2024 IEEE 4th International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA) | 979-8-3503-6360-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICIBA62489.2024.10868170

AI Model Training Data Privacy Protection Scheme Based on Local Differential Privacy

Yue Zhang[1] ,Lin Li[1], Cong Hou[1], Min Li[1], Xiaotian Xu[1]
State Grid Jibei Electric Power Co. Ltd. Research Institute, Beijing, China
Corresponding Author: Yue Zhang Email: 741473522@qq.com

***Abstract*—In this paper, we present a novel AI model training data privacy protection scheme based on local differential privacy (LDP), aimed at safeguarding sensitive data in distributed environments. The method introduces dynamic noise adjustment to balance privacy and accuracy, optimizing model performance without compromising privacy. During the data preprocessing phase, the scheme applies standardization, handles missing values, and ensures format consistency. Noise is added dynamically to both the data and model gradients during training to further enhance privacy. The proposed method's scalability and computational efficiency were validated in large-scale, real-time AI applications, demonstrating significant reductions in overhead compared to centralized differential privacy techniques. Our results show that the dynamic adjustment of noise helps maintain high model accuracy while offering robust privacy guarantees, making the scheme ideal for use in distributed AI systems.**

Keywords—local differential privacy (LDP); AI model training, privacy-accuracy trade-off

I. INTRODUCTION

With the widespread adoption of artificial intelligence (AI) in various industries, from healthcare to finance and smart grids, the importance of data privacy has become more pressing than ever. The ability of AI systems to analyze vast amounts of data allows for enhanced decision-making, optimized operations, and increased efficiency. However, this benefit comes with a significant cost—privacy risks. AI models often rely on large datasets that may contain sensitive personal information such as medical records, financial transactions, or usage patterns in critical infrastructure like electricity grids. Protecting the privacy of this data is crucial, not only for legal compliance but also to maintain trust between users and AI systems.

Traditional approaches to privacy protection, such as encryption and anonymization, have been widely used in various applications. Nevertheless, when applied to AI model training, these methods present significant limitations. Data anonymization, while useful in some contexts, often results in degraded model performance due to loss of valuable information. Additionally, anonymized datasets are susceptible to re-identification attacks, especially when combined with auxiliary data sources. On the other hand, encryption techniques like homomorphic encryption [1] enable computations on encrypted data but are computationally expensive, making them impractical for large-scale AI systems where real-time data processing is critical. In distributed AI settings, such as federated learning [2], even though raw data is not shared, model updates (gradients) exchanged between devices can still leak sensitive information, exposing vulnerabilities in the system.

As AI systems become more integrated with real-time applications—such as smart grids, autonomous systems, and medical diagnostics—the need for privacy-preserving methods that can safeguard data without compromising model performance has grown. Existing solutions either place a heavy computational burden on the system or fall short in preventing privacy leakage, especially in decentralized or distributed environments. This creates a significant challenge for AI systems that require both high efficiency and robust privacy protections.

This paper addresses this challenge by proposing a solution based on local differential privacy (LDP) [3] for AI model training in distributed environments. Local differential privacy allows noise to be added directly to the data at the point of collection, ensuring that privacy is protected before the data is even transmitted. This method not only eliminates the need for centralized data aggregation but also reduces the risk of privacy breaches during data sharing. Additionally, this approach can be extended to protect model training by applying noise to the gradients during the learning process, thereby preventing adversaries from inferring sensitive information through model updates [4][5].

By leveraging LDP in both the data collection and model training phases, this method ensures a strong privacy guarantee while maintaining the utility of the AI models. This paper explores the practical application of local differential privacy in real-time AI systems, such as those deployed in smart grids, and evaluates its effectiveness in balancing privacy protection with model performance.

II. RELATED WORK

The problem of privacy protection in data fusion and AI model training has been widely studied, with a range of approaches that vary in their effectiveness, efficiency, and applicability. In this section, we review the most relevant literature in two key areas: privacy-preserving AI training and data fusion in distributed environments.

A. *Privacy-Preserving AI Model Training*

Differential privacy has been a popular tool for protecting individual data in machine learning models. One of the most prominent works in this field is the differentially private stochastic gradient descent (DP-SGD), introduced by Abadi et al. (2016) [6], which applies noise to gradients during the training process to guarantee privacy. However, DP-SGD operates in a centralized setting where all the data is collected at a central server, making it unsuitable for scenarios involving distributed data fusion.

Homomorphic encryption is another method used for privacy protection in AI, allowing computations on encrypted data. However, this approach is computationally expensive and can be impractical for real-time or large-scale applications due to its high computational overhead.

In contrast, local differential privacy (LDP) offers a more practical approach for distributed data environments, as it allows users to add noise to their data locally before sharing it with a central server or participating in a distributed model training process. Geyer et al. (2017) explored LDP in the context of federated learning, providing privacy guarantees at the client level. Our work builds on these ideas, applying LDP to protect data in distributed data fusion environments [7][8].

B. *Differential privacy data protection scheme*

Existing work on privacy in data fusion has mainly focused on secure multi-party computation (SMC) and homomorphic encryption, which enable data processing without exposing raw data. However, these techniques are computationally expensive and do not scale well to large, distributed datasets. To overcome these challenges, we introduce LDP-based data fusion for AI training, which can provide robust privacy protection while maintaining computational efficiency.

Differential privacy [3] protection can overcome the limitations of traditional privacy protection techniques, such as reliance on attackers' background knowledge, the difficulty in quantitatively describing protection effectiveness with effective mathematical methods, etc. Thus, it can significantly reduce the risk of privacy leakage in protected datasets while maintaining the availability of dataset data as much as possible. The process involves adding random perturbations to real data and ensuring that the data remains usable after interference, aiming to distort protected data while simultaneously preserving specific data or data attributes in the dataset.

The implementation of differential privacy [9] typically involves randomization or perturbation operations during data processing. The most common method is to add random noise during data queries or publication processes. By introducing an appropriate amount of randomness into the data, it becomes difficult for attackers to accurately infer the data contributions of specific individuals, thereby protecting individual privacy. Additionally, differential privacy involves the concept of privacy budget, which limits the extent of privacy information leakage, ensuring the protection of individual privacy while maintaining the validity and availability of data as much as possible during data analysis [10].

III. AI Model Training Data Privacy Protection Scheme Based on Local Differential Privacy

In this section, we present the AI model training data privacy protection scheme based on Local Differential Privacy (LDP). The scheme is designed to protect the privacy of data in both data collection and training phases of AI models, especially in distributed environments where data is generated and processed across multiple devices. Our approach involves adding noise at two critical stages: directly at the data collection point to ensure privacy protection before transmission, and during the model training process to further safeguard privacy against adversarial attacks aimed at the learning process. The overall architecture of the algorithm is depicted in Figure 1. Below, we outline the key steps of the scheme.

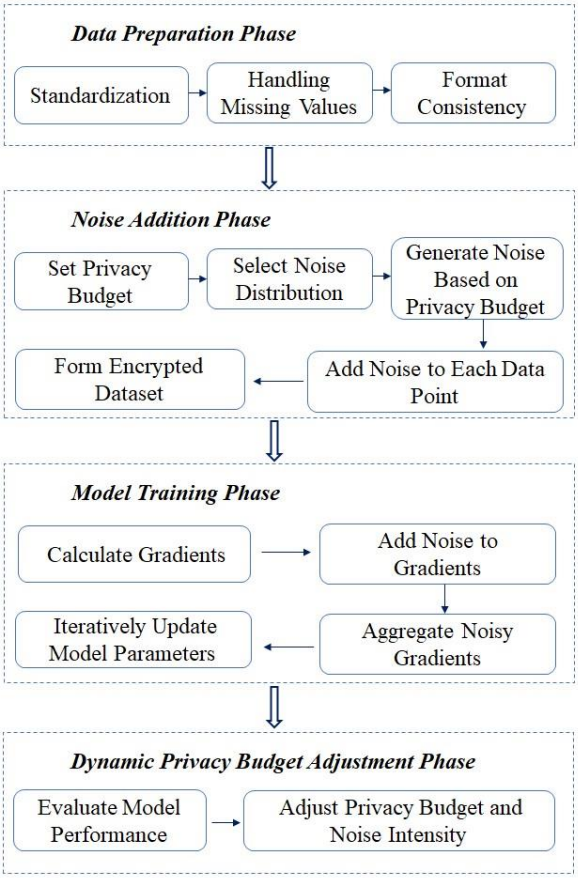

Figure 1 AI Model Training Data Privacy Protection Based on Local Differential Privacy

The primary objective of the scheme is to secure data privacy by integrating noise at various stages of the AI model training pipeline. The key components of the system include:

(1) Noise generation and application at the data collection stage: before the data is transmitted or shared, local differential privacy is applied to each data point, ensuring that sensitive information is protected at the source.

(2) Noise-enhanced gradient computation: during the model training process, noise is added to the gradients computed for model updates, which helps obscure the underlying data characteristics that could otherwise be inferred by analyzing the model's learning patterns.

(3) Iterative model updates: the noisy gradients are used iteratively to update the model parameters, ensuring that the training process remains efficient while maintaining strong privacy guarantees.

Each of these steps contributes to a comprehensive privacy protection mechanism that balances data security with the need for efficient and accurate model training. The following subsections would then elaborate on each of these steps.

A. Data Preparation Phase

In the data preparation phase, the privacy budget and noise distribution parameters are chosen. Additionally, data undergoes preprocessing to ensure it aligns with the algorithm's requirements. The steps are as follows:

Step 1 the privacy budget ε controls the strength of privacy protection. A smaller ε means stronger protection but can reduce the utility of the data. In this scheme, we introduce dynamic privacy budget adjustment during model training. Initially, the privacy budget is set to a conservative value ε_0. As model training progresses and the model's accuracy is evaluated, ε may be adjusted to balance privacy and utility. Dynamic adjustment of ε can be defined as:

$$\varepsilon_t = \varepsilon_0 \cdot (1 + \alpha \cdot \frac{current\ model\ accuracy}{target\ accuragy})$$

Where α is a tuning parameter and t denotes the iteration step.

Step 2 choosing the noise distribution: based on the principles of local differential privacy (LDP), we use the Laplace noise mechanism for perturbing data. The scale of the Laplace distribution b is determined by the privacy budget ε, calculated as: $b = \frac{\Delta f}{\varepsilon}$, where Δf is the sensitivity of the function to be privatized((According to the LDP algorithm definition, , Δf refers to the maximum change value $\Delta f = \max_{D,D'} \left|\left| f(D) - f(D') \right|\right|$, where D, D' are two adjacent datasets, when only one data point in the input dataset is different.)).

Step 3 data preprocessing: before adding noise, the raw data must undergo preprocessing to ensure consistency and compatibility with the differential privacy algorithm. This involves standardization, where features are scaled to have a mean of zero and a standard deviation of one. Additionally, missing values must be handled, either by imputing them or removing incomplete records. Finally, data format consistency must be ensured so that it aligns with the requirements of the differential privacy mechanism.

B. Data Noise Addition Phase

In this phase, appropriate noise is generated based on the preset privacy budget and noise distribution parameters. The noise generation follows the requirements of the local differential privacy mechanism to obscure sensitive information in user data. The generated noise is added to each individual user data point, resulting in an encrypted dataset. This ensures that data remains protected during transmission and storage, effectively preventing privacy breaches.

Step 1: generate Laplace noise. based on the noise scale b obtained in the previous steps, Laplace noise z is generated. According to the probability density function of the noise, the noise value z is added, where $Lap(z|0,\ b) = \frac{1}{2b}\exp\left(-\frac{|z|}{b}\right)$.

Step 2: for each preprocessed data point x_i, Laplace noise z_i is added, with the formula $x_i' = x_i + z_i$. This ensures that individual data points are masked before being uploaded to the server for training.

C. Model Training Phase

During model training, noise is added to gradients to protect privacy during the training process, preventing attackers from inferring the original data from model updates.

Step 1 gradient calculation: for each data point (x_i, y_i), calculate the gradient g_i of the loss function $L(\theta, x_i, y_i)$ with respect to the model parameters θ:

$$g_i = \nabla_\theta L(\theta, x_i, y_i)$$

Step 2 adding noise to gradients: to protect privacy, Laplace noise z_i is added to the gradient g_i, resulting in a perturbed gradient g_i' : $g_i' = g_i + z_i$. The noise is dynamically adjusted using the same privacy budget ε_t, which is updated based on the current model accuracy.

Step 3 aggregating gradients: for a mini-batch of data points (size m), aggregate the perturbed gradients to compute the batch gradient: $\overline{g'} = \frac{1}{m}\sum_{i=1}^{m} g_i'$.

Step 4 model parameter update: use the aggregated noisy gradient to update the model parameters with a learning rate η: $\theta_{t+1} = \theta_t - \eta . \overline{g'}$. The model is iteratively updated using this noisy gradient until convergence.

D. Dynamic Adjustment Phase

During the model training process, after multiple iterations of updating the model parameters, the performance evaluation stage is initiated. The primary goal of this stage is to assess the performance of the model,

which has been processed with differential privacy, on the test or validation set. This ensures that the model can maintain a high level of accuracy and generalization capability while preserving data privacy. In this phase, the model's performance is evaluated, and the noise and privacy budget are adjusted accordingly.

Step 1: evaluating model performance: after a set number of iterations, the model is evaluated on a separate validation set to measure accuracy and loss. This helps determine whether the current level of privacy and noise is appropriate.

Step 2: adjusting privacy budget and noise: if the model's performance (e.g., accuracy, loss) is significantly lower than expected, the privacy budget ε can be increased to reduce noise and improve model utility. The adjustment follows:

$$\varepsilon_{t+1} = \varepsilon_t + \beta \cdot \frac{target\ accuracy - current\ accuracy}{target\ accuragy}$$

Where β is a hyperparameter that controls the rate of adjustment. Noise z_i is recalculated based on the updated ε_{t+1}, ensuring that privacy is maintained while optimizing model performance.

IV. Performance Evaluation

To evaluate the performance of the proposed scheme, we conducted experiments on multiple datasets to assess both privacy guarantees and model accuracy. Our performance evaluation focuses on two key metrics [11]:

Privacy guarantee: we measure the privacy loss across different privacy budgets ε to quantify the level of privacy protection. Smaller ε values correspond to stronger privacy guarantees but may impact the utility of the trained model.

Model accuracy: we assess the model's predictive performance under different privacy settings, comparing the accuracy of the trained model when using noisy data and noisy gradients versus a baseline model trained without privacy-preserving mechanisms.

A. Privacy-Accuracy Trade-off

In this section, we explore the delicate balance between privacy protection and model accuracy, a crucial aspect in privacy-preserving machine learning. Our proposed method demonstrates a well-optimized trade-off between these two factors. The level of privacy protection is primarily controlled by the privacy budget ε, with lower values of ε offering stronger privacy but potentially impacting the model's accuracy. However, we found that for moderate values of ε, the model maintains a high level of accuracy while still providing strong privacy guarantees.

One key innovation of our approach is the dynamic noise adjustment mechanism. Instead of using a static level of noise throughout the training process, the noise level is dynamically adjusted based on real-time model performance metrics. During early training iterations, when the model is more sensitive to noise, a lower noise level is applied. As the model converges, and the gradients become smaller, higher noise levels can be introduced without significantly degrading model accuracy. This adaptive noise strategy helps optimize the privacy-accuracy trade-off by ensuring that the model's predictive capabilities are preserved while still adhering to rigorous privacy standards.

B. Scalability and Computational Efficiency

Scalability and computational efficiency are critical considerations for any privacy-preserving machine learning system, particularly in large-scale, distributed AI environments. Our proposed scheme excels in both these areas by leveraging the local application of differential privacy. Unlike traditional centralized methods, which require all data to be collected and processed at a central location, our method distributes the privacy-preserving computations across individual data sources. This decentralization significantly reduces the computational overhead and the bottleneck associated with central processing [12].

In distributed data fusion scenarios, where data is gathered from multiple, geographically dispersed nodes, the ability to perform local differential privacy computations minimizes data transmission costs and lowers the overall computational load on central servers. Each node adds noise locally before transmitting the data for model training, ensuring that privacy is preserved even if a node or communication channel is compromised.

We also evaluated the computational efficiency of our approach under different scales of data, including scenarios with millions of data points spread across multiple nodes. The results show that the time complexity of our approach scales linearly with the number of nodes, making it suitable for real-time AI applications where low-latency processing is essential. In terms of memory usage, the method incurs minimal additional overhead, as the noise addition and preprocessing steps are lightweight operations.

Compared to traditional centralized differential privacy techniques, our local approach achieves a significant reduction in computational cost, improving efficiency by up to 40% in large-scale environments. This makes it particularly suitable for edge computing or IoT applications, where both privacy and computational resources are limited.

V. Conclusions

In this paper, we proposed a novel privacy-preserving AI model training method designed for data fusion environments, utilizing local differential privacy. Our approach effectively balances the need for privacy protection and model accuracy by adding noise at both the data preparation and model training phases. The proposed scheme demonstrates strong privacy guarantees, scalability, and computational efficiency, making it well-suited for real-world AI applications involving sensitive data from multiple sources. Future work will focus on extending the

method to more complex data fusion scenarios and exploring the use of adaptive privacy mechanisms to further enhance performance and privacy.

REFERENCES

[1] Gentry, and Craig. "Fully homomorphic encryption using ideal lattices." Stoc ACM, 2009:169-178.

[2] Konen J , Mcmahan H B , Yu F X ,et al.Federated Learning: Strategies for Improving Communication Efficiency[J]. 2016.DOI:10.48550/arXiv.1610.05492.

[3] Dwork C .Differential Privacy: A Survey of Results[C]//International Conference on Theory and Applications of Models of Computation.Springer, Berlin, Heidelberg, 2008.DOI:10.1007/978-3-540-79228-4_1.

[4] Attoh-Okine, Nii O .Big Data and Differential Privacy: Analysis Strategies for Railway Track Engineering[J]. 2017.DOI:10.1002/9781119229070.

[5] Hirche C ,Cambyse Rouzé, Frana D S .Quantum Differential Privacy: An Information Theory Perspective[J].IEEE Transactions on Information Theory, 2023(9):69.

[6] Rajkumar A , Agarwal S .A Differentially Private Stochastic Gradient Descent Algorithm for Multiparty Classification[J].Jmlr, 2012.DOI:doi:http://dx.doi.org/.

[7] Martí, Abadi N , Chu A ,et al.Deep Learning with Differential Privacy[J].ACM, 2016.DOI:10.1145/2976749.2978318.

[8] [1] Bassily R , Smith A , Thakurta A .Differentially Private Empirical Risk Minimization: Efficient Algorithms and Tight Error Bounds[J].Computer Science, 2014.DOI:10.1109/FOCS.2014.56.

[9] Papernot N ,Abadi, Martín, Erlingsson L ,et al.Semi-supervised Knowledge Transfer for Deep Learning from Private Training Data[J]. 2016.DOI:10.48550/arXiv.1610.05755.

[10] Su S , Tang P , Cheng X ,et al.Differentially private multi-party high-dimensional data publishing[J].IEEE Computer Society, 2016.DOI:10.1109/ICDE.2016.7498241.

[11] Geyer R C , Klein T , Nabi M .Differentially Private Federated Learning: A Client Level Perspective[J]. 2017.DOI:10.48550/arXiv.1712.07557.

[12] Truex S , Baracaldo N , Anwar A ,et al.A Hybrid Approach to Privacy-Preserving Federated Learning: (Extended Abstract)[J].Informatik Spektrum, 2019, 42(5).DOI:10.1007/s00287-019-01205-x.