JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

DIFFERENTIAL PRIVACY TECHNIQUES IN MACHINE LEARNING FOR ENHANCED PRIVACY PRESERVATION

[1]Keyur Dodiya, [2]SarangKumar Radadia, [3]Deval Parikh

[1]System Engineer, [2]Principal/Associate Dir
Software Development/ Engineering, [3]Sr Software Engineer

Abstract : The increasing reliance on machine learning models has prompted growing concerns regarding the privacy of sensitive information used in the training process. As a result, using differential privacy techniques has become a viable paradigm for attaining strong privacy preservation without sacrificing the models' usefulness. This study investigates and summarises important approaches in the field of machine learning differential privacy. Adding controlled noise at various points in the machine learning pipeline is the first class of approaches. To avoid unintentionally revealing private information, Laplace and Gaussian noise are deliberately added to training data, predictions, and model parameters. By employing strategies like randomised response mechanisms, data perturbation can enhance privacy for each individual without compromising the model's quality. Collaborative model training is made easier by privacy-preserving aggregation techniques like Secure Multi-Party Computation (SMPC), which protects raw data. By adding noise to gradients during training, Differential Privacy Stochastic Gradient Descent (DP-SGD) provides privacy guarantees during the optimization stage. When differential privacy is combined with federated learning, it allows for decentralized model training across devices while maintaining the security and localization of sensitive data. By allowing computations on encrypted data or safely aggregating model updates, advanced cryptographic approaches like homomorphic encryption and secure aggregation protocols give another degree of privacy. When taken as a whole, these methods add to a thorough framework for machine learning differential privacy that strikes a balance between the need to protect individual privacy and the drive to create accurate models.

IndexTerms **– Differential Privacy, Noise, SGD, Federated Learning, Machine Learning.**

1. Introduction

The concept of differential privacy, which aims to describe privacy from a different perspective [1]. Differential privacy is a type of privacy that allows you to provide relevant information about a dataset without releasing any personal information about it [2]. It is a mathematical framework for ensuring the privacy of individuals in datasets. Since it permits data analysis without disclosing private information about each individual included in the dataset, it can offer a robust guarantee of privacy. It gains insights from large datasets while still maintaining privacy.

Differential privacy works by using an algorithm to add a controlled amount of randomness. It changes responses at a pre-determined frequency helps to protect the privacy of the participants. More noise increases privacy but reduces data accuracy, represented by the epsilon parameter (ε)[3]. The more noise added to original responses the more privacy is protected, but the less accurate the data becomes, it uses ε(epsilon)which is the privacy parameter of differential privacy; low ε, high accuracy, low accuracy.

It can be divided into two categories: local and global. Each has a unique strategy for protecting privacy while using machine learning and data analysis. In global differential privacy often referred to as centralised differential privacy, an analysis or computation, usually carried out on a centralised server, adds noise to the final result. The idea is to conceal information about any specific data point while maintaining the integrity of the model or overall statistics. In local differential privacy, before each individual data point is delivered to a data aggregator or analysis server, noise is introduced. Sensitive data is safeguarded even before it leaves the user's device [4] since each user adds noise locally to their own data.

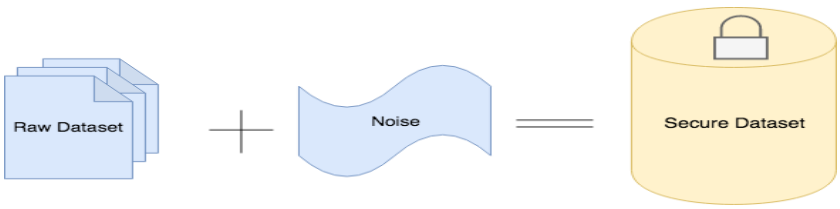

Figure 1.1: Differential Privacy [18]

2. Differential Privacy Techniques

Differential privacy is a framework that offers robust privacy protection by making sure that adding or removing a specific data point doesn't materially affect the result of a calculation or analysis. For those whose data is included in a dataset, it offers a robust guarantee of privacy. Enabling the extraction of meaningful information from the data while reducing the possibility of disclosing private information about any particular person is the goal [5]. Differential privacy strategies are useful in machine learning because they maintain a balance between the need to protect sensitive personal data and the development of accurate models [6]. The following are some crucial machine learning strategies for enhanced privacy preservation:

2.1 Noise injection

Noise injection, namely Laplace and Gaussian noise, is a necessary method to protect personal privacy in the context of differential privacy while maintaining the utility of machine learning models.

1. Laplace noise is a widely used technique that adds noise to the model's output or to the model's parameters during training. The sensitivity of the function being computed and a privacy parameter called "epsilon" (ε) are required for the calibration of this noise. A lower value of ε indicates a greater degree of privacy protection. The distinctive fat tails of the Laplace distribution add unpredictability that effectively masks the contribution of individual data points [7].
2. Gaussian noise is also used as a substitute for Laplace noise, providing comparable advantages in terms of maintaining anonymity [8]. During the machine learning process, Gaussian noise can be added to the model parameters or predictions. Similar to Laplace noise, the privacy parameter ε and the sensitivity of the function must be taken into account when calibrating Gaussian noise. The Gaussian distribution, which is distinguished by its bell-shaped curve, evenly distributes noise and is especially useful when a more widely dispersed noise profile is required.

An external observer would find it difficult to determine the impact of any one data entry on the overall model output in either scenario due to the meticulous noise calibration that protects the privacy of individual data points. In the constantly changing field of privacy-preserving data analytics, these noise injection approaches are crucial to striking a careful balance between the requirements of privacy protection and the usefulness of machine learning models.

2.2 Perturbation of data

Perturbation of data in the context of privacy preservation involves two key techniques: data perturbation and randomized response.

1. Data perturbation: Addition of noise directly to the input data before training a model. One way to accomplish this is by applying changes to individual data points or adding random variants to the training set. The idea is to introduce deliberate randomness into the dataset in order to mask particular information that might be used to identify individual records. The privacy of individual contributors is improved by perturbing the training data, which makes it harder for outside parties to identify sensitive information while still enabling useful model training. The degree of disturbance is frequently meticulously adjusted to achieve a balance between data utility and privacy protection.
2. Randomised Response: In the context of classification issues, randomised response protects the genuine labels of training data while maintaining anonymity. This method uses a randomised response mechanism to disturb the labels. A randomization approach is used, adding noise and ambiguity to the labelling process in place of disclosing the true label. This makes sure that even when the model is being trained, the true label of each individual data point is kept safe. When handling sensitive data in classification tasks [9], randomised response is especially helpful as it adds a layer of anonymity without sacrificing the machine learning model's overall efficacy and accuracy.

The major goal of both data perturbation and randomised response is to intentionally add noise and unpredictability while protecting individual privacy during training and preserving the integrity and usefulness of the data for efficient machine learning model building. These methods add to the larger body of work on privacy-preserving approaches for data-driven applications.

2.3 Privacy-Preserving Aggregation: Secure Multi-Party Computation (SMPC)

In the field of privacy-preserving aggregation, Secure Multi-Party Computation (SMPC) is a potent technique that allows several parties to work together to jointly compute a function over their inputs while protecting the confidentiality of those inputs [10]. By using SMPC, it makes sure that no one finds out more about the contributions of the other parties than what is necessary to understand the final combined outcome [11]. This is especially important when it comes to machine learning, when it's necessary to aggregate parameters or model updates from several sources without disclosing the raw data [12].

Parties participate in cryptographic protocols inside the SMPC framework, which enable them to jointly compute a desired function while keeping their individual inputs secret. Every partner maintains their own private data, and computations are done in a way that protects privacy [13]. The computing of a function over each party's private inputs, with no party having access to the full set of inputs, is the final product [14].

This method is useful for gathering model updates in a way that protects privacy. For instance, SMPC can be used to aggregate local model updates without disclosing the raw data from each device in federated learning scenarios where models are taught across

decentralised devices. By doing this, confidentiality is preserved and all parties' contributions are reflected in the aggregated model without jeopardising personal information.

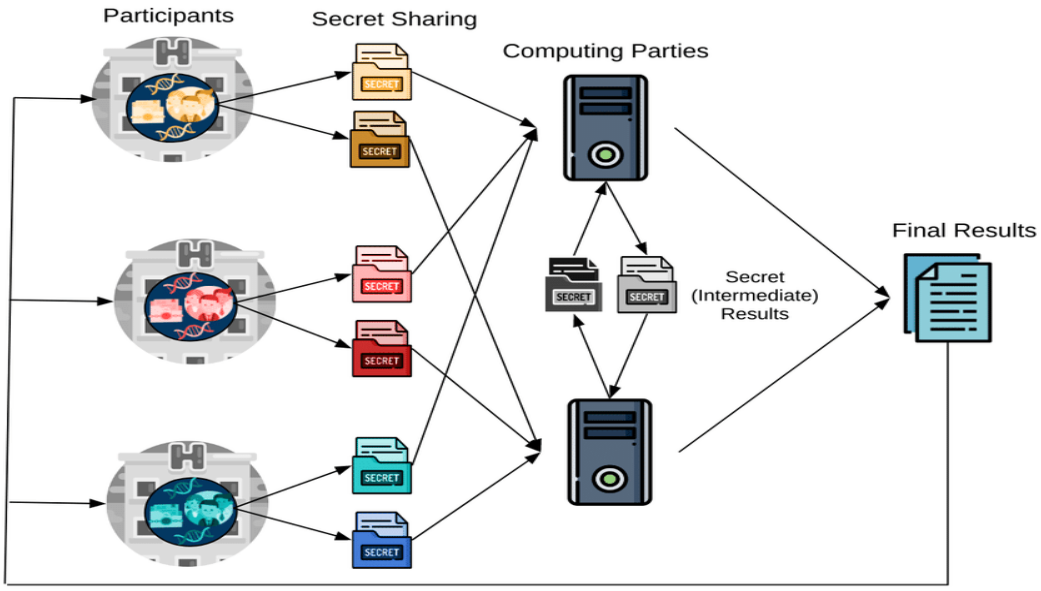

Figure 2.3 Secure Multi-Party Computation (SMPC) [20]

2.4 Differentially Private Stochastic Gradient Descent (DP-SGD):

A privacy-preserving optimisation technique called Differentially Private Stochastic Gradient Descent (DP-SGD) incorporates differential privacy ideas into the machine learning model training process. It does this by adding precisely calibrated noise to the gradients that stochastic gradient descent (SGD) was used to generate during the training phase. The main objective is to provide differential privacy in the changes to the model parameters, which implies that the impact of each individual training data point on the training output is restricted.

For each iteration of the SGD optimisation procedure, noise is added to the gradients in DP-SGD. Based on the gradients' sensitivity and a privacy parameter, commonly represented by epsilon (ε), the quantity of noise added is calculated. Although there may be a trade-off with the model's usefulness, a smaller ε offers more privacy.

This strategy is very helpful in industries like healthcare or finance, where privacy is of utmost importance and sharing sensitive training data can be risky [14]. A compromise between maintaining privacy and improving model accuracy can be achieved by using DP-SGD to enable collaborative model training on distributed datasets without disclosing contributor identities [15].

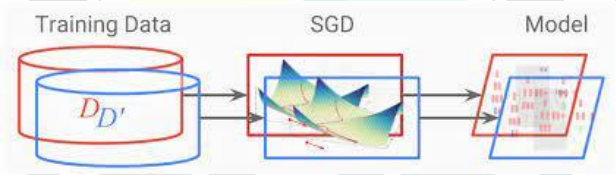

Figure 2.4 Differentially Private Stochastic Gradient Descent [18]

2.5 Federated Learning

Federated Learning is a decentralised machine learning technique that eliminates the need to centralise raw data and allows model training across numerous edge devices or servers [16]. Federated Learning with Differential Privacy, or FL-DP, is an expansion of federated learning that further protects user privacy during the model aggregation process by incorporating the ideas of differential privacy.

The federated learning paradigm distributes the training process across local devices rather than transferring raw data to a central server for model training. Based on its local data, each device computes a model update; only the model updates are transmitted to the central server. The global model is updated by the central server aggregating these modifications. To make improvements to the overall model, this process is performed iteratively without disclosing specific data [16].

In FL-DP, differential privacy is added by adding noise to the model updates during the aggregation stage. This noise guarantees some degree of privacy protection by obscuring the contributions of individual devices. Like other differential privacy approaches, the quantity of noise supplied is regulated by the privacy parameter epsilon (ε). Higher privacy assurances are achieved with a smaller ε, but the utility of the aggregated model may suffer [2].

2.6 Objective Perturbation:

Objective perturbation is a technique used in machine learning to enhance privacy by introducing noise directly into the optimization objective function during the training process [17]. Achieving differential privacy—that is, limiting the influence of a

given data point's existence or absence on the final model parameters—is the main objective. This method is especially useful in situations where it is necessary to prevent sensitive information from being revealed by the optimisation target itself [21].

An optimisation algorithm aims to minimise or maximise an objective function that measures the discrepancy between the true values in the training data and the model predictions when a machine learning model is being trained [22]. The process of adding precisely calibrated noise to this objective function is known as objective perturbation. Privacy parameters that regulate the degree of privacy protection, like epsilon (ε), are used to calculate the quantity of noise generated.

Objective perturbation adds to the larger context of differential privacy in machine learning by causing perturbations to the objective function. This method is particularly helpful in scenarios where more conventional differential privacy techniques, like adding noise to gradients or model parameters, might not be appropriate or sufficient.

2.7 Advanced Cryptographic Techniques:

Homomorphic Encryption: This advanced cryptographic method allows calculations to be done on encrypted material without having to first decrypt it. Homomorphic encryption permits model training on encrypted data in the context of privacy-preserving machine learning, guaranteeing that sensitive data is kept private at all times. When sharing raw data is impractical and data privacy is a top priority, this method comes in handy. There are various variations of homomorphic encryption, including fully and partially homomorphic encryption, which offer varying degrees of computational power while preserving data security [23].

Secure Aggregation methods: To safely aggregate model updates from several parties without disclosing the individual updates, secure aggregation methods employ cryptographic techniques. In cases of collaborative learning, like federated learning, where decentralised devices or servers contribute model updates, safe aggregation makes assurance that the aggregated model is updated without disclosing the individual contributions from each participant [24]. This aids in maintaining secrecy and privacy, particularly when handling sensitive datasets that are dispersed among several organizations [25]. To accomplish secure aggregation, a variety of cryptographic primitives can be used, such as secret sharing and secure multi-party computation [26].

3. Comparison of Privacy-Preserving Techniques in Machine Learning

Table 3.1 Comparison of different techniques [27-34]

Techniques	Challenges	Merits	Demerits
Laplace Noise	• Determining optimal noise scale (epsilon) • Balancing privacy and model accuracy • Ensuring appropriate sensitivity calculation	• Simplicity of implementation • Provides strong privacy guarantees	• May not scale well with high-dimensional data • Sensitivity to noise
Gaussian Noise	• Choosing the right noise distribution • Balancing privacy and model accuracy • Impact on the convergence of optimization	• Smooth perturbation of model parameters • Effective in continuous and differentiable models	• May not achieve strong privacy guarantees • Effective in continuous and differentiable models
Data Perturbation	• Determining optimal noise level • Balancing privacy and model accuracy • Impact on model training convergence	• Directly perturbs input data, preserving privacy • Compatible with various machine learning models	• May distort data distribution and patterns • Sensitivity to noise
Randomized Response	• Choosing appropriate response probabilities • Handling imbalances in label distribution • Balancing privacy and classification accuracy	• Protects individual labels in classification	• May introduce bias in label perturbation
Secure Multi-Party Computation (SMPC)	• Communication overhead in multi-party setup • Ensuring integrity of computations • Handling malicious participants	• Enables secure aggregation of model updates • Suitable for distributed environments	• Complexity in setup and coordination • Increased computational and communication costs
Federated Learning	• Communication overhead in decentralized setup • Ensuring model convergence across devices	• Privacy preservation in collaborative learning • Utilizes local data without sharing raw data	• Challenges in handling non-IID data distribution • Increased communication and computation costs

	• Privacy amplification across iterations		
DP-SGD	• Tuning noise hyperparameters • Convergence challenges in optimization • Balancing privacy and model accuracy	• Privacy during model training • Compatible with various machine learning models	• Trade-off between privacy and model accuracy • Sensitivity to noise
Objective Perturbation	• Choosing appropriate noise in objective • Balancing privacy and model accuracy • Impact on optimization convergence	• Directly incorporates privacy in optimization • Compatible with various optimization algorithms	• May hinder model convergence • Sensitivity to noise
Homomorphic Encryption	• Performance overhead in computation • Ensuring secure key management • Balancing privacy and model accuracy	• Enables computations on encrypted data • Strong privacy guarantees	• Computational complexity and speed limitations • Limited support for certain types of operations
Secure Aggregation Protocols	• Ensuring secure communication channels • Handling malicious participants • Balancing privacy and model accuracy	• Protects individual model updates • Protects individual model updates	• Increased computational and communication costs • Complexity in setup and coordination

4. CONCLUSION

In conclusion, differential privacy techniques in machine learning offer a robust framework for preserving privacy while maintaining model accuracy. Advanced cryptographic techniques include homomorphic encryption, federated learning, DP-SGD, data perturbation, noise injection, and privacy-preserving aggregation are some of the methods that lead to improved privacy protection. The particular use case, the type of data, and the required level of privacy all influence the technique selection. The publications that are cited offer further details on these methods, but in order to learn about the most recent developments in privacy-preserving machine learning, one must keep up with the area as it is always changing.

REFERENCES

[1] Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., & Naor, M. (2006). Our data, ourselves: Privacy via distributed noise generation. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (pp. 486-503).
[2] McSherry, Frank Dwork, Cynthia. "Calibrating noise to sensitivity in private data analysis." Theory of Computing 4, 120-149, 2007.
[3] Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy.
[4] Li, Ninghui, Tiancheng Li, and Ting Wang. "T-closeness: Privacy beyond k-anonymity and l-diversity." In International Conference on Data Engineering, pp. 55-64. IEEE, 2007
[5] Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016). Deep Learning with Differential Privacy. In ACM CCS.
[6] Li, Ninghui, Xiaokui Sun, and Tiancheng Li. "On the tradeoff between privacy and utility in k-anonymization." ACM Transactions on Information and System Security (TISSEC) 14.1 (2011): 1-34.
[7] Rebollo-Monedero, Daniel, Francisco Javier Ortega, and Carlos A. Cuevas-Recuerda. "Differential privacy in machine learning: A survey." arXiv preprint arXiv:1809.02530 (2018).
[8] Balle, Moussa, et al. "Privacy-preserving learning with additive noise." Foundations of Computer Science, 2012. IEEE 53rd Annual Symposium on. IEEE, 2012.
[9] Warner, S. L. (1965). Randomized response: A survey technique for eliminating evasive answer bias. Journal of the American Statistical Association, 60(309), 63-69.
[10] Mohassel, Peyman, and Andrew Shulman-Blum. "Secure multi-party computation." SIAM Journal on Computing 37.1 (2007): 38-87.
[11] Beimel, Amos, Yevgeniy Dodis, and Marten van den Bosch. "Perfect multi-party computation on circuits." SIAM Journal on Computing 39.5 (2010): 1773-1808.
[12] Phong, Le Viet et al. "Privacy-preserving federated learning: A comprehensive survey." arXiv preprint arXiv:2007.05959 (2020).
[13] Koutsos, Anna et al. "Secure aggregation for distributed learning." Proceedings of the 2015 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2015.
[14] Xu, R., et al. (2021). A survey on differentially private machine learning in healthcare. arXiv preprint arXiv:2108.05467.
[15] Shokri, R., & Shmatikov, V. (2015). Privacy-preserving deep learning. Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, 1310-1321.

[16] McMahan, Brendan, et al. "Federated learning: Machine learning on mobile devices." arXiv preprint arXiv:1607.03495 (2016).
[17] Chaudhuri, Kamalika, et al. "Differentially private empirical risk minimization." Journal of Machine Learning Research 12.Mar (2011): 1069-1109.
[18] https://medium.com/secure-and-private-ai-writing-challenge/differential-privacy-e5c7b933ef9e
[19] https://yanlitao.github.io/fastDP/
[20] Torkzadehmahani, Reihaneh & Nasirigerdeh, Reza & Blumenthal, David & Kacprowski, Tim & List, Markus & Matschinske, Julian & Späth, Julian & Wenke, Nina & Bihari, Béla & Frisch, Tobias & Hartebrodt, Anne & Hauschild, Anne-Christin & Heider, Dominik & Holzinger, Andreas & Hötzendorfer, Walter & Kastelitz, Markus & Mayer, Rudolf & Nogales, Cristian & Pustozerova, Anastasia & Baumbach, Jan. (2020). Privacy-preserving Artificial Intelligence Techniques in Biomedicine.
[21] Kifer, Daniel, et al. "Private convex empirical risk minimization and high-dimensional regression." Proceedings of the 22nd international conference on Neural Information Processing Systems (NIPS'09). Curran Associates Inc., 2009.
[22] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.
[23] Gentry, Craig. "Computing arbitrary functions on encrypted data." Communications of the ACM 53.4 (2010): 50-60.
[24] Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., ... & Velingker, A. (2019). Towards federated learning at scale: System design. arXiv preprint arXiv:1902.01046.
[25] Kairouz, Peter, Ofer Dekel, Ahmad Shamir, and T-H. Hubert Chan. "Secure aggregation using secret sharing." In EUROCRYPT 2020, pp. 202-234. Springer, Berlin, Heidelberg, 2020.
[26] Mohassel, Pooya, and Peter Rindal. "Pink: Private instantiation of k-means clustering." In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 270-285. ACM, 2017.
[27] McMahan, H. B., Ramage, D., Talwar, K., & Zhang, L. (2017). Learning Differentially Private Recurrent Language Models. In ICLR.
[28] Duchi, J. C., Jordan, M. I., & Wainwright, M. J. (2013). Privacy aware learning. In Proceedings of the 30th International Conference on Machine Learning (ICML-13).
[29] Boenisch, Thomas, Florian Kerschbaum, and Martin Nöckel. "Secure multi-party computation with resource-constrained mobile devices." ACM SIGSAC Symposium on Principles of Distributed Computing. 2016.
[30] Bassily, R., Smith, A., & Thakurta, A. (2014). Private empirical risk minimization: Efficient algorithms and tight error bounds. In Proceedings of the forty-sixth annual ACM symposium on Theory of computing (STOC).
[31] Gentry, C. (2009). A Fully Homomorphic Encryption Scheme. Stanford University.
[32] Hardt, M., Talwar, K., & Ullman, J. (2012). On the geometry of differential privacy. In Proceedings of the 44th symposium on theory of computing (pp. 705-724).
[33] Yao, A. C. (1982). Protocols for secure computations. In Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science (pp. 160-164).
[34] Domingo-Ferrer, J., & Mateo-Sanz, J. M. (2011). Practical data-oriented microaggregation for statistical disclosure control. IEEE Transactions on Knowledge and Data Engineering, 23(5), 703-716.