Local Differential Privacy for Deep Learning

SAMARTH JAIN

22125033

Agenda

Introduction

Working of Differential Privacy Model

Algorithms

Results

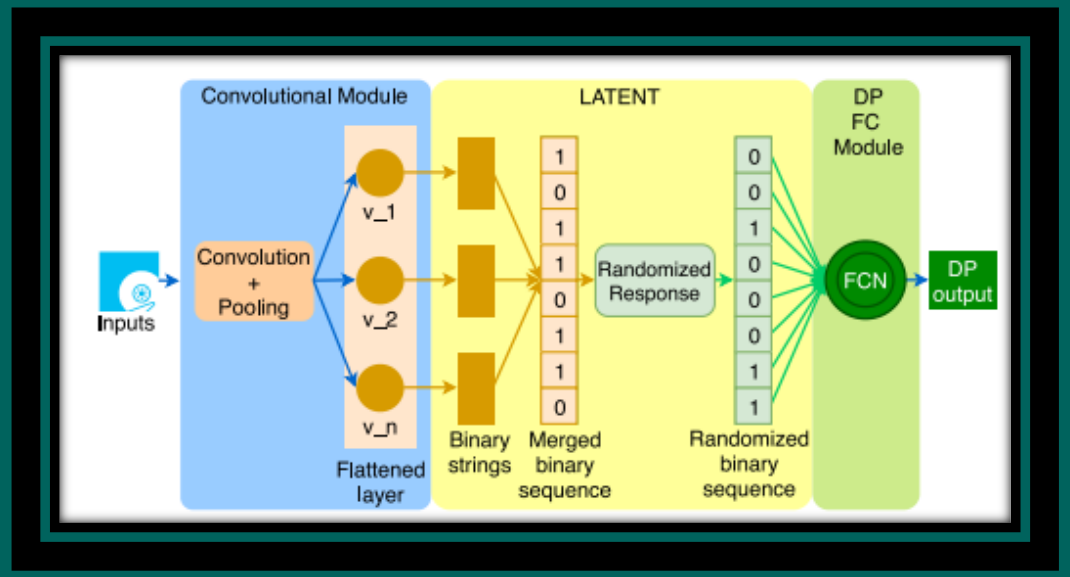
Conclusion

Introduction

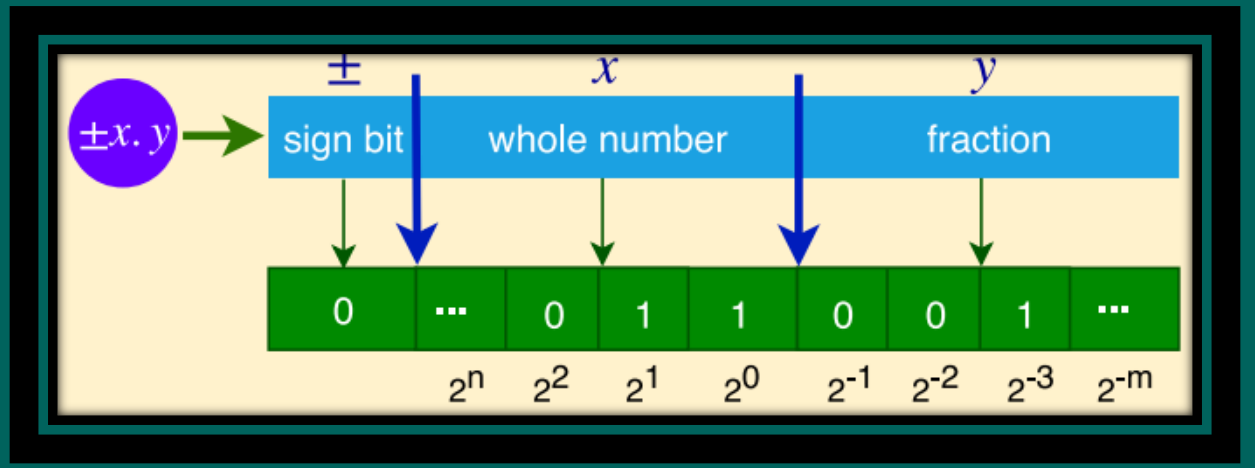
- IoT generates vast amounts of data, driving advancements in technologies like big data analytics and machine learning, unlocking new opportunities across various industries.
- Integrating machine learning algorithms into distributed IoT environments, particularly those controlled by SDNs and NFVs, faces challenges due to server-centric architectures and privacy concerns.
- Deep learning models, often trained on sensitive data, can pose privacy risks, especially in distributed, cloud-based environments where adversaries may exploit vulnerabilities to extract sensitive information.
- Differential privacy (DP) emerges as a robust framework for mitigating privacy risks in deep learning. Local DP (LDP) is favored over global DP (GDP) due to its ability to preserve privacy without relying on a trusted curator.

Introduction

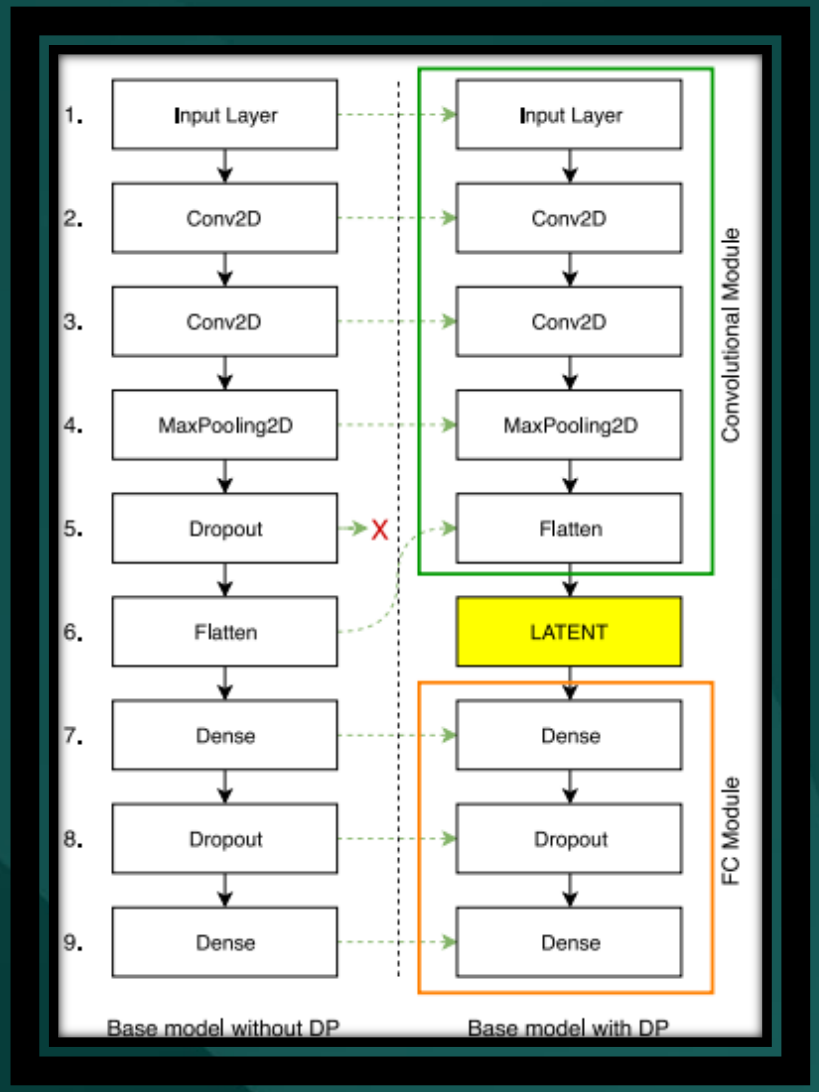
- LATENT, a distributed LDP mechanism, employs a novel LDP protocol utilizing properties of randomized response. This protocol enhances privacy preservation in convolutional neural network (CNN) models released as black-box models.
- UER protocol, an improvement over existing LDP protocols, enhances utility during data randomization. It introduces flexibility in choosing randomization probabilities, leading to improved accuracy and privacy preservation.
- LATENT's integration into modern environments, such as SDN-controlled NFV, is facilitated by moving the randomization layer to run as an NFV service. Its effectiveness is demonstrated through experiments, showing high accuracy and efficiency even on general-purpose computers.



CNN architecture with the LATENT randomization layer



Direct mapping of a float/integer to binary



Architectural differences between the nonprivate (NPCNN) and differentially private (DPCNN) baseline models for the MNIST data set

Working of Differential Privacy Model Generation

- Define the CNM as explained
- Declare, **l = (m + n + 1)**
- Feed {x1,…, xj} to the CNM and generate the sequence of 1-D feature arrays $\{d_1,\ldots, d_j\}$
- Convert each field (x) of d_q (where, q = 1,…, j) to binary using, $\mathbf{g(i) = (2^{-k}\ |x|\ \bmod 2)^{\,n}_{\,k=-m}}$; where, i = k + m
- Generate array $\{b_1,\ldots, b_j\}$ of the merged binary arrays for the elements in $\{d_1,\ldots, d_j\}$

- Determine the length (r) of a single element of {d1,…, dj}
- Calculate randomization probability according to Equation
- Randomize each element of {b1,…, bj} using UER with probability p to generate {pb1,…, pbj}
- Train the FC module of the CNN using {pb1,…, pbj}
- Optimize the FC module using regularization, image augmentation and/or hyperparameter tuning
- Return the DPFC module;

Algorithms

Probability of randomizing the ith bit of the binary encoded string of v for any inputs v1, v2 with a sensitivity = r x l ;

$$p(B[i]v) = \begin{cases} \Pr[\boldsymbol{B}[v_1] = 1|v_1] = \frac{\alpha}{1+\alpha}, & \text{if } i \in 2n;\ n \in \mathbb{N} \\ \Pr[\boldsymbol{B}[v_2] = 0|v_1] = \frac{\alpha e^{\frac{\varepsilon}{rl}}}{1+\alpha e^{\frac{\varepsilon}{rl}}} & \text{,,} \\ \Pr[\boldsymbol{B}[v_1] = 1|v_1] = \frac{1}{1+\alpha^3}, & \text{if } i \in 2n+1. \\ \Pr[\boldsymbol{B}[v_2] = 0|v_1] = \frac{\alpha e^{\frac{\varepsilon}{rl}}}{1+\alpha e^{\frac{\varepsilon}{rl}}} & \text{,,} \end{cases}$$

Algorithms

- ε ← privacy budget
- n ← number of bits for the whole number of the binary representation
- m ← number of bits for the fraction of the binary representation
- a ← privacy budget coefficient
- r ← no of outputs of flattening layer of CNM

$$\varepsilon = \ln\left(\frac{p(1-q)}{(1-p)q}\right)$$

$$\Pr[B'[i] = 1] = \begin{cases} p, & \text{if } B[i] = 1 \\ q, & \text{if } B[i] = 0. \end{cases}$$

Results

Dataset		NPCNN	LATENT	
			ε=2	ε=0.5
MNIST	Training	93.69%	11.33%	11.52%
	Testing	93.70%	11.35%	11.56%
CIFAR-10	Training	79.88%	10.00%	10.14%
	Testing	79.87%	10.01%	10.17%

Conclusion

- The proposed LDP mechanism achieves exceptional accuracy even with stringent privacy constraints (e.g., $\varepsilon = 0.5$), outperforming existing differentially private approaches for MNIST and CIFAR-10 datasets.
- By enabling a distribution of the CNN structure between data owners and servers, the method alleviates the need for a trusted curator, enhancing privacy while reducing computational burden on data owners.
- The distributed architecture enhances flexibility in data processing, particularly beneficial in big data contexts, and facilitates adaptation to emerging technologies like SDN and NFV in edge-cloud interactions.
- The method allows private sharing of sensitive data and mitigates privacy leaks in distributed ML scenarios, offering a robust solution for preserving privacy in collaborative learning environments.
- Architectural independence between the LDP component (LATENT) and the fully connected ANN component (FC module) streamlines parameter selection, enabling easy training and tuning of the FC module for higher accuracy and extreme privacy.
- The approach suggests avenues for future research, including exploring methods to reduce data sensitivity, testing the method on diverse DL architectures like LSTM, and evaluating its performance on various large datasets to assess its generalizability and scalability.

Thank You

SAMARTH JAIN
22125033
DATA SCIENCE AND ARTIFICIAL INTELLIGENCE