Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

Privacy-Preserving Federated Learning Platform using Differential Privacy And Secure Aggregation

Bhuvaneshwari. S
M.E Cse
Kings College Of Engineering,Punalkulam
Pudukkottai,India

Dr. K. Abhirami
Prof. Dept of CSE
Kings College of Engineering,Punalkulam
Pudukkottai,India

Abstract **– Federated learning enables decentralized machine learning model training on distributed client devices whilepreserving raw data privacy. However, vulnerability to model inversion and gradient leakage attacks remains acritical security threat during parameter communication. This paper proposes a comprehensive Privacy-Preserving Federated Learning (PPFL) framework integrating Differential Privacy (DP) via per-sample gradient clipping and Gaussian noise addition with cryptographic Secure Aggregation (SecAgg). In the proposed platform, participating edge clients retain their datasets locally and execute private model training. Local updates are privatized using Rényi Differential Privacy (RDP) accounting before undergoing double-masked secret sharing for secure server-side aggregation. Experimental validation using Python PyTorch,Flower FLwr, and Opacus frameworks demonstrates robust privacy protection with minimal utility degradation across multi-round distributed training.**

Keywords - Federated Learning, Differential Privacy, Secure Aggregation, Privacy-Preserving MachineLearning, DP-SGD, Flower Framework, Opacus, RDP Accounting.

I. INTRODUCTION

In recent years, artificial intelligence and deep learning have become fundamental technologies acrosssensitive domains including healthcare, banking, education, and cybersecurity. These models rely heavily on large, diverse datasets to achieve generalized accuracy. Traditionally, data collection requires centralizing private datasets onto a single server. However, centralized storage poses severe data privacy, compliance, and cyber-attack vulnerabilities. Organizations are increasingly restricted by regulatory frameworks such as GDPR and HIPAA from sharing raw data.Federated Learning (FL) addresses this challenge by allowing multiple decentralized clients to collaboratively train a shared global model without exchanging raw datasets. Each client executes local model training using its local private dataset and transmits only parameter updates or gradients to a central server. The central aggregator combines client updates using algorithms like Federated Averaging (FedAvg) to form an improved global model.Despite keeping raw data on local devices, basic Federated Learning does not guarantee absolute privacy.Adversaries can inspect exposed parameter updates to perform gradient inversion, membership inference, and attribute reconstruction attacks. To mitigate these vulnerabilities, this project develops an integrated Privacy Preserving Federated Learning Platform incorporating Differential Privacy (DP) and Secure Aggregation(SecAgg). Differential Privacy adds calibrated statistical noise to limit individual record leakage, while Secure Aggregation cryptographically masks client updates so that the central server learns only the aggregated sum without accessing individual client updates.

A. Federated Learning and Its Working Process

Federated Learning is a distributed machine learning technique in which multiple clients collaboratively train a shared global model while keeping their original training data on their respective devices or local systems. Unlike 2 traditional centralized machine learning, Federated Learning does not require every participating client to transfer its complete dataset to a central storage system. Instead, each client independently trains a local model and shares the necessary model updates with a central server. The working process generally begins when the central server creates an initial global machine learning model and distributes it to the participating clients

B. Differential Privacy and Secure Aggregation

Differential Privacy is a mathematical privacy protection technique designed to reduce the possibility of identifying an individual's contribution to a dataset or computation. It achieves this by introducing carefully controlled random noise into the output of a computation. The amount of privacy protection is commonly described using privacy parameters such as epsilon (ε) and delta (δ). Generally, a smaller epsilon value indicates stronger privacy protection, although it may also reduce model accuracy. Differential Privacy can be applied to Federated Learning by clipping individual client

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

contributions and adding calibrated noise to model updates or aggregated results.

C. Privacy-Preserving Federated Learning Platform

The proposed project develops a Privacy-Preserving Federated Learning Platform using Differential Privacy and Secure Aggregation. The main objective is to enable multiple clients to collaboratively train a machine learning model without sharing their original private datasets with a central server. The platform introduces privacy protection mechanisms into the model training and aggregation process to reduce the possibility of sensitive information exposure

II. LITERATURE SURVEY

A. Communication-Efficient Learning of Deep Networks

Introduced Federated Learning and the FedAvg algorithm, proving that local iterative SGD drastically reduces communication overhead compared to synchronized training. (Authors: McMahan et al. (2017))

Advantages: Protects raw data locality and reduces network bandwidth.

Disadvantages: Susceptible to gradient inference attacks and non-IID data degradation.

B. Deep Learning with Differential Privacy

Formulated DP-SGD featuring per-sample gradient clipping and moments accountant tracking for privacy loss bounds in neural networks. (Authors: Abadi et al. (2016))

Advantages: Provides mathematical privacy guarantees with quantifiable epsilon bounds.

Disadvantages: Calibrated noise insertion degrades model utility and convergence speed.

C. Practical Secure Aggregation for Privacy-Preserving Machine Learning

Designed a lightweight secret-sharing cryptographic protocol enabling high-dimensional vector aggregation resilient to client dropouts. (Authors: Bonawitz et al. (2017))

Advantages: Ensures individual client updates remain zero-knowledge to the central server.

Disadvantages: High computational complexity during key agreement and secret reconstruction.

D. Survey of Privacy-Preserving Federated Learning

Provided a taxonomy of threat models, attack vectors, and defense mechanisms in distributed learning frameworks. (Authors: Yin et al. (2021))

Advantages: Establishes structured privacy categorization across architecture layers.

Disadvantages: Qualitative review without empirical bench-marking.

E. Decentralized Privacy-Preserving Aggregation

Explored ADMM-based consensus optimization for decentralized privacy without relying on central aggregators. (Authors: Jeon et al. (2021))

Advantages: Eliminates single-point server trust requirements.

Disadvantages: Increases peer-to-peer communication latency significantly.

III. SYSTEM ANALYSIS

Traditional machine learning relies on centralized data aggregation, exposing sensitive records to single-point breaches. Standard Federated Learning improves data locality but remains vulnerable to gradient leakage. The proposed framework solves these limitations by combining two complementary privacy layers: Differential Privacy (preventing output inference) and Secure Aggregation (preventing server-side update inspection).

A. System Requirements

Hardware: Intel Core i5/i7 Processor, 8GB/16GB RAM, 256GB SSD storage.
Software: Python 3.11, PyTorch (Deep Learning Core), Flower FLwr (FL Protocol Engine), Opacus (Differential Privacy), Flask (Monitoring Interface), NumPy & Pandas.

B. Architecture & Dataflow Workflow

The platform execution sequence follows five structuralstages:
1. Client Registration & Dataset Preprocessing: Clients initialize local models with global weights.
2. Differentially Private Local Training: Clients run local epoch SGD, apply per-sample gradient clipping (norm C), and add Gaussian noise scaled by multiplier sigma.
3. Cryptographic Masking: Clients generate pairwise secrets and apply additive blinding masks to update vectors.
4. Secure Server Aggregation: The server sums blinded updates, canceling pairwise masks automatically to compute true FedAvg weights.
5. Global Model Update & Evaluation: Updated parameters are evaluated and broadcasted for subsequent rounds.

IV. IMPLEMENTATION

- Client Registration & Local Training Module: Manages client authentication and PyTorch modeliteration.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

- Differential Privacy Engine: Employs Opacus DP-SGD wrappers for gradient clipping and Gaussian noise injection.
- Secure Aggregation Protocol Module: Manages cryptographic pairwise secret exchange and mask generation.
- Global Model Coordination Engine: Coordinates Flower FLwr multi-round training loops.
- Performance & RDP Accounting Dashboard: Monitors accuracy, loss, and privacy budget epsilon via Flask UI.

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

The platform was empirically validated across multiple simulated clients running local differential privacy. The evaluation metrics track model global accuracy, loss reduction, and cumulative privacy consumption (epsilon) over 10 training rounds.

VI.CONCLUSION AND FUTURE ENHANCEMENTS

The developed Privacy-Preserving Federated Learning Platform successfully proves that robust data privacy and high machine learning accuracy can co-exist. By integrating Differential Privacy (Opacus) with Secure Aggregation (Flower FLwr), the framework eliminates both single-point raw data exposure and server-side update interception. Future enhancements will focus on deploying hardware-accelerated TEE (Trusted Execution Environments), expanding to non-IID datasets, and optimizing network overhead for mobile IoT edge environments.

REFERENCES

[1] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, 'Communication-Efficient Learning of Deep Networks from Decentralized Data,' AISTATS, pp. 1273-1282, 2017.
[2] M. Abadi et al., 'Deep Learning with Differential Privacy,' ACM CCS, pp. 308-318, 2016.
[3] K. Bonawitz et al., 'Practical Secure Aggregation for Privacy-Preserving Machine Learning,' ACM CCS, pp. 1175-1191, 2017.
[4] P. Kairouz et al., 'Advances and Open Problems in Federated Learning,' Foundations and Trends in Machine Learning, vol. 14, no. 1-2, pp. 1-210, 2021.
[5] X. Yin, Y. Zhu, and J. Hu, 'A Comprehensive Survey of Privacy-Preserving Federated Learning,' ACM Computing Surveys, vol. 54, no. 6, 2021.
[6] I. Mironov, 'Rényi Differential Privacy,' IEEE CSF, pp. 263-275, 2017.
[7] C. Dwork and A. Roth, 'The Algorithmic Foundations of Differential Privacy,' Foundations and Trends in TCS, vol. 9, no. 3-4, pp. 211-407, 2014.
[8] Q. Yang et al., 'Federated Machine Learning: Concept and Applications,' ACM TIST, vol. 10, no. 2, 2019.
[9] T. Li et al., 'Federated Learning: Challenges, Methods, and Future Directions,' IEEE Signal Processing Magazine, vol. 37, no. 3, pp. 50-60, 2020.
[10] W. N. Chen et al., 'The Fundamental Price of Secure Aggregation in Differentially Private Federated Learning,' ICML, vol. 162, 2022

(This work is licensed under a Creative Commons Attribution 4.0 International License.)