}}(\mathcal{M}(D), x) + \delta \\ &\geq e^{\epsilon}\left(\frac{F_{\mathbb{B}}(\mathcal{M}(D'), x) - \delta}{e^{\epsilon}}\right) + \delta \\ &&& \text{(Because of Eq. 12)} \\ &= F_{\mathbb{B}}(\mathcal{M}(D'), x),\end{aligned}$$

which indicates that the prediction of $\mathcal{M}(D')$ at x is $\mathbb{A}$ by definition.

□

Before we prove Theorem 2, we introduce the following lemma:

Lemma 3. *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level (ϵ, δ)-DP. For two user sets B and B' that differ k users, D and D' are the corresponding training datasets. For a test input x, for any $c \in [C]$, $f_c(\mathcal{M}(D), x) \in [0, 1]$ is the class confidence, then the expected class confidence $F_c(\mathcal{M}(D), x) := \mathbb{E}[f_c(\mathcal{M}(D), x)]$ meets the following property:*

$$F_c(\mathcal{M}(D), x) \leq e^{k\epsilon} F_c(\mathcal{M}(D'), x) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta. \tag{13}$$

and

$$F_c(\mathcal{M}(D'), x) \leq e^{k\epsilon} F_c(\mathcal{M}(D), x) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta.$$

Proof. Define $\Theta(a) := \{\theta : f_c(\theta, x) > a\}$. Then

$$\begin{aligned}
F_c(\mathcal{M}(D), x) &= \int_0^1 \mathbb{P}\left[f_c(\mathcal{M}(D), x) > a\right] da \\
&= \int_0^1 \mathbb{P}\left[\mathcal{M}(D) \in \Theta(a)\right] da \\
&\le \int_0^1 \left(e^{k\epsilon}\mathbb{P}\left[\mathcal{M}(D') \in \Theta(a)\right] + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\right) da \\
&\qquad \text{(Because of Group DP property in Definition 2)} \\
&= \int_0^1 e^{k\epsilon}\mathbb{P}\left[f_c(\mathcal{M}(D'), x) > a\right] da + \int_0^1 \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta da \\
&= e^{k\epsilon}F_c(\mathcal{M}(D'), x) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta
\end{aligned}$$

Similarly, due to the symmetric property of adjacent datasets in the DP definition (Definition 1) and Group DP definition (Definition 2), D and D' are interchangeable, and therefore we have

$$\begin{aligned}
F_c(\mathcal{M}(D'), x) &= \int_0^1 \mathbb{P}\left[f_c(\mathcal{M}(D'), x) > a\right] da \\
&= \int_0^1 \mathbb{P}\left[\mathcal{M}(D') \in \Theta(a)\right] da \\
&\le \int_0^1 \left(e^{k\epsilon}\mathbb{P}\left[\mathcal{M}(D) \in \Theta(a)\right] + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\right) da \\
&\qquad \text{(Because of Group DP property in Definition 2)} \\
&= \int_0^1 e^{k\epsilon}\mathbb{P}\left[f_c(\mathcal{M}(D), x) > a\right] da + \int_0^1 \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta da \\
&= e^{k\epsilon}F_c(\mathcal{M}(D), x) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta
\end{aligned}$$

□

We recall Theorem 2.

Theorem 2 (Upper Bound of k for Certified Prediction)**.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level (ϵ, δ)-DP. For two user sets B and B' that differ by k users, let D and D' be the corresponding training datasets. For a test input x, suppose $\mathbb{A}, \mathbb{B} \in [C]$ satisfy $\mathbb{A} = \arg\max_{c\in[C]} F_c(\mathcal{M}(D), x)$ and $\mathbb{B} = \arg\max_{c\in[C]:c\neq\mathbb{A}} F_c(\mathcal{M}(D), x)$, then $H(\mathcal{M}(D'), x) = H(\mathcal{M}(D), x) = \mathbb{A}, \forall k < \mathsf{K}$ where K is the certified number of adversarial users:*

$$\mathsf{K} = \frac{1}{2\epsilon}\log\frac{F_{\mathbb{A}}(\mathcal{M}(D), x)(e^{\epsilon}-1)+\delta}{F_{\mathbb{B}}(\mathcal{M}(D), x)(e^{\epsilon}-1)+\delta} \tag{6}$$

Proof. According to Lemma 3, we have

$$F_{\mathbb{A}}(\mathcal{M}(D), x) \le e^{k\epsilon}F_{\mathbb{A}}(\mathcal{M}(D'), x) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta \tag{14}$$

$$F_{\mathbb{B}}(\mathcal{M}(D'), x) \le e^{k\epsilon}F_{\mathbb{B}}(\mathcal{M}(D), x) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta. \tag{15}$$

We can re-write the given condition $k < \mathsf{K}$ according to Eq. (6) as

$$e^{2k\epsilon}F_{\mathbb{B}}(\mathcal{M}(D), x) + (1+e^{k\epsilon})\frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta < F_{\mathbb{A}}(\mathcal{M}(D), x). \tag{16}$$

Then

$$\begin{aligned}
F_{\mathbb{A}}(\mathcal{M}(D'), x) &\ge \frac{F_{\mathbb{A}}(\mathcal{M}(D), x) - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta}{e^{k\epsilon}} \qquad \text{(Because of Eq. 14)} \\
&> \frac{e^{2k\epsilon}F_{\mathbb{B}}(\mathcal{M}(D), x) + (1+e^{k\epsilon})\frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta}{e^{k\epsilon}} \\
&\qquad \text{(Because of the given condition Eq.16)} \\
&= e^{k\epsilon}F_{\mathbb{B}}(\mathcal{M}(D), x) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta \\
&\ge e^{k\epsilon}\left(\frac{F_{\mathbb{B}}(\mathcal{M}(D'), x) - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta}{e^{k\epsilon}}\right) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta \\
&\qquad \text{(Because of Eq. 15)} \\
&= F_{\mathbb{B}}(\mathcal{M}(D'), x),
\end{aligned}$$

which indicates that the prediction of $\mathcal{M}(D')$ at x is $\mathbb{A}$ by definition. □

We recall Theorem 3.

Theorem 3 (Attack Inefficacy with k Attackers)**.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level (ϵ, δ)-DP. For two user sets B and B' that differ k users, D and D' are the corresponding training datasets. Let $J(D)$ be the expected attack inefficacy where $|C(\theta)| \le \bar{C}, \forall\theta$. Then,*

$$\begin{aligned}
&\min\{e^{k\epsilon}J(D) + \frac{e^{k\epsilon}-1}{e^{\epsilon}-1}\delta\bar{C}, \bar{C}\} \ge J(D') \\
&\ge \max\{e^{-k\epsilon}J(D) - \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}, 0\}, \quad \textit{if} \quad C(\cdot) \ge 0 \\
&\min\{e^{-k\epsilon}J(D) + \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}, 0\} \ge J(D') \\
&\ge \max\{e^{k\epsilon}J(D) - \frac{e^{k\epsilon}-1}{e^{\epsilon}-1}\delta\bar{C}, -\bar{C}\}, \quad \textit{if} \quad C(\cdot) \le 0
\end{aligned} \tag{7}$$

Proof. We first consider $C(\cdot) \ge 0$. Define $\Theta(a) = \{\theta : C(\theta) > a\}$.

$$\begin{aligned}
J(D) &= \int_0^{\bar{C}} \mathbb{P}\left[C(\mathcal{M}(D)) > a\right] da \\
&= \int_0^{\bar{C}} \mathbb{P}\left[\mathcal{M}(D)) \in \Theta(a)\right] da \\
&\le \int_0^{\bar{C}} \left(e^{k\epsilon}\mathbb{P}\left[\mathcal{M}(D')) \in \Theta(a)\right] + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\right) da \\
&\qquad \text{(Because of Group DP property in Definition 2)} \\
&= \int_0^{\bar{C}} e^{k\epsilon}\mathbb{P}\left[\mathcal{M}(D')) \in \Theta(a)\right] da + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C} \\
&= \int_0^{\bar{C}} e^{k\epsilon}\mathbb{P}\left[C(\mathcal{M}(D')) > a\right] da + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C} \\
&= e^{k\epsilon}J(D') + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C}
\end{aligned}$$

i.e.,

$$J(D') \ge e^{-k\epsilon}J(D) - \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}.$$

Switch the role of D and D', we have

$$J(D') \le e^{k\epsilon}J(D) + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C}.$$

Also note that $0 \le J(D') \le \bar{C}$ trivially holds due to $0 \le C(\cdot) \le \bar{C}$, thus

$$\min\{e^{k\epsilon}J(D) + \frac{e^{k\epsilon}-1}{e^{\epsilon}-1}\delta\bar{C}, \bar{C}\} \ge J(D') \ge \max\{e^{-k\epsilon}J(D) - \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}, 0\}.$$

Next, we consider $C(\cdot) \le 0$. Define $\Theta(a) = \{\theta : C(\theta) < a\}$.

$$\begin{aligned}
J(D) &= -\int_{-\bar{C}}^{0} \mathbb{P}\left[C(\mathcal{M}(D)) < a\right] da \\
&= -\int_{-\bar{C}}^{0} \mathbb{P}\left[\mathcal{M}(D)) \in \Theta(a)\right] da \\
&\ge -\int_{-\bar{C}}^{0} \left(e^{k\epsilon}\mathbb{P}\left[\mathcal{M}(D')) \in \Theta(a)\right] + \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\right) da \\
&\qquad \text{(Because of Group DP property in Definition 2)} \\
&= -\int_{-\bar{C}}^{0} e^{k\epsilon}\mathbb{P}\left[\mathcal{M}(D')) \in \Theta(a)\right] da - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C} \\
&= -\int_{-\bar{C}}^{0} e^{k\epsilon}\mathbb{P}\left[C(\mathcal{M}(D')) < a\right] da - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C} \\
&= e^{k\epsilon}J(D') - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C}
\end{aligned}$$

i.e.,

$$J(D') \le e^{-k\epsilon}J(D) + \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}.$$

Switch the role of D and D', we have

$$J(D') \ge e^{k\epsilon}J(D) - \frac{1-e^{k\epsilon}}{1-e^{\epsilon}}\delta\bar{C}.$$

Also note that $-\bar{C} \le J(D') \le 0$ trivially holds due to $-\bar{C} \le C(\cdot) \le 0$, thus

$$\min\{e^{-k\epsilon}J(D) + \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}, 0\} \ge J(D') \ge \max\{e^{k\epsilon}J(D) - \frac{e^{k\epsilon}-1}{e^{\epsilon}-1}\delta\bar{C}, -\bar{C}\}$$

□

We recall Corollary 1.

Corollary 1 (Lower Bound of k Given τ, extended from [50])**.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level (ϵ, δ)-DP. Let attack inefficacy function be $C(\cdot)$, the expected attack inefficacy be $J(\cdot)$. In order to achieve $J(D') \le \frac{1}{\tau}J(D)$ for $\tau \ge 1$ when $0 \le C(\cdot) \le \bar{C}$, or achieve $J(D') \le \tau J(D)$ for $1 \le \tau \le -\frac{\bar{C}}{J(D)}$ when $-\bar{C} \le C(\cdot) \le 0$, the number of adversarial users should satisfy the following:*

$$k \ge \frac{1}{\epsilon}\log\frac{(e^{\epsilon}-1)J(D)\tau + \bar{C}\delta\tau}{(e^{\epsilon}-1)J(D) + \bar{C}\delta\tau} \quad \text{or } k \ge \frac{1}{\epsilon}\log\frac{(e^{\epsilon}-1)J(D)\tau - \bar{C}\delta}{(e^{\epsilon}-1)J(D) - \bar{C}\delta},$$

Proof. We first consider $C(\cdot) \ge 0$. According to the lower bound in Theorem 3, when B' and B differ k users, $J(D') \ge e^{-k\epsilon}J(D) - \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C}$. Since we require $J(D') \le \frac{1}{\tau}J(D)$, then $e^{-k\epsilon}J(D) - \frac{1-e^{-k\epsilon}}{e^{\epsilon}-1}\delta\bar{C} \le \frac{1}{\tau}J(D)$. Rearranging gives the result.

Next, we consider $C(\cdot) \le 0$. According to the lower bound in Theorem 3, when B' and B differ k users, $J(D') \ge e^{k\epsilon}J(D) - \frac{e^{k\epsilon}-1}{e^{\epsilon}-1}\delta\bar{C}$. Since we require $J(D') \le \tau J(D)$, then $e^{k\epsilon}J(D) - \frac{e^{k\epsilon}-1}{e^{\epsilon}-1}\delta\bar{C} \le \tau J(D)$. Rearranging gives the result.

□

We note that all the above robustness certification-related proofs are built upon the user-level (ϵ, δ)-DP property and the Group DP property. According to Definition 3 and Definition 4, the definition of user-level DP and instance-level DP are both induced from DP (Definition 1) despite the different definitions of adjacent datasets. By applying the definition of instance-level (ϵ, δ)-DP and following the proof steps of Theorem 1, 2, 3 and Corollary 1, we can derive similar theoretical conclusions for instance-level DP, leading to Theorem 4 to achieve the certifiably robust FL given the DP property.

D CERTIFIED ROBUSTNESS ANALYSIS VIA RÉNYI DP AND RANDOMIZED SMOOTHING

D.1 Preliminary

We start by providing preliminaries on Rényi Differential Privacy [55] and the f-divergence-based randomized smoothing [20], which is a relaxation of ℓ_p-norm-based randomized smoothing [19].

Definition 5. *(Rényi Divergence) For two probability distributions ρ and ν, the Rényi divergence of order $\alpha > 1$ is*

$$D_{\alpha}(\rho\|\nu) \triangleq \frac{1}{\alpha-1}\log \mathrm{E}_{x\sim\nu}\left(\frac{\rho(x)}{\nu(x)}\right)^{\alpha} \tag{17}$$

Definition 6. *($(\alpha, \epsilon_{R,\alpha})$-RDP [55]) A randomized mechanism $\mathcal{M} : \mathcal{D} \to \Theta$ with domain $\mathcal{D}$ and output set Θ satisfies $(\alpha, \epsilon_{R,\alpha})$ Rényi Differential Privacy (RDP) if for any pair of two adjacent datasets $d, d' \in \mathcal{D}$, it holds that*

$$D_{\alpha}(\mathcal{M}(d)\|\mathcal{M}(d')) \le \epsilon_{R,\alpha} \tag{18}$$

Definition 7. *(Group Rényi DP [55]) For mechanism $\mathcal{M}$ that satisfies $(\alpha, \epsilon_{R,\alpha})$-RDP, it satisfies $(\alpha/2^k, 3^k\epsilon_{R,\alpha})$-DP for groups of size k. That is, for any $d, d' \in \mathcal{D}$ that differ by k individuals, it holds that*

$$D_{\alpha/2^k}(\mathcal{M}(d)\|\mathcal{M}(d')) \le 3^k\epsilon_{R,\alpha} \tag{19}$$

Lemma 4. *(Rényi DP and DP conversion [55]) The mechanism $\mathcal{M}$ that satisfies $(\alpha, \epsilon_{R,\alpha})$-RDP $\alpha > 1$, also satisfies $(\epsilon_{R,\alpha} + \frac{\log 1/\delta}{\alpha-1}, \delta)$-DP for any $0 < \delta < 1$.*

Lemma 5. *(Certificates for Rényi-divergence [Table 4 of [20]]) Given two distributions ρ and ν with bounded Rényi divergences $(\alpha \ge 0)$ $D_{\alpha}(\rho\|\nu) \le \epsilon_{R,\alpha}$, and two probabilities p_a, p_b that satify $p_a, p_b \ge 0$, $p_a + p_b \le 1$, and define the class of specification S as*

$$S = \left\{\phi : \mathcal{X} \to \{-1, 0, +1\} \text{ s.t. } \mathop{\mathbb{P}}_{x\sim\rho}[\phi(x) = +1] \ge p_a, \mathop{\mathbb{P}}_{x\sim\rho}[\phi(x) = -1] \le p_b\right\}. \tag{20}$$

It is certified that $\mathbb{E}_{x\sim\nu}[\phi(x)] \ge 0$ for all ν and $\phi \in S$ if

$$\epsilon_{R,\alpha} \le -\log\left(1 - p_a - p_b + 2\eta\right), \text{ with } \quad \eta = \left(\frac{p_a^{(1-\alpha)} + p_b^{(1-\alpha)}}{2}\right)^{\left(\frac{1}{1-\alpha}\right)}.$$

D.2 Main Results on RDP-based Certified Prediction

We present our main results for certified robustness against FL poisoning attacks based on Rényi DP (RDP) [55] and Randomized Smoothing via Rényi Divergence [20]. Theorem 5 states the certification under one adversarial user and Theorem 6 further extends the certification to k adversarial suers.

Theorem 5 (RDP-based Certified Prediction under One Adversarial User)**.** *Suppose a randomized mechanism $\mathcal{M}$ satisfies user-level $(\alpha, \epsilon_{R,\alpha})$-RDP, which also satisfies user-level $(\epsilon_{R,\alpha} + \frac{\log 1/\delta}{\alpha-1}, \delta)$-DP, where $\alpha > 1$ and $0 < \delta < 1$. For two user sets B and B' that differ by one user, let D and D' be the corresponding training datasets. Define the classifier as $h : (\theta, \mathbb{R}^d) \to [C]$ with the finite set of labels $[C]$, and the randomly smoothed classifier h_s as $h_s(\mathcal{M}(D), x) := \arg\max_{c\in[C]} \mathbb{P}[h(\mathcal{M}(D), x) = c]$. For a test input x, suppose that*

$$\mathbb{P}[h(\mathcal{M}(D), x) = \mathbb{A}] \geq p_a \geq p_b \geq \arg\max_{c\in[C]:c\neq\mathbb{A}} \mathbb{P}[h(\mathcal{M}(D), x) = c].$$

Then, it is guaranteed that $h_s(\mathcal{M}(D'), x) = h_s(\mathcal{M}(D), x) = \mathbb{A}$ if:

$$\epsilon_{R,\alpha} \leq -\log\left(1 - p_a - p_b + 2\left(\frac{p_a^{(1-\alpha)} + p_b^{(1-\alpha)}}{2}\right)^{\left(\frac{1}{1-\alpha}\right)}\right).$$

Theorem 6 (RDP-based Certified Prediction under k Adversarial User)**.** *Using the same setting as in Theorem 5 but let two user sets B and B' differ by k users, and D and D' be the corresponding training datasets. Then, it is guaranteed that $h_s(\mathcal{M}(D'), x) = h_s(\mathcal{M}(D), x) = \mathbb{A}$ if:*

$$\epsilon_{R,\alpha} \leq -\frac{1}{3^k}\log\left(1 - p_a - p_b + 2\left(\frac{p_a^{(1-\alpha/2^k)} + p_b^{(1-\alpha/2^k)}}{2}\right)^{\left(\frac{1}{1-\alpha/2^k}\right)}\right).$$

Remark. From Theorem 5 and Theorem 6, we observe that **(1)** RDP-based certifications are more complex than DP-based certifications due to the additional tunable parameter, the RDP order α, and its foundational Rényi Divergence-based privacy definition. **(2)** Theorem 6 presents a more intricate RHS, making it challenging to derive a simple closed-form upper bound K for the certified number of attackers where $k < \mathsf{K}$, as seen in Theorem 2. Nevertheless, Theorem 6 can be utilized to perform a binary check for certified robustness by verifying if the current RDP privacy budget satisfies the inequality. **(3)** Different from DP-based certifications in Theorem 1 and Theorem 2 that are built upon the *expected class confidence* F_A and F_B, RDP-based certifications are built upon the *probability of model prediction*, e.g., the probability of the model predicting a certain class $\mathbb{P}[h(\mathcal{M}(D), x) = \mathbb{A}]$, where $h(\mathcal{M}(D), x)$ is the predicted class. To compute RDP-based certifications in practice, one can also use Marto Carlo sampling to approximate $\mathbb{P}[h(\mathcal{M}(D), x) = \mathbb{A}]$.

D.3 Proofs

We now provide the proofs for Theorem 5 and Theorem 6 below.

Proof for Theorem 5. Recall that we define the classifer $h : (\theta, \mathbb{R}^d) \to [C]$ with the finite set of labels $[C]$, and the randomly smoothed classifer h_s as

$$h_s(\mathcal{M}(D), x) := \arg\max_{c\in[C]} \mathbb{P}[h(\mathcal{M}(D), x) = c], \tag{21}$$

where x is a test sample, $\mathcal{M}(D)$ is the stochastic model trained from the randomized DP mechanism $\mathcal{M}$ on a training dataset D.

For a test input x, suppose that

$$\mathbb{P}[h(\mathcal{M}(D), x) = \mathbb{A}] \geq p_a \geq p_b \geq \arg\max_{c\in[C]:c\neq\mathbb{A}} \mathbb{P}[h(\mathcal{M}(D), x) = c].$$

Therefore, $\mathbb{A} = h_s(\mathcal{M}(D), x)$.

Let $\mathbb{B} = \arg\max_{c\in[C]:c\neq\mathbb{A}} \mathbb{P}[h(\mathcal{M}(D), x) = c]$. We define the specification $\phi_{\mathbb{A},\mathbb{B}}$ as follows:

$$\phi_{\mathbb{A},\mathbb{B}}(\mathcal{M}(D)) = \begin{cases} +1 & \text{if } h(\mathcal{M}(D), x) = \mathbb{A} \\ -1 & \text{if } h(\mathcal{M}(D), x) = \mathbb{B} \\ 0 & \text{otherwise} \end{cases} \tag{22}$$

Based on the certificates for Rényi-divergence in Lemma 5 and Definition 6 for Rényi DP, if

$$\epsilon_{R,\alpha} \leq -\log\left(1 - p_a - p_b + 2\eta\right), \text{with} \quad \eta = \left(\frac{p_a^{(1-\alpha)} + p_b^{(1-\alpha)}}{2}\right)^{\left(\frac{1}{1-\alpha}\right)},$$

and if the mechanism $\mathcal{M}$ satisfies $(\alpha, \epsilon_{R,\alpha})$-RDP ($\alpha > 1$)

$$D_\alpha(\mathcal{M}(D)\|\mathcal{M}(D')) \leq \epsilon_{R,\alpha}, \tag{23}$$

it is certified that $\mathbb{E}[\phi_{\mathbb{A},\mathbb{B}}(\mathcal{M}(D'))] = \mathbb{P}[h(\mathcal{M}(D'), x) = \mathbb{A}] - \mathbb{P}[h(\mathcal{M}(D'), x) = \mathbb{B}] \geq 0$, that is,

$$\mathbb{P}[h(\mathcal{M}(D'), x) = \mathbb{A}] \geq \mathbb{P}[h(\mathcal{M}(D'), x) = \mathbb{B}].$$

It further implies that

$$h_s(\mathcal{M}(D'), x) = h_s(\mathcal{M}(D), x) = \mathbb{A}.$$

Finally, we can convert Rényi DP to DP by Lemma 4 □

Proof for Theorem 6. According to the group Rényi DP in Definition 7, the mechanism $\mathcal{M}$ that satifies user-level $(\alpha, \epsilon_{R,\alpha})$-RDP also satifies user-level $(\alpha/2^k, 3^k\epsilon_{R,\alpha})$-RDP for two user sets B and B' that differ by k users. That is,

$$D_{\alpha/2^k}(\mathcal{M}(D)\|\mathcal{M}(D')) \leq 3^k\epsilon_{R,\alpha}. \tag{24}$$

For a test input x, suppose that

$$\mathbb{P}[h(\mathcal{M}(D), x) = \mathbb{A}] \geq p_a \geq p_b \geq \arg\max_{c\in[C]:c\neq\mathbb{A}} \mathbb{P}[h(\mathcal{M}(D), x) = c].$$

Then, according to Lemma 5 and following similar steps in the proofs of the Theorem 5, if

$$3^k\epsilon_{R,\alpha} \leq -\log\left(1 - p_a - p_b + 2\eta\right),$$
$$\text{with} \quad \eta = \left(\frac{p_a^{(1-\alpha/2^k)} + p_b^{(1-\alpha/2^k)}}{2}\right)^{\left(\frac{1}{1-\alpha/2^k}\right)},$$

it is certified that

$$h_s(\mathcal{M}(D'), x) = h_s(\mathcal{M}(D), x) = \mathbb{A}.$$

□