The 37th International Conference on Machine Learning (ICML'20), Jul 12th - 18th, 2020.

Scalable Differential Privacy with Certified Robustness in Adversarial Learning

NhatHai Phan[1], My T. Thai[2], Han Hu[1], Ruoming Jin[3], Tong Sun[4], and Dejing Dou[5]

[1] Ying Wu College of Computing, New Jersey Institute of Technology
[2] Department of Computer & Information Sciences & Engineering, University of Florida
[3] Computer Science Department, Kent State University
[4] Adobe Research Lab
[5] Computer and Information Science Department, University of Oregon

Email: phan@njit.edu

Outline

- Motivation and Background
- Differential Privacy (DP) in Adversarial Learning
- Composition of Certified Robustness
- Stochastic Batch Training (StoBatch)
- Experimental Results and Conclusion

Motivation

- DNNs are vulnerable to both privacy attacks and adversarial examples
- Existing efforts only focus on either preserving DP or deriving certified robustness, but not both DP and robustness!
  - private models are unshielded under adversarial examples
  - robust models (adversarial training) do not offer privacy protections to the training data
- Bounding the robustness of a model (protects data privacy and is robust against adversarial examples) at scale is nontrivial
  - adversarial examples introduces a previously unknown privacy risk
  - unrevealed interplay (trade-off) among DP preservation, adversarial learning, and robustness bounds

Goals

- Develop a novel mechanism (StoBatch) to: 1) preserve DP of the training data, 2) be provably and practically robust to adversarial examples, 3) retain high model utility, and 4) be scalable.

Methods

- Privacy-preserving (Laplace) noise is injected into inputs and hidden layers to achieve DP in learning private model parameters.
- The privacy noise p is projected on the scale of the robustness noise r.
  - a composition of certified robustness in both input and latent spaces
- Leverage the recipe of distributed adversarial training to develop a stochastic batch training
  - disjoint and fixed batches are distributed to local DP trainers

Results

- Established a connection among DP preservation to protect the training data, adversarial learning, and certified robustness.
- Derived a sequential composition robustness in both input and latent spaces.
- Addressed the trade-off among model utility, privacy loss, and robustness.
- Rigorous experiments shown that our mechanism significantly enhances the robustness and scalability of DP DNNs.

Deliverables

- Algorithms and models: https://github.com/haiphanNJIT/StoBatch

Differential Privacy

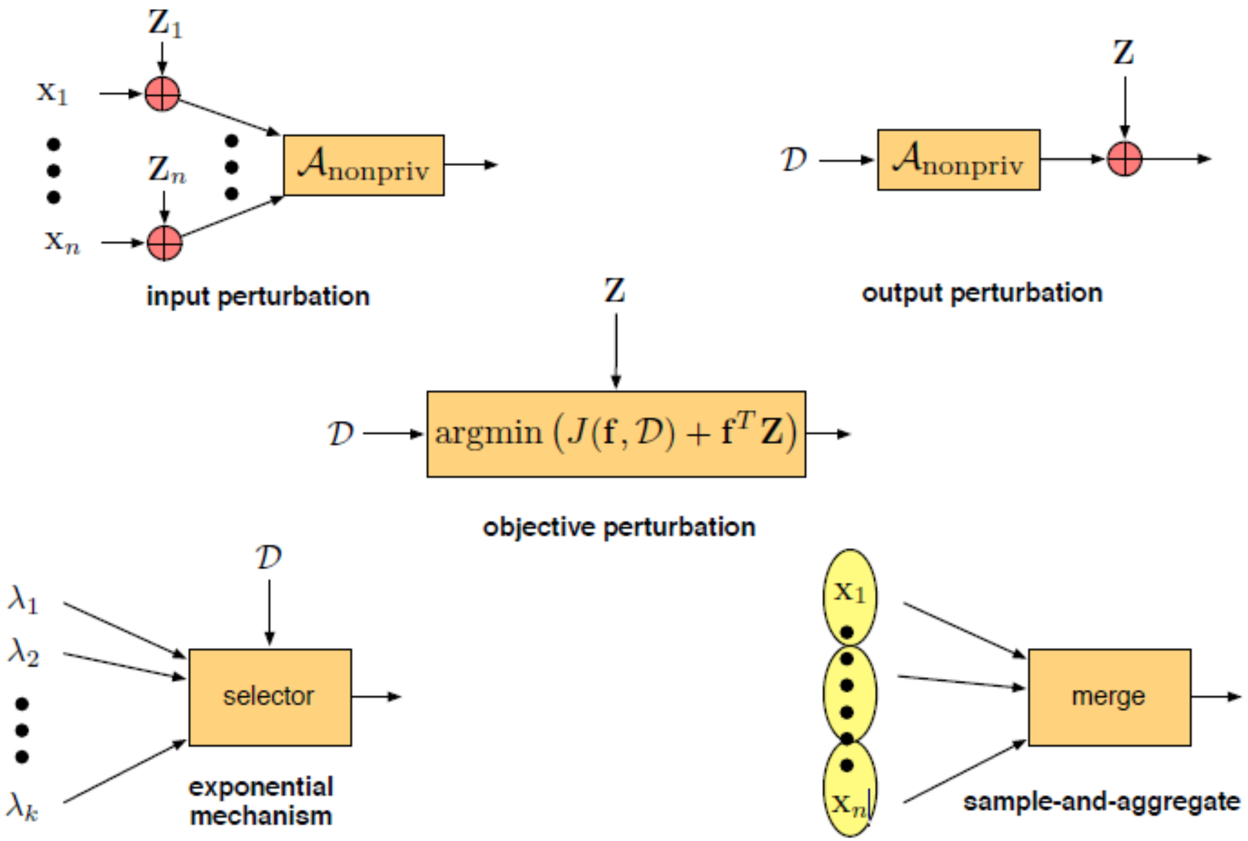
- Databases D and D' are neighbors if they are different in one individual's contribution
- (ϵ, δ)-Differential Privacy: for all D, D' neighbors, the distribution of $A(D)$ is (nearly) the same as the distribution of $A(D')$ for all $\mathbf{o}$:

$$Pr[A(D) = \mathbf{o}] \leq e^{\epsilon} Pr[A(D') = \mathbf{o}] + \delta$$

privacy loss

NJIT New Jersey Institute of Technology

DP Mechanisms



[Chaudhuri & Sarwate]

Robustness Condition [Lécuyer et al., 2019]

$$\forall \alpha \in l_p(\mu) \colon f_k(x+\alpha) > \max_{i:i \neq k} f_i\,(x+\alpha)$$

where $k = y(x)$, indicating that a small perturbation in the input does not change the predicted label $y(x)$.

NJIT New Jersey Institute of Technology

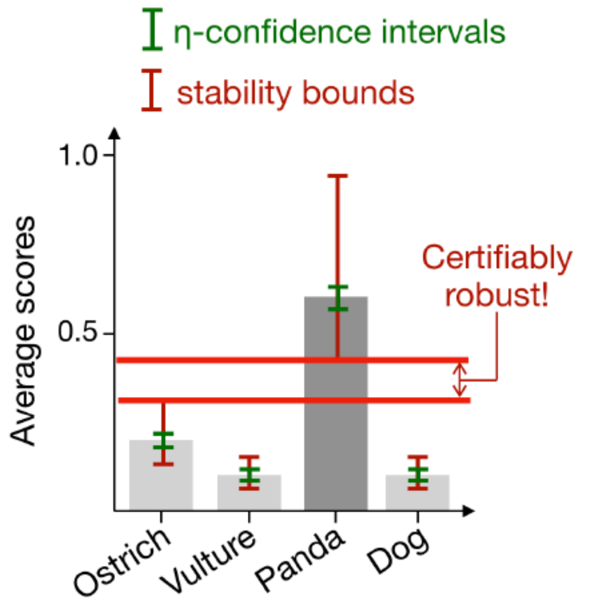
DP with Certified Robustness

[Lécuyer et al., 2019]

- Image level: $x = x + N(0, \sigma_r^2)$
  - $\sigma_r \geq \sqrt{2 \ln\left(\frac{1.25}{\delta_r}\right)} \Delta_r / \epsilon_r$

$$\forall \alpha \in l_p(\mu = 1) : \hat{\mathbb{E}}_{lb} f_k(x) > e^{2\epsilon_r} \max_{i: i \neq k} \hat{\mathbb{E}}_{ub} f_i(x) + (1 + e^{\epsilon_r})\delta_r$$

where $\hat{\mathbb{E}}_{lb}$ and $\hat{\mathbb{E}}_{ub}$ are the lower bound and upper bound of the expected value $\hat{\mathbb{E}} f(x) = \frac{1}{N} \sum_N f(x)_N$, derived from the Monte Carlo estimation with an η-confidence, given N is the number of invocations of $f(x)$ with independent draws in the noise σ_r.



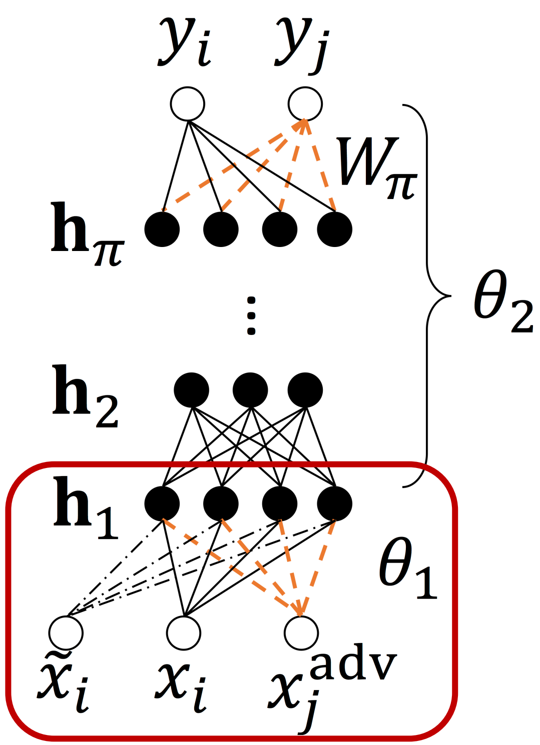
Robustness Test Example

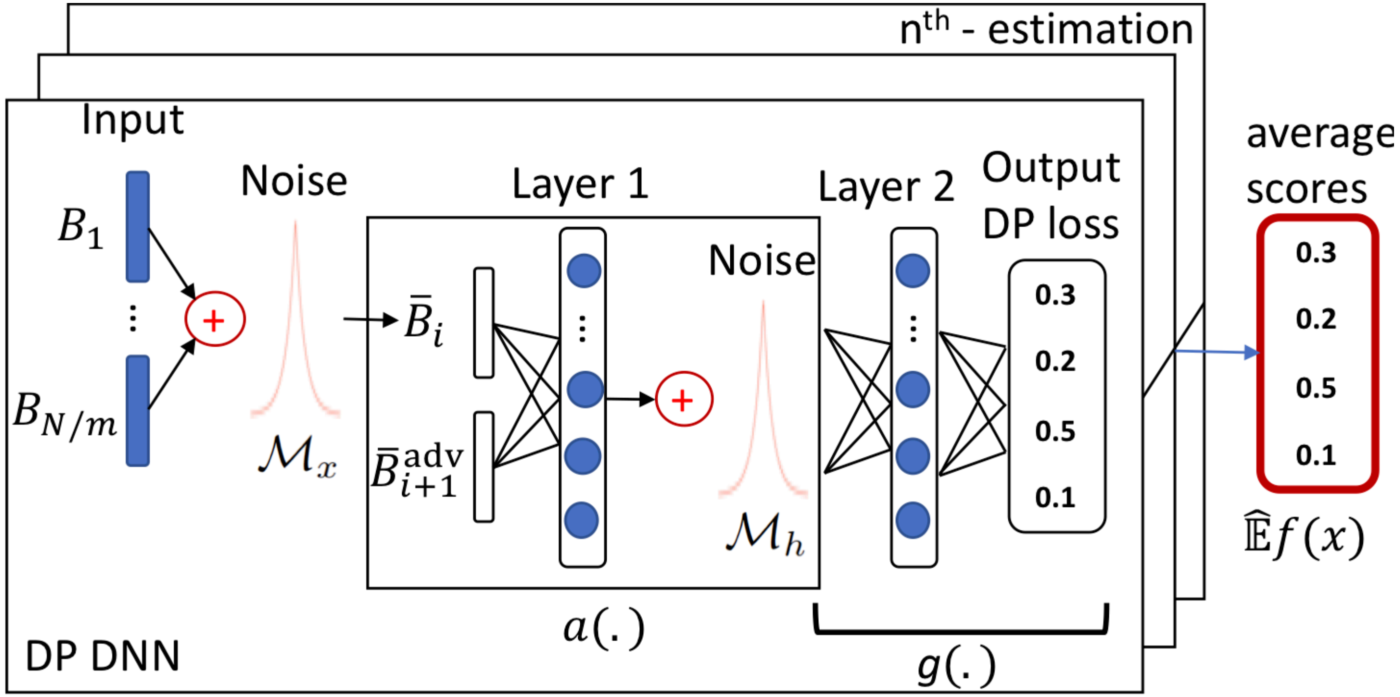
Outline

- Motivation and Background
- Differential Privacy in Adversarial Learning
- Composition of Certified Robustness
- Stochastic Batch Training (StoBatch)
- Experimental Results and Conclusion

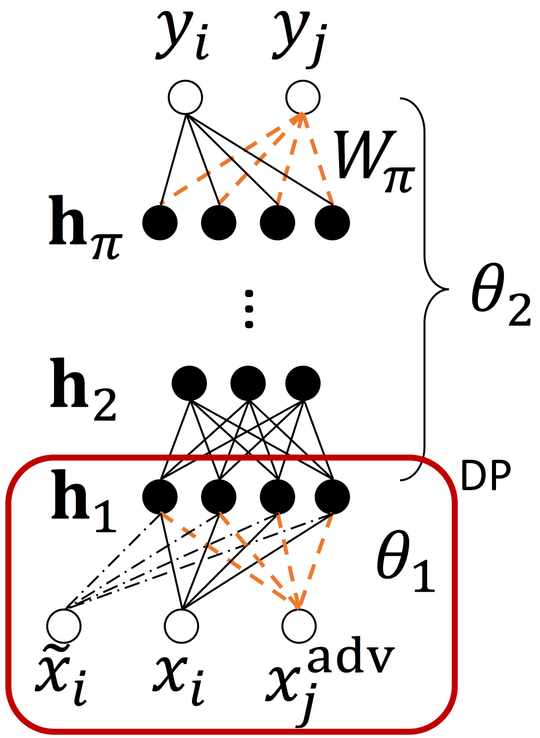
Differential Privacy in Adversarial Learning [Overview]

- $f(x) = g(a(x, \theta_1), \theta_2)$
  - easier to train, small sensitivity bounds, and reusability





DP Auto-Encoder



$$\bar{\mathcal{R}}_{\bar{B}_t}(\theta_1) = \sum_{x_i \in \bar{B}_t} \left[\sum_{j=1}^{d} \left(\frac{1}{2} \theta_{1j} \bar{h}_i \right) - \bar{x}_i \tilde{x}_i \right]$$

$$\bar{x}_i = x_i + \frac{1}{m} Lap\left(\frac{\Delta_{\mathcal{R}}}{\varepsilon_1}\right), \text{and } \bar{h}_i = \theta_1^T \bar{x}_i + \frac{2}{m} Lap\left(\frac{\Delta_{\mathcal{R}}}{\varepsilon_1}\right)$$

Theorem 1 *The gradient descent-based optimization of* $\overline{\mathcal{R}}_{\overline{B}_t}(\theta_1)$ *preserves* $(\epsilon_1/\gamma_{\mathbf{x}} + \epsilon_1)$*-DP in learning* θ_1.

Lemma 2 *The global sensitivity of* $\widetilde{\mathcal{R}}$ *over any two neighboring batches,* B_t *and* B'_t*, is:* $\Delta_{\mathcal{R}} \leq d(\beta + 2)$.

Adversarial Learning with DP

Lemma 3 *The computation of the batch $\overline{B}_t$ as the input layer is $(\epsilon_1/\gamma_{\mathbf{x}})$-DP, and the computation of the affine transformation $\overline{\mathbf{h}}_{1\overline{B}_t}$ is (ϵ_1/γ)-DP.*

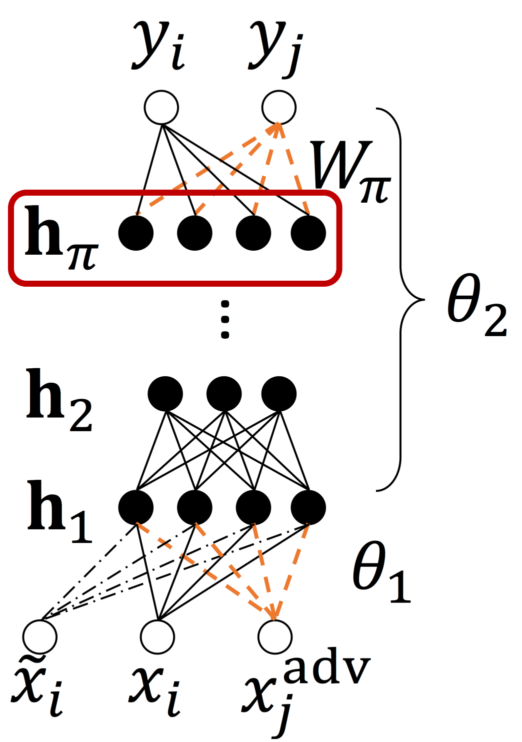
- DP Adversarial Examples

$$\overline{x}_j^{\text{adv}} = \overline{x}_j + \mu \cdot \text{sign}\Big(\nabla_{\overline{x}_j}\mathcal{L}\big(f(\overline{x}_j,\theta), y(\overline{x}_j)\big)\Big)$$

- DP Objective function

privacy leakage

as: $\mathcal{L}_{\overline{B}_t}(\theta_2) \approx \sum_{k=1}^{K}\sum_{\overline{x}_i}\big[\mathbf{h}_{\pi i}W_{\pi k} - (\mathbf{h}_{\pi i}W_{\pi k})y_{ik} - \frac{1}{2}|\mathbf{h}_{\pi i}W_{\pi k}| + \frac{1}{8}(\mathbf{h}_{\pi i}W_{\pi k})^2\big] \approx \mathcal{L}_{1\overline{B}_t}(\theta_2) - \mathcal{L}_{2\overline{B}_t}(\theta_2)$, where $\mathcal{L}_{1\overline{B}_t}(\theta_2) = \sum_{k=1}^{K}\sum_{\overline{x}_i}\big[\mathbf{h}_{\pi i}W_{\pi k} - \frac{1}{2}|\mathbf{h}_{\pi i}W_{\pi k}| + \frac{1}{8}(\mathbf{h}_{\pi i}W_{\pi k})^2\big]$, and $\mathcal{L}_{2\overline{B}_t}(\theta_2) = \sum_{k=1}^{K}\sum_{\overline{x}_i}(\mathbf{h}_{\pi i}y_{ik})W_{\pi k}$.



Algorithm 1 Adversarial Learning with DP

Input: Database D, loss function L, parameters θ, batch size m, learning rate ϱ_t, privacy budgets: ϵ_1 and ϵ_2, robustness parameters: ϵ_r, Δ_r^x, and Δ_r^h, adversarial attack size μ_a, the number of invocations n, ensemble attacks A, parameters ψ and ξ, and the size $|\mathbf{h}_\pi|$ of $\mathbf{h}_\pi$

1: **Draw Noise** $\chi_1 \leftarrow [Lap(\frac{\Delta_\mathcal{R}}{\epsilon_1})]^d$, $\chi_2 \leftarrow [Lap(\frac{\Delta_\mathcal{R}}{\epsilon_1})]^\beta$, $\chi_3 \leftarrow [Lap(\frac{\Delta_{\mathcal{L}2}}{\epsilon_2})]^{|\mathbf{h}_\pi|}$

2: **Randomly Initialize** $\theta = \{\theta_1, \theta_2\}$, $\mathbf{B} = \{B_1, \dots, B_{N/m}\}$ s.t. $\forall B \in \mathbf{B} : B$ is a batch with the size m, $B_1 \cap \dots \cap B_{N/m} = \emptyset$, and $B_1 \cup \dots \cup B_{N/m} = D$, $\overline{\mathbf{B}} = \{\overline{B}_1, \dots, \overline{B}_{N/m}\}$ where $\forall i \in [1, N/m]$: $\overline{B}_i = \{\overline{x} \leftarrow x + \frac{\chi_1}{m}\}_{x \in B_i}$

3: **Construct** a deep network f with **hidden layers** $\{\mathbf{h}_1 + \frac{2\chi_2}{m}, \dots, \mathbf{h}_\pi\}$, where $\mathbf{h}_\pi$ is the last hidden layer

4: **for** $t \in [T]$ **do**

5: **Take** a batch $\overline{B}_i \in \overline{\mathbf{B}}$ where $i = t\%(N/m)$, $\overline{B}_t \leftarrow \overline{B}_i$

6: **Ensemble DP Adversarial Examples:**

7: **Draw Random Perturbation Value** $\mu_t \in (0, 1]$

8: **Take** a batch $\overline{B}_{i+1} \in \overline{\mathbf{B}}$, **Assign** $\overline{B}_t^{\text{adv}} \leftarrow \emptyset$

9: **for** $l \in A$ **do**

10: **Take** the next batch $\overline{B}_a \subset \overline{B}_{i+1}$ with the size $m/|A|$

11: $\forall \overline{x}_j \in \overline{B}_a$: **Craft** $\overline{x}_j^{\text{adv}}$ by using attack algorithm $A[l]$ with $l_\infty(\mu_t)$, $\overline{B}_t^{\text{adv}} \leftarrow \overline{B}_t^{\text{adv}} \cup \overline{x}_j^{\text{adv}}$

12: **Descent:** $\theta_1 \leftarrow \theta_1 - \varrho_t \nabla_{\theta_1} \overline{\mathcal{R}}_{\overline{B}_t \cup \overline{B}_t^{\text{adv}}}(\theta_1)$; $\theta_2 \leftarrow \theta_2 - \varrho_t \nabla_{\theta_2} \overline{L}_{\overline{B}_t \cup \overline{B}_t^{\text{adv}}}(\theta_2)$ with the noise $\frac{\chi_3}{m}$

Output: $\epsilon = (\epsilon_1 + \epsilon_1/\gamma_\mathbf{x} + \epsilon_1/\gamma + \epsilon_2)$-DP parameters $\theta = \{\theta_1, \theta_2\}$, robust model with an ϵ_r budget

Algorithm

$$L_{\overline{B}_t \cup \overline{B}_t^{\text{adv}}}(\theta_2) = \frac{1}{m(1+\xi)}\Big(\sum_{\overline{x}_i \in \overline{B}_t} \mathcal{L}\big(f(\overline{x}_i, \theta_2), y_i\big) + \xi \sum_{\overline{x}_j^{\text{adv}} \in \overline{B}_t^{\text{adv}}} \Upsilon\big(f(\overline{x}_j^{\text{adv}}, \theta_2), y_j\big)\Big)$$

Theorem 4 *Algorithm 1 achieves* $(\epsilon_1 + \epsilon_1/\gamma_\mathbf{x} + \epsilon_1/\gamma + \epsilon_2)$*-DP parameters* $\overline{\theta} = \{\overline{\theta}_1, \overline{\theta}_2\}$ *on the private training data* D *across* T *gradient descent-based training steps.*

Outline

- Motivation and Background
- Differential Privacy in Adversarial Learning
- Composition of Certified Robustness
- Stochastic Batch Training (StoBatch)
- Experimental Results and Conclusion

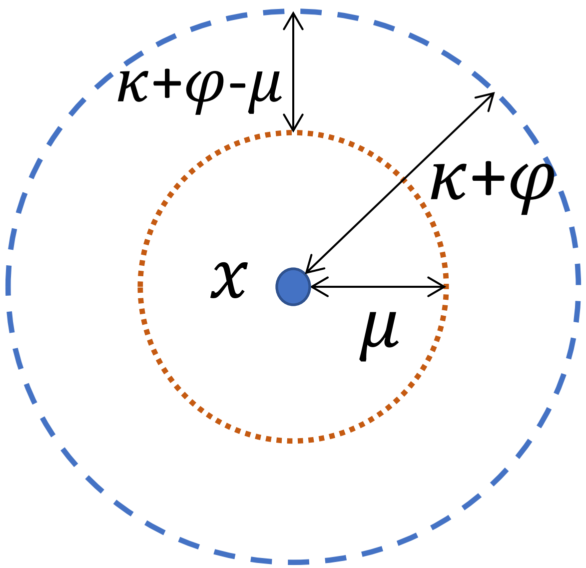
NJIT New Jersey Institute of Technology

Composition of Certified Robustness

- Project the privacy noise p on the scale of the robustness noise r.

$$\kappa = \frac{\Delta_{\mathcal{R}}}{m\varepsilon_1} / \frac{\Delta_r^x}{\varepsilon_r}, \qquad \bar{x}_i = x_i + Lap\left(\frac{\kappa \Delta_r^x}{\varepsilon_r}\right)$$

$$\varphi = \frac{\Delta_{\mathcal{R}}}{m\varepsilon_1} / \frac{\Delta_r^h}{\varepsilon_r}, \qquad \bar{h}_i = h_i + Lap\left(\frac{\varphi \Delta_r^h}{\varepsilon_r}\right)$$



- What is the general robustness bound, given κ and φ?

$$f(\mathcal{M}_1, \ldots, \mathcal{M}_S | x) : \mathbb{R}^d \rightarrow \prod_{s \in [1,S]} f^s(x) \in \mathbb{R}^K$$

Sequential Composition of Certified Robustness: Lemma 5, Theorem 5

$$\forall \alpha \in l_p(\kappa + \varphi) : \hat{\mathbb{E}} f_k(x + \alpha) > \max_{i: i \neq k} \hat{\mathbb{E}} f_i(x + \alpha)$$

Verified Inference

- StoBatch Robustness

$$(\kappa + \varphi)_{max} = \max_{\epsilon_r} \frac{\Delta_{\mathcal{R}}\epsilon_r}{m\epsilon_1}\left(\frac{1}{\Delta_r^x} + \frac{2}{\Delta_r^h}\right) \text{ s.t.}$$

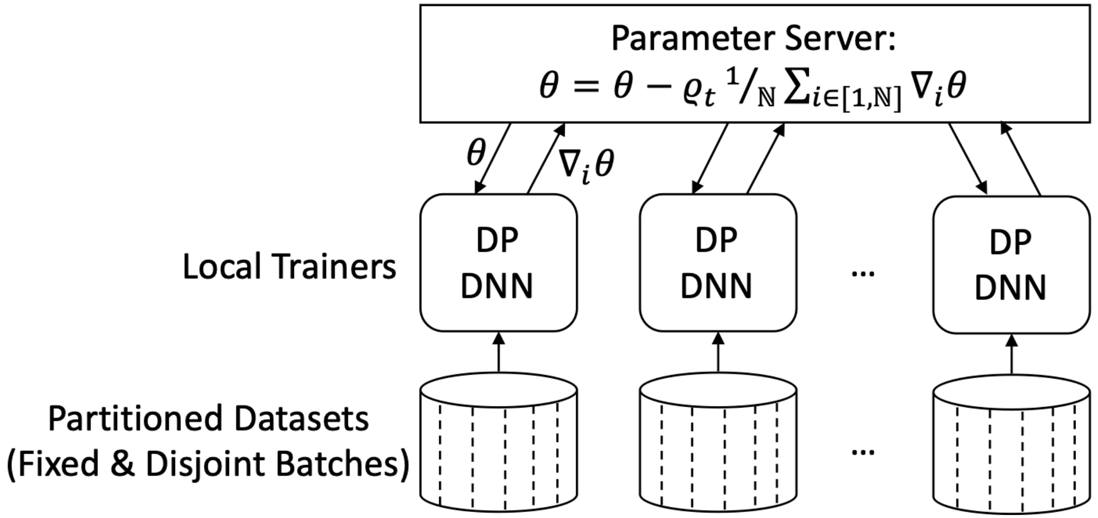
$$\hat{\mathbb{E}}_{lb} f_k(x) > e^{2\epsilon_r} \max_{i:i\neq k} \hat{\mathbb{E}}_{ub} f_i(x) \text{ and}$$

$$\overline{x} = x + Lap\left(\frac{\kappa\Delta_r^x}{\epsilon_r}\right), \quad \overline{h} = h + Lap\left(\frac{\varphi\Delta_r^h}{\epsilon_r}\right)$$

$$\forall \alpha \in l_p(\kappa + \varphi)_{max}: f_k(x + \alpha) > \max_{i:i\neq k} f_i(x + \alpha)$$

where $k = y(x)$, indicating that a small perturbation in the input does not change the predicted label $y(x)$.

Stochastic Batch Mechanism



- Under the same DP protection.
- Training from multiple batches with more adversarial examples, without affecting the DP bound.
- The optimization of one batch does not affect the DP protection at any other batch and at the dataset level D, across T training steps.

Outline

- Motivation and Background
- Differential Privacy in Adversarial Learning
- Composition of Certified Robustness
- Stochastic Batch Training (StoBatch)
- Experimental Results and Conclusion

Experimental Results

- Interplay among model utility, privacy loss, and robustness bounds
  - privacy budget
  - attack sizes
  - scalability

- CNNs on MNIST, CIFAR-10
- ResNet-18 on Tiny ImageNet

- Baseline approaches
  - PixelDP [Lecuyer et al., S&P'19]
  - DPSGD [Abadi et al., CCS'16]
  - AdLM [Phan et al., ICDM'17]
  - Secure-SGD [Phan et al., IJCAI'19] with AGM [Balle et al., ICML'18]

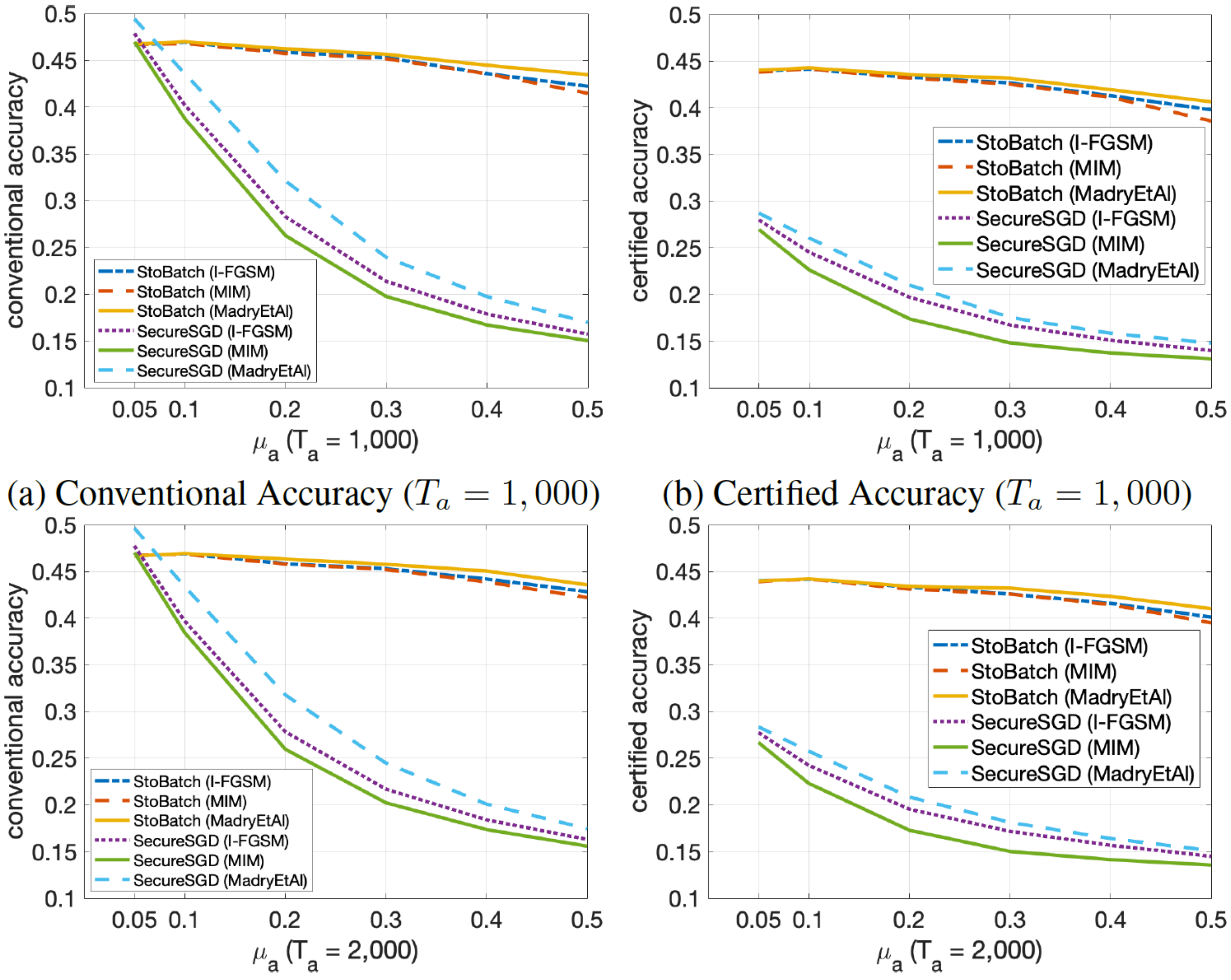
$$conventional\ acc = \sum_{i=1}^{|test|} \frac{isCorrect(x_i)}{|test|}$$

$$certified\ acc = \sum_{i=1}^{|test|} \frac{isCorrect(x_i)\ \&\ isRobust(x_i)}{|test|}$$

[Lécuyer et al., 2019]

CIFAR-10

- StoBatch
  - 45.25 ± 1.6% (conventional)
  - 42.59 ± 1.58% (certified)
- SecureSGD
  - 29.08 ± 11.95% (conventional)
  - 19.58 ± 5.0% (certified)
- p < 2.75e-20
  - 2-tail t-test



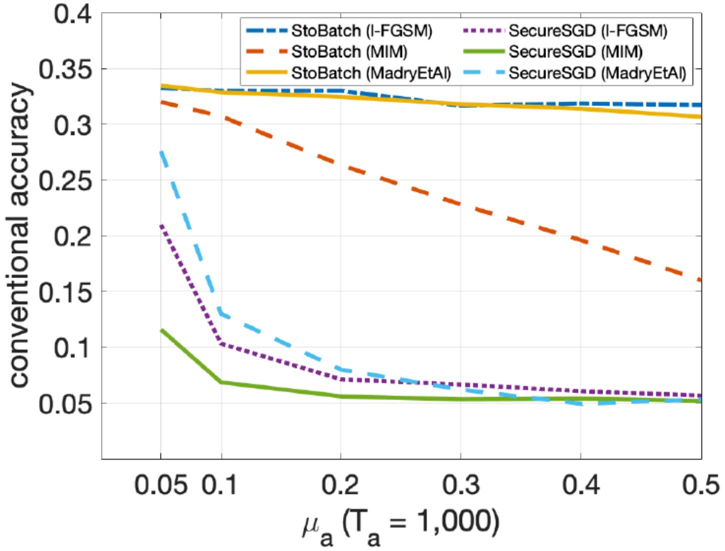
(a) Conventional Accuracy ($T_a = 1,000$) (b) Certified Accuracy ($T_a = 1,000$)

(c) Conventional Accuracy ($T_a = 2,000$) (d) Certified Accuracy ($T_a = 2,000$)

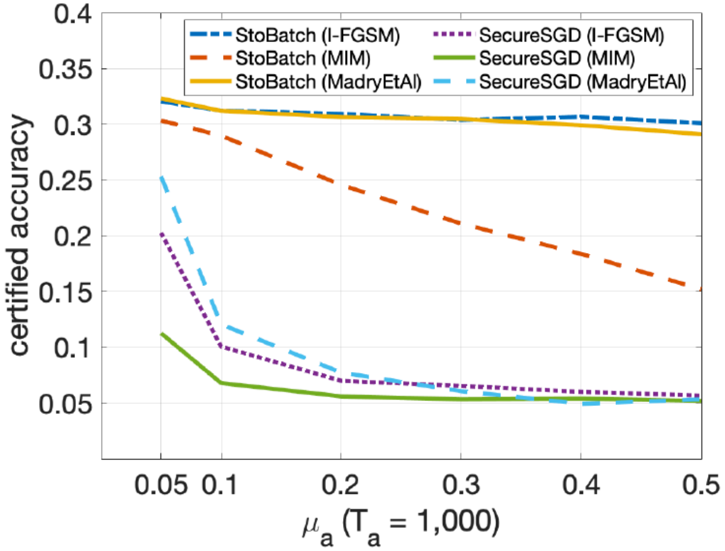
Accuracy on the CIFAR-10 dataset, under Strong Iterative Attacks ($T_a = 1,000; 2,000$). ϵ is set to 2 (tight DP protection).

Tiny ImageNet

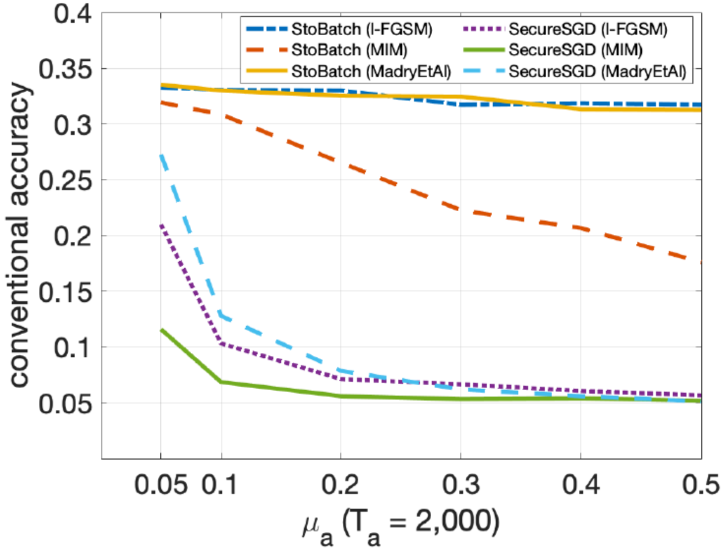
- StoBatch
  - 29.78 ± 4.8% (conventional)
  - 28.31 ± 1.58% (certified)
- SecureSGD
  - 8.99 ± 5.95% (conventional)
  - 8.72 ± 5.5% (certified)
- p < 1.55e-42
  - 2-tail t-test



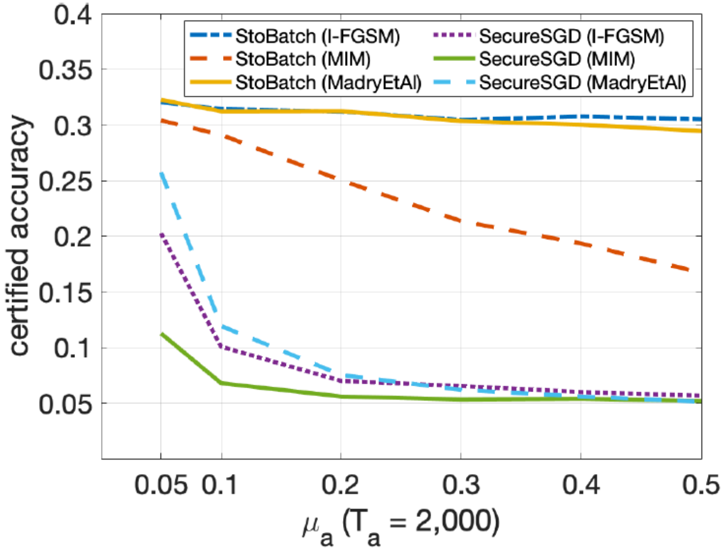
(a) Conventional Accuracy ($T_a = 1,000$)



(b) Certified Accuracy ($T_a = 1,000$)



(c) Conventional Accuracy ($T_a = 2,000$)



(d) Certified Accuracy ($T_a = 2,000$)

Accuracy on the Tiny ImageNet dataset, under Strong Iterative Attacks ($T_a = 1,000; 2,000$). ϵ is set to 5.

Conclusion

- Established a connection among DP preservation to protect the training data, adversarial learning, and certified robustness.
- Derived a sequential composition robustness in both input and latent spaces.
- Addressed the trade-off among model utility, privacy loss, and robustness.
- Rigorous experiments shown that our mechanism significantly enhances the robustness and scalability of DP DNNs.

Thank you!

phan@njit.edu, we are hiring!