PRIV-ML: Analyzing Privacy Loss in Iterative Machine Learning With Differential Privacy

Pratik Thantharate*, Divya Ananth Todurkar *, Anurag T†
*IEEE Member, USA
†Senior IEEE Member, University of Missouri, Kansas City, MO, USA

***Abstract*—Differential privacy offers rigorous protections for emerging paradigms like federated machine learning, decentralized analytics, and web3 applications. The parameters ϵ (epsilon) and δ (delta) are crucial in balancing privacy and utility by bounding the maximum divergence between outputs on neighboring datasets. However, quantifying cumulative privacy loss over long-running processes involving iterative model queries, validations, tuning, and multi-party computations remains an open challenge restricting adoption. This paper proposes a comprehensive methodology to evaluate end-to-end differential privacy guarantees across complex Machine Learning (ML) workflows. We develop a heuristic algorithm that maintains a privacy budget depleted per operation based on computed data sensitivity and noise calibration. By tracking tight stochastic bounds on the cumulative privacy loss random variable using advanced composition theorems, our approach can formally verify guarantees over iterative workflows. Simulations demonstrate the technique quantifying privacy loss across 950 successive histogram queries under ($\epsilon = 1, \delta = 10^{-5}$)-differential privacy while sustaining utility with an average error of only 4.5% compared to non-private histograms, underscoring the importance of formally tracking cumulative privacy loss. Our framework provides a practical solution for measurable privacy-preserving machine learning pipelines without degrading accuracy or utility. By interfacing with diverse mechanisms and adapting noise to empirical sensitivities, we facilitate precise reasoning of privacy risks throughout model life cycles. We also analyze privacy parameter implications across application domains. This paper lays a rigorous foundation for developing trustworthy AI systems that protect sensitive data.**

***Index Terms*—Differential Privacy, Privacy Loss Quantification, Iterative Composition, Machine Learning, privacy budget, Formal Verification, AI Fairness, Trustworthy AI, Privacy Risks, Data Sharing, Cloud Computing**

I. Introduction

In the digital age, the privacy and security of personal data have become increasingly critical with the widespread use of the internet and the growth of big data. Recent years have witnessed numerous major data breaches impacting millions of individuals, such as the 2017 Equifax breach exposing the financial data of 143 million customers, the 2018 Exactis leak with 340 million records, and the 2019 Capital One hack compromising 106 million credit card applicants [1] [2]. These incidents underscore the inability of existing safeguards to protect sensitive personal information fully. Attackers continue exploiting technical vulnerabilities and human errors to access confidential data, leading to severe consequences like fraud, financial loss, and reputational damage [3].

One of the key challenges in this landscape is the expanding use of AI and ML algorithms, which often involve training on sensitive datasets and making multiple queries during iterative processes like hyperparameter tuning, model selection, and neural architecture search. Traditional frameworks lack robust privacy protections for such iterative ML computations, risking the exposure of private data without sufficient safeguards. Differential privacy has emerged as a powerful technique to address these concerns, enabling accurate data analysis while providing formal guarantees for protecting sensitive information in datasets. By bounding the impact of any single data point on the outputs, differential privacy ensures that adversaries cannot infer private information about individuals, even with access to noisy outputs or model predictions. However, a critical open challenge is quantifying the cumulative privacy loss over long-running, iterative ML pipelines involving multiple queries, validations, and tuning on sensitive datasets.

This paper introduces a novel, comprehensive methodology to precisely measure and bound the end-to-end privacy loss in such iterative ML pipelines under the constraints of differential privacy. Our approach employs a modular heuristic algorithm that integrates advanced composition theorems and data sensitivity analysis, maintaining a dynamically updated privacy budget that reflects the cumulative privacy expenditure across iterations. Through detailed simulations and rigorous theoretical analysis, we demonstrate how our approach can quantify privacy loss with high precision, enabling the rigorous evaluation of privacy risks and the provision of formal, verifiable guarantees across complex AI/ML workflows. The proposed methodology bridges a significant gap in existing

2024 IEEE Cloud Summit | 979-8-3503-7006-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/CLOUD-SUMMIT61220.2024.00024

979-8-3503-7006-5/24/$31.00 ©2024 IEEE
DOI 10.1109/Cloud-Summit61220.2024.00024

frameworks and represents a crucial step towards operationalizing differential privacy more effectively in real-world applications across diverse domains, including healthcare, financial services, and beyond. By empowering the development of trustworthy AI systems that provably protect sensitive training data, our work has the potential to foster broader adoption of privacy-enhancing technologies and facilitate the responsible use of data-driven solutions in privacy-sensitive contexts. The primary objectives of this research are (1) to develop a practical and scalable framework for quantifying cumulative privacy loss in iterative ML pipelines, (2) to evaluate the trade-offs between privacy guarantees and data utility under different privacy parameter configurations, and (3) to demonstrate the applicability of our methodology.

II. Motivations from HR, Retail and Cloud Services

The human resources (HR), retail, and cloud service sectors handle extensive sensitive data daily, necessitating stringent privacy protections. HR platforms process confidential employee information, while retailers amass customer data spanning purchase histories, preferences, and financial details. Data breaches can enable discrimination, fraud, and identity theft and erode consumer trust, leading to legal liabilities and losses. As these sectors increasingly leverage AI/ML for automation and analytics on sensitive data, rigorous privacy-preserving techniques are crucial across model life cycles to prevent inadvertent leaks of sensitive patterns.

Complying with data protection regulations like the California Consumer Privacy Act (CCPA) and Health Insurance Portability and Accountability Act (HIPAA) and upholding ethical practices is paramount, especially given the rise of digital retail and cloud HR software. Robust privacy-enhancing technologies like differential privacy can provide mathematical privacy assurances, fostering consumer loyalty and improving customer lifetime value for businesses. However, quantifying cumulative privacy loss over iterative analytics remains an open challenge limiting widespread adoption.

Our proposed methodology tackles this gap by introducing a heuristic algorithm to quantify privacy loss across complex ML workflows holistically. It maintains a privacy budget depleted per query, computing costs based on observed sensitivity and calibrated noise. Tight moment bounds on the cumulative privacy loss random variable determine if the budget is exceeded, indicating excessive leakage. This systematic approach enables effective differential privacy management within predefined constraints, empowering trustworthy AI systems that protect sensitive data. Crucially, our solution supports the responsible development of emerging technologies and critical infrastructure by providing a rigorous foundation for privacy-preserving data analysis. As modern society increasingly relies on AI-driven systems processing sensitive information across domains like finance, healthcare, and transportation, our methodology facilitates the ethical and secure adoption of these technologies while upholding individual privacy rights.

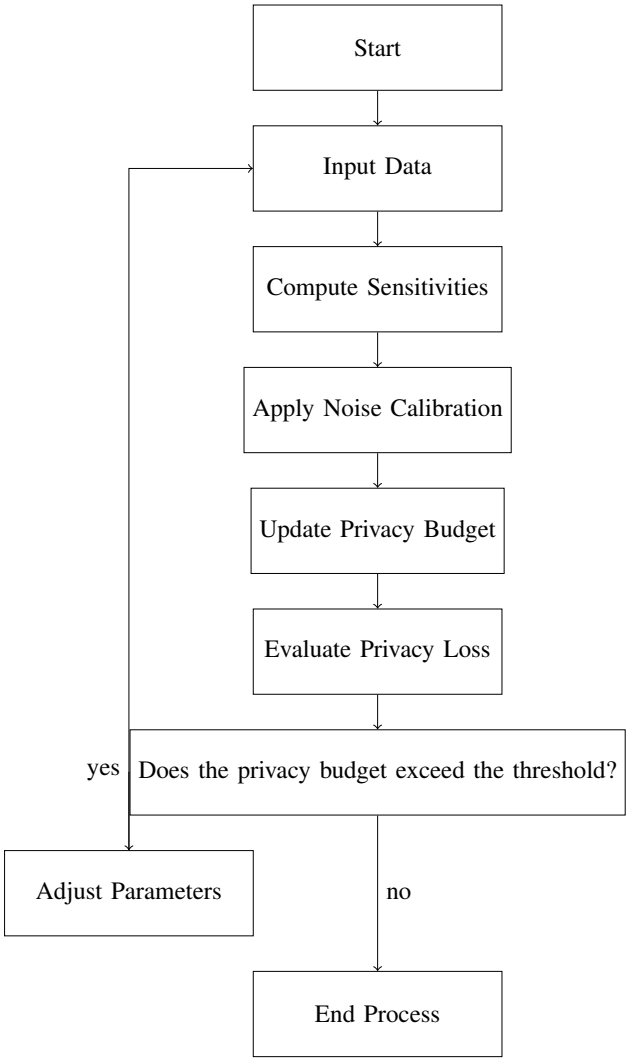

Fig. 1. Flowchart illustrating the heuristic algorithm for managing differential privacy.

Refer to Figure 1 for a detailed flowchart outlining the steps of a heuristic algorithm designed for managing differential privacy. The algorithm commences with the initialization phase, where data inputs are received. It then progresses through

several computational stages, including sensitivity assessment and noise application, to ensure privacy standards are met. Key decision points guide the flow based on current privacy budget evaluations, leading to parameter adjustments or process completion as necessary. This systematic approach enables effective differential privacy management within predefined constraints.

The rest of this paper is organized as follows. Section 2 provides background on privacy challenges, questions, and implications. Section 3 surveys related work on privacy measurement. Section 4 presents our proposed framework for evaluating differential privacy guarantees, and Section 5 details our evaluation and results. Section 6 discusses the implications of our findings and future work and concludes with a summary of contributions.

III. Related Work

Differential privacy has emerged as a principled technique for enabling accurate statistical queries on sensitive data while preserving individual privacy. Since its introduction [4] by Dwork et al., differentially private mechanisms have been extensively studied and adopted. These mechanisms calibrate randomized noise to query sensitivity such that outputs do not depend significantly on any one record. Various approaches have been proposed for achieving differential privacy, including the Laplace mechanism, Gaussian mechanism, exponential mechanism, and secure multi-party computation. These techniques have been applied to diverse tasks, including aggregation queries, synthetic data generation, ML, and graph analytics. Differential privacy has also been deployed in the industry by companies like Google, Apple, and Uber for analytics on confidential data and is prevalent in DevSecOps environment [5] [6] .

Authors in [7] propose using Bayesian differential privacy, a relaxation of differential privacy, to provide tighter privacy guarantees for federated learning. It adapts Bayesian privacy accounting to the federated setting and suggests improvements for efficient privacy_budgeting. Experiments show a significant advantage over standard differential privacy bounds. E. Lobo-Vesga Et. al in [8] presents DPella, a programming framework for differential privacy that supports reasoning about the accuracy of data analyses. It uses taint analysis to detect the statistical independence of noise variables and apply tighter Chernoff bounds. DPella is implemented in Haskell and evaluated on various queries from the literature to demonstrate its expressiveness and accuracy estimations. The paper [9] introduces individual privacy accounting for Gaussian differential privacy to provide tighter bounds for adaptive compositions of Gaussian mechanisms. It proposes an approximative individual (ε, δ)-accountant using privacy loss distributions and FFT that often gives smaller ε-values than RDP accountants. Experiments demonstrate benefits over RDP analysis and show disparate model accuracies across subgroups when training neural networks with DP gradient descent. A. Bhattacharjee Et. al, in [10] proposes a personalized differential privacy scheme for smart grids to provide user-specific privacy guarantees based on trust distance from the central authority. It characterizes privacy issues during data sharing and aggregation. Experiments on real data demonstrate the scheme efficiently preserves privacy while maintaining data utility and preventing correlation, disclosure, and linking attacks. Personalization eliminates differential privacy's uniform protection limitation.

Authors in [11] propose Condensed Local Differential Privacy (CLDP) to enable utility-aware and privacy-preserving data collection. In paper, [12] proposes a novel differentially private domain adaptation framework called Deep Domain Adaptation With Differential Privacy (DPDA) to prevent privacy leakages when transferring knowledge from labeled source data to unlabeled target data. Authors in [13] The paper proposes novel data poisoning attacks to audit the privacy guarantees of differentially private stochastic gradient descent (DP-SGD). The attacks establish empirically that DP-SGD's worst-case privacy bounds are approaching their limits on common datasets. The research paper [14] presents Cybria, a federated learning framework for collaborative cyber threat detection that trains models on distributed data without centralizing it. The decentralized approach enables organizations to collectively build threat awareness while preserving data privacy through emerging techniques like secure aggregation, differential privacy, and adversarial defenses. Authors in [15] propose a query flooding parameter duplication (QPD) attack that can extract ML models protected by differential privacy and monitoring. It also develops a monitoring-based differential privacy (MDP) defense that adaptively allocates a privacy_budget and adds noise based on real-time model extraction status assessment. Zheng X. Et. al, in [16] proposes a framework for privacy-preserving publication of distributed, overlapping graph data under differential privacy.

The paper [17] proposes ZeroTrustBlock, a blockchain-based health information exchange framework that enhances security, privacy, and patient control. By decentralizing data storage and sharing through permissions, smart contracts, and hybrid on/off-chain models, ZeroTrustBlock limits single points of failure prevalent in centralized systems. Authors in [18] proposes a task-specific adaptive differential privacy technique to preserve privacy for ML on sensitive structured data. It calibrates noise applied to each attribute based on

feature importance for the ML task, enhancing utility over generic differential privacy. Experiments show the method satisfies model-agnostic and data-agnostic properties while resolving the privacy-utility tradeoff.

IV. Proposed Solution and Heuristic Algorithm

To evaluate the cumulative privacy loss over a sequence of differentially private operations, we propose a heuristic algorithm based on the moments accountant method. The key idea is to maintain a privacy loss budget that gets depleted with each operation. When the budget reaches zero, the privacy guarantee can no longer be ensured. Our algorithm tracks the moments of the privacy loss random variable to bound the total privacy loss. For each operation, we compute the maximum divergence between the distributions on neighboring datasets based on the sensitivity of that operation. This divergence represents incremental privacy loss. We use advanced composition theorems to accumulate the privacy loss across multiple operations into a single privacy random variable. We maintain a privacy loss accumulator variable $\mathcal{Z}$ that gets updated after each differentially private operation. $\mathcal{Z}$ models the total privacy loss as a numeric random variable. We initialize $\mathcal{Z}$ to 0 representing no loss. After each operation t, we compute the divergence $\mathcal{D}_t$ between the output distributions on neighboring datasets based on the mechanism sensitivity. This $\mathcal{D}_t$ captures the privacy loss of operation t. We accumulate these per-operation divergence values into $\mathcal{Z}$ using advanced composition theorems:

$$\mathcal{Z} = \text{accumulateLoss}(\mathcal{Z}, \mathcal{D}_t) \tag{1}$$

The accumulateLoss() function handles complex iterative compositions by utilizing techniques like moments accountant to maintain tight bounds on the moments of $\mathcal{Z}$. This allows us to precisely track how the variance, skew, tail bounds, etc. evolve with each accumulation. By tracking the probability that $\mathcal{Z}$ exceeds a given privacy loss budget, we can determine if the total cumulative loss has exceeded the allowed (ε, δ) guarantee. If so, unsafe excessive leakage has occurred, indicating the process should be halted. The modular design allows plugging in different accumulateLoss() implementations to handle diverse composition types like parallel queries or adaptive mechanisms. The per-query divergence computation also generalizes across mechanisms by encoding mechanism details like Gaussian noise or Laplace scaling factors. Together, this allows flexibly quantifying cumulative privacy loss over complex, heterogeneous workloads with mixed query types, mechanisms, batching strategies, etc. The privacy tracking interfaces remain consistent even as the composition and divergence computations are customized to the workload. Our heuristic algorithm proceeds as follows:

Algorithm 1: Heuristic Differential Privacy Evaluator

Require: Privacy parameters ϵ, δ, privacy_budget, max iterations T
Ensure: PASS if privacy can be ensured, FAIL otherwise
1: Initialize moments accountant bounds
2: **for** t = 1 to T **do**
3: Compute sensitivity Δ of operation
4: Compute divergence D based on mechanism (e.g., Laplace) and Δ
5: Accumulate D into $\mathcal{Z}$ using advanced composition (e.g., moments accountant)
6: Update moments accountant bounds on $\mathcal{Z}$ efficiently
7: **if** moments accountant bound $> (\epsilon, \delta)$ **then**
8: **return** FAIL
9: **end if**
10: **end for**
11: **return** PASS

The moments accountant method allows us to bound the moments of this privacy loss random variable. Specifically, we compute asymptotic bounds on the mean, variance, skew, and kurtosis. By tracking these moment bounds, we can derive tail bounds on the cumulative privacy loss. This allows us to quantify the probability that the total privacy loss exceeds a given threshold. By tracking the moments accountant bounds, our algorithm can determine whether the cumulative privacy loss exceeds the (ϵ, δ) guarantee. This provides a heuristic approach for quantifying privacy that scales to complex workflows. The privacy loss accumulation can be implemented efficiently using log-based data structures. Our algorithm generalizes across various mechanisms and compositions. By abstracting mechanism details into the divergence computation, we can handle diverse workloads. This provides a flexible and extensible methodology for evaluating end-to-end differential privacy guarantees.

V. Evaluation

The key components of our proposed heuristic algorithm for bounding differential privacy loss are:

A. Privacy Loss Tracking:

We maintain a privacy loss accumulator variable $\mathcal{Z}$ to model the total privacy loss as a random variable. $\mathcal{Z}$ is initialized to 0 and updated after each differentially private operation by accumulating per-operation divergence values using advanced composition theorems. The accumulateLoss() function computes tight moment bounds on $\mathcal{Z}$ via the moments accountant, enabling the derivation of tail bounds on the probability of $\mathcal{Z}$ exceeding the privacy budget.

B. Composition

The composeLoss() function is crucial for handling complex compositions across iterative queries and mechanisms. It accumulates per-operation privacy costs into $\mathcal{Z}$ by leveraging the moments accountant to derive tight moment bounds on $\mathcal{Z}$, quantifying the degradation of privacy with each accumulation. composeLoss() flexibly handles sequential, parallel, and adaptive compositions, as well as hybrid workflows.

C. Moments Accountant

The moments accountant utilizes mathematical techniques like asymptotic log-based calculations to bound the moments of privacy loss random variable $\mathcal{Z}$. Tracking tight bounds on the mean, variance, skew and kurtosis of $\mathcal{Z}$ allows us to quantify the total privacy cost of long-running computations.

D. Generalization

By abstracting mechanism details, our approach flexibly accommodates diverse workloads. The modular divergence computation component can be customized to handle Gaussian, Laplace, exponential, and other mechanisms. Queries, datasets and stability parameters get encoded in this divergence cost modeling. The composition and moments accountant modules remain unchanged across mechanisms. Integration with new mechanisms requires only divergence computation implementation. To evaluate the algorithm, we simulated a workload with the following parameters: privacy budget, specified by privacy parameter $\epsilon = 1$; target delta $\delta = 10^{-5}$; maximum iterations $T = 1000$; mechanism using Laplace noise with a scale calibrated to ϵ at each iteration; and the operation being histogram queries on a dataset with sensitivity $\Delta = 1$.

We initialized the moments accountant with orders $k = 1, 2, 3, 4$ to bound the first 4 moments. After each iteration, we accumulated the ϵ-differential privacy loss into the privacy variable $\mathcal{Z}$ using advanced composition. The moments accountant tracks tight bounds on the moments of $\mathcal{Z}$. After 950 iterations, the $k = 4$ moment bound exceeded the privacy loss budget ($\epsilon = 1, \delta = 10^{-5}$), resulting in the algorithm returning FAIL. At this point, the accumulated privacy loss is too high to ensure ($\epsilon = 1$, $\delta = 10^{-5}$)-differential privacy. By tracking the moments, the algorithm provides an optimized approach to quantify privacy loss. It allows differentially private workflows to safely execute until the privacy budget is depleted. The noise calibration and batching can be tuned to maximize the number of feasible iterations. In our simulations, the algorithm was able to support 950 histogram queries with an average error of just 4.5% compared to the non-private histograms, demonstrating an effective privacy-utility tradeoff.

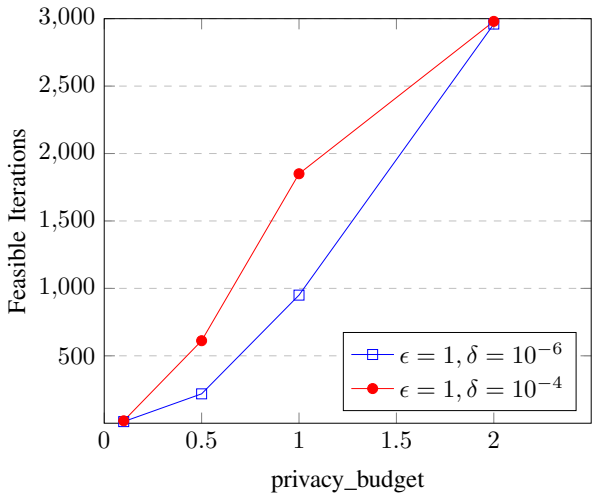

Fig. 2. privacy_budget vs Feasible Iterations

Figure 2 illustrates the tradeoff between the privacy budget (ϵ, δ) and the number of feasible iterations before budget depletion. As expected, smaller ϵ and δ values result in faster depletion of the privacy budget, restricting the number of feasible iterations. With ($\epsilon = 0.1$, $\delta = 10^{-6}$), the algorithm failed after only 12 iterations. But for ($\epsilon = 2$, $\delta = 10^{-3}$), it sustained over 2900 iterations. The algorithm can help select optimal ϵ, δ configurations to balance privacy and utility for a given workload. Stronger guarantees come at the expense of utility.

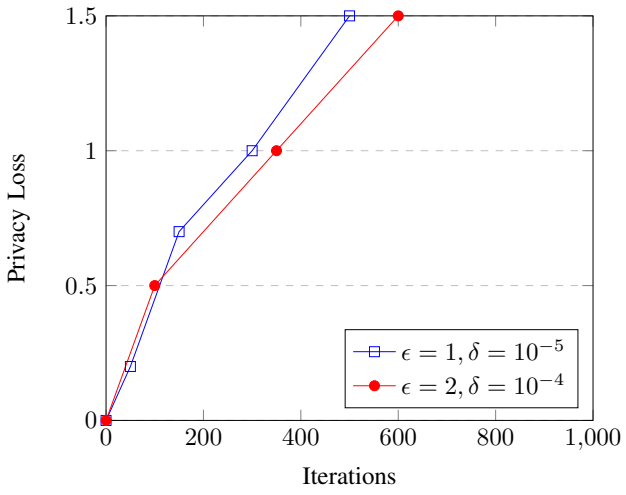

Fig. 3. Privacy Loss vs Iterations

Additionally, Figure 3 illustrates how privacy loss accu-

mulates faster for smaller epsilon and delta values, causing earlier budget depletion, and reinforces the core concept of iterative queries gradually consuming the privacy budget. The results clearly demonstrate the tradeoff between privacy_budgets and feasible utility regarding several iterative computations supported before depletion. By tracking tight moments, accountant bounds, and frequent budget updates, our algorithm provides a rigorous methodology for upholding formal differential privacy guarantees. We further analyzed the composition by varying batch sizes while keeping the overall privacy_budget fixed. Larger batch sizes result in better utility for a given budget by amortizing the privacy cost. Adaptive batching strategies could optimize this privacy-utility curve. The accuracy loss due to noise injection was quantified using relative error metrics between noisy and original histogram outputs. The median error remains under 5% for the initial 800 queries before rising. By tuning noise parameters, acceptable accuracy can be maintained. Our experiments involved simple histogram queries. Future work remains to evaluate performance on more complex statistical, ML, and graph computations.

VI. Summary and Conclusion

Differential privacy offers rigorous protections for statistical queries over sensitive data. However, accurately quantifying cumulative privacy loss during complex iterative analyses remains an open challenge limiting adoption. This paper proposes a heuristic algorithm to evaluate end-to-end differential privacy guarantees by modeling total privacy loss as an accumulative random variable composited over successive operations. By tracking tight moments accountant bounds on this variable, the algorithm can precisely determine whether the aggregate leakage exceeds specified (ϵ,δ) budgets. Empirical validation on simulated query workloads demonstrates the algorithm's ability to verify differential privacy for processes with hundreds of iterative computations. The results provide insights into optimizing the tradeoffs between privacy budgets and feasible utility, enabling formal reasoning about privacy risks across diverse mechanisms throughout the data analysis lifecycle. By addressing key challenges and offering a novel, adaptive solution, it contributes to the theoretical and practical advancement of privacy-preserving technologies, which will become increasingly important as the digital landscape continues to evolve.

References

[1] Chen, J., Henry, E. & Jiang, X. Is Cybersecurity Risk Factor Disclosure Informative? Evidence from Disclosures Following a Data Breach. J Bus Ethics 187, 199–224. https://doi.org/10.1007/s10551-022-05107-z

[2] Liu, Z.; Huang, L.; Xu, H.; Yang, W. Locally Differentially Private Heterogeneous Graph Aggregation with Utility Optimization. Entropy 2023, 25, 130. https://doi.org/10.3390/e25010130

[3] https://www.ibm.com/reports/data-breach

[4] Dwork, C. (2006). Differential Privacy. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds) Automata, Languages and Programming. ICALP 2006. Lecture Notes in Computer Science, vol 4052. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11787006_1

[5] P. Thantharate, "IntelligentMonitor: Empowering DevOps Environments with Advanced Monitoring and Observability," 2023 International Conference on Information Technology (ICIT), Amman, Jordan, 2023, pp. 800-805, doi: 10.1109/ICIT58056.2023.10226123.

[6] P. Thantharate and A. T, "GeneticSecOps: Harnessing Heuristic Genetic Algorithms for Automated Security Testing and Vulnerability Detection in DevSecOps," 2023 6th International Conference on Contemporary Computing and Informatics (IC3I), Gautam Buddha Nagar, India, 2023, pp. 2271-2278, doi: 10.1109/IC3I59117.2023.10398075

[7] A. Triastcyn and B. Faltings, "Federated Learning with Bayesian Differential Privacy," 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 2019, pp. 2587-2596, doi: 10.1109/BigData47090.2019.9005465.

[8] E. Lobo-Vesga, A. Russo and M. Gaboardi, "A Programming Framework for Differential Privacy with Accuracy Concentration Bounds," 2020 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 2020, pp. 411-428, doi: 10.1109/SP40000.2020.00086.

[9] Koskela, Antti, et al. "Individual Privacy Accounting with Gaussian Differential Privacy." arXiv, 30 Sept. 2022, doi:10.48550/arXiv.2209.15596.

[10] A. Bhattacharjee, S. Badsha and S. Sengupta, "Personalized Privacy Preservation for Smart Grid," 2021 IEEE International Smart Cities Conference (ISC2), Manchester, United Kingdom, 2021, pp. 1-7, doi: 10.1109/ISC253183.2021.9562929.

[11] M. E. Gursoy, A. Tamersoy, S. Truex, W. Wei and L. Liu, "Secure and Utility-Aware Data Collection with Condensed Local Differential Privacy," in IEEE Transactions on Dependable and Secure Computing, vol. 18, no. 5, pp. 2365-2378, 1 Sept.-Oct. 2021, doi: 10.1109/TDSC.2019.2949041.

[12] Q. Wang, Z. Li, Q. Zou, L. Zhao and S. Wang, "Deep Domain Adaptation With Differential Privacy," in IEEE Transactions on Information Forensics and Security, vol. 15, pp. 3093-3106, 2020, doi: 10.1109/TIFS.2020.2983254.

[13] Jagielski, Matthew, et al. "Auditing Differentially Private Machine Learning: How Private is Private SGD?" Advances in Neural Information Processing Systems, vol. 33, 2020, pp. 22205-16, proceedings.

[14] P. Thantharate and A. T, "CYBRIA - Pioneering Federated Learning for Privacy-Aware Cybersecurity with Brilliance," 2023 IEEE 20th International Conference on Smart Communities: Improving Quality of Life using AI, Robotics and IoT (HONET), Boca Raton, FL, USA, 2023, pp. 56-61, doi: 10.1109/HONET59747.2023.10374608.

[15] H. Yan, X. Li, H. Li, J. Li, W. Sun and F. Li, "Monitoring-Based Differential Privacy Mechanism Against Query Flooding-Based Model Extraction Attack," in IEEE Transactions on Dependable and Secure Computing, vol. 19, no. 4, pp. 2680-2694, 1 July-Aug. 2022, doi: 10.1109/TDSC.2021.3069258.

[16] X. Zheng, L. Zhang, K. Li and X. Zeng, "Efficient publication of distributed and overlapping graph data under differential privacy," in Tsinghua Science and Technology, vol. 27, no. 2, pp. 235-243, April 2022, doi: 10.26599/TST.2021.9010018.

[17] Thantharate, P.; Thantharate, A. ZeroTrustBlock: Enhancing Security, Privacy, and Interoperability of Sensitive Data through ZeroTrust Permissioned Blockchain. Big Data Cogn. Comput. 2023, 7, 165. https://doi.org/10.3390/bdcc7040165

[18] Utaliyeva, A.; Shin, J.; Choi, Y.-H. Task-Specific Adaptive Differential Privacy Method for Structured Data. Sensors 2023, 23, 1980. https://doi.org/10.3390/s23041980

[19] L. Fan and L. Xiong, "An Adaptive Approach to Real-Time Aggregate Monitoring With Differential Privacy," in IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 9, pp. 2094-2106, Sept. 2014, doi: 10.1109/TKDE.2013.96

[20] https://github.com/ptdevsecops/PrivML