Federated Learning with Privacy-Preserving Optimization for Distributed Intelligent Systems

Rahul L. Bose
Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS, USA.
rahulbose@ku.edu

Abstract

The proliferation of distributed intelligent systems, ranging from edge-based Internet of Things networks to autonomous mobile platforms, has created an urgent need for machine learning paradigms that respect data locality and user privacy. Federated learning has emerged as a foundational framework that enables collaborative model training across decentralized devices without transferring raw data to a central server. However, the practical deployment of federated learning at scale reveals a complex landscape of technical and socio-technical challenges, particularly regarding the optimization of privacy-preserving mechanisms. This paper presents a comprehensive system-level analysis of federated learning architectures that incorporate privacy-preserving optimization techniques, including differential privacy, secure multi-party computation, and homomorphic encryption. The analysis emphasizes structural trade-offs among communication efficiency, computational overhead, model accuracy, and privacy guarantees. It further examines the governance and policy implications of deploying such systems in critical domains such as healthcare, finance, and smart infrastructure. Robustness against adversarial threats, fairness across heterogeneous client populations, and long-term sustainability are evaluated from an interdisciplinary perspective. Case illustrations from cross-domain deployments highlight the necessity of context-aware privacy budgets and adaptive optimization schedules. The paper concludes with forward-looking recommendations for designing privacy-preserving federated systems that balance technical performance with ethical and regulatory compliance, thereby supporting the responsible evolution of distributed artificial intelligence.

Keywords

federated learning, privacy-preserving optimization, differential privacy, secure aggregation, distributed systems, adversarial robustness, fairness, governance, infrastructure sustainability, socio-technical systems.

1. Introduction

The shift toward decentralized data processing in distributed intelligent systems has been driven by two converging forces: the exponential growth of data generated at the network edge and the increasing stringency of data protection regulations such as the General Data Protection Regulation in Europe and the California Consumer Privacy Act in the United States. Centralized machine learning, which aggregates all training data into a single repository, becomes infeasible or illegal under such constraints because it violates the fundamental principle of data minimization. Federated learning addresses this dilemma by moving the computation to the data rather than the data to the computation, thereby preserving the raw data at its source while enabling model updates to be aggregated across many devices [1]. Since its formalization in 2017, federated learning has been applied to a

wide array of applications, from next-word prediction on smartphones to diagnostic imaging in healthcare consortia [2]. Yet the very architecture that ensures data locality introduces new vulnerabilities, particularly concerning the privacy of the model updates themselves. Gradient information, even when transmitted without raw data, can leak sensitive attributes about local datasets through model inversion or membership inference attacks [3]. Consequently, privacy-preserving optimization has become a central research thrust within the federated learning community, aiming to protect not only the raw data but also the intermediate computation results exchanged during training.

This paper adopts a system-oriented perspective to analyze how privacy-preserving techniques can be integrated into federated learning pipelines without compromising the distributed system's operational efficiency, robustness, or fairness. The discussion moves beyond algorithmic details to examine the structural trade-offs inherent in scaling such systems across heterogeneous devices, network topologies, and regulatory regimes. The motivation for this systemic view stems from the observation that many proposed privacy-preserving mechanisms, while theoretically sound, impose prohibitive communication or computation burdens that degrade real-world performance. Identifying the regimes in which each technique is most advantageous requires a holistic understanding of the underlying infrastructure, including bandwidth constraints, client hardware capabilities, and the dynamic nature of device availability [4]. Furthermore, the societal implications of deploying privacy-preserving federated learning extend to issues of algorithmic fairness, as differential privacy budgets can disproportionately affect minority subgroups, and to questions of governance, as the distribution of trust across multiple parties complicates accountability frameworks [5]. By examining these interconnected dimensions, this paper aims to provide a reference for researchers and practitioners who seek to design sustainable, privacy-respecting distributed intelligent systems.

2. System Architecture and Communication Infrastructure

The foundational architecture of a federated learning system consists of a coordinating server and a population of clients, each possessing a local dataset that is not shared with other clients or with the server. Training proceeds in iterative rounds: the server broadcasts the current global model parameters to a subset of clients, each client performs local stochastic gradient descent on its own data, and the resulting model updates are sent back to the server for aggregation, typically by weighted averaging [1]. This client-server model can be realized in various topologies, including star networks, hierarchical networks with intermediate aggregators, and fully decentralized peer-to-peer structures in which no central server exists [6]. The choice of topology has profound implications for privacy, security, and communication overhead. Star topologies simplify coordination but introduce a single point of failure and a potential privacy bottleneck if the server is compromised. Hierarchical architectures distribute the aggregation load across regional aggregators, reducing communication distances and latency, but they require careful management of trust among the intermediate nodes [7]. Fully decentralized topologies eliminate the central server entirely, relying on gossip protocols for model diffusion; however, they face challenges in ensuring convergence and synchronizing updates in the presence of Byzantine nodes [8].

Communication infrastructure remains a critical bottleneck in most federated deployments, particularly when clients are mobile or resource-constrained. The frequent exchange of model updates over wireless links can consume substantial bandwidth and battery power, motivating the development of compression techniques, quantization schemes, and adaptive sampling

strategies [9]. Privacy-preserving mechanisms compound these demands because they typically encrypt or perturb the communicated values. For example, secure aggregation protocols using additive secret sharing require all participating clients to exchange pairwise masks before sending their masked updates to the server, which increases both the number of messages and the cryptographic operations per round [3]. Homomorphic encryption, which allows computation on ciphertexts, incurs even higher computational costs and larger message sizes, making it practical only for small models or low-latency-tolerant applications [10]. The trade-off between privacy guarantees and communication efficiency must be navigated through careful system design, such as batching aggregations over multiple rounds or selecting subsets of clients with sufficient computational resources to handle encryption operations.

3. Privacy-Preserving Optimization Methods

Three principal categories of privacy-preserving optimization have been integrated into federated learning frameworks: differential privacy, secure multi-party computation, and homomorphic encryption. Differential privacy provides a formal guarantee that the inclusion or exclusion of any single data point does not significantly alter the output of the algorithm. In the federated context, differential privacy is typically applied at the client level by clipping and adding noise to the model updates before they are aggregated, thereby obscuring the contribution of any individual device [2]. The magnitude of the noise is calibrated to a privacy budget epsilon, which controls the trade-off between privacy and accuracy. A lower epsilon yields stronger privacy but higher variance in the aggregated model, potentially slowing convergence and degrading final model quality. The choice of epsilon is further complicated by the need to compose privacy guarantees across multiple training rounds, as the cumulative privacy loss grows with the number of iterations. Advanced composition theorems and privacy amplification via subsampling can mitigate this growth, but they introduce additional analytical complexity [11]. Moreover, the effectiveness of differential privacy in a federated setting depends on the distribution of data across clients, because non-IID data can amplify the impact of noise on rare or skewed classes, leading to fairness concerns [5].

Secure multi-party computation (SMPC) encompasses a family of cryptographic protocols that allow multiple parties to jointly compute a function over their private inputs without revealing those inputs to each other. In federated learning, the most widely adopted SMPC technique is secure aggregation, which computes the sum of the clients' updates without the server ever seeing individual updates [3]. Secure aggregation protocols typically use secret sharing: each client splits its update into shares distributed among other clients, and the server reconstructs only the aggregated sum from the combined shares. This approach prevents the server from inferring the update of any single client, thereby defending against gradient leakage attacks. However, secure aggregation requires that all participating clients remain online during the computation to reveal their shares, a condition that is often violated in real-world mobile networks due to dropped connections or battery failures. Dropout handling mechanisms, such as threshold secret sharing and redundant shares, introduce additional communication rounds and complexity [12]. Furthermore, secure aggregation alone does not protect against inference attacks that leverage the aggregated model itself over many rounds, which is why it is frequently combined with differential privacy in a hybrid framework.

Homomorphic encryption enables computations to be performed directly on encrypted data, thereby allowing the server to aggregate encrypted updates without ever decrypting them. Fully homomorphic encryption supports arbitrary computations but remains computationally

prohibitive for large models. Partially homomorphic encryption schemes, such as those supporting only addition, are more efficient and are therefore more commonly used in federated aggregation [10]. Despite their lower overhead, partially homomorphic protocols still require substantial cryptographic operations at the client side, lengthening the time per training round and increasing energy consumption on battery-powered devices. The choice among these three approaches—or their combination—depends on the threat model assumed. Differential privacy defends against a strong adversary who may have auxiliary information about the training set, while secure aggregation and homomorphic encryption protect against an honest-but-curious server that tries to learn individual updates. A comprehensive privacy-preserving system often layers these techniques to achieve defense in depth, but the cumulative resource cost must be justified by the sensitivity of the data and the regulatory environment [13].

4. Structural Trade-Offs: Accuracy, Efficiency, and Privacy

The integration of privacy-preserving optimization into federated learning introduces a multi-objective trade-off space in which improving one dimension often degrades others. The most studied trade-off is between privacy and model accuracy. Adding noise or cryptographic overhead can increase the variance of the aggregated model, resulting in slower convergence and potentially a lower final test accuracy. Empirical studies have shown that for a given privacy budget epsilon, the accuracy degradation is more pronounced when the local data distributions are highly non-IID, because the noise disproportionately affects clients with few samples or atypical feature distributions [4]. This phenomenon raises questions about the fairness of privacy guarantees across clients, as those with smaller datasets may bear a higher accuracy cost for the same nominal privacy protection. To address this, adaptive privacy budgeting schemes have been proposed that allocate a larger noise tolerance to clients with more data or that dynamically adjust the clipping threshold based on the observed gradient norms [11]. Yet such adaptations introduce additional parameters that must be tuned per deployment, complicating the system's governance.

Communication efficiency is another axis of trade-off. Both secure aggregation and homomorphic encryption increase the size of messages exchanged and the number of communication rounds required. For example, in a secure aggregation protocol with secret sharing, each client must communicate with multiple peers before the final aggregation, leading to a quadratic increase in the number of messages relative to the number of clients in the round [3]. Compression techniques such as gradient quantization and sparsification can reduce message sizes, but they interact poorly with cryptographic primitives because the compressed values may not be compatible with the arithmetic operations required for encryption [9]. Similarly, differential privacy adds noise that reduces the signal-to-noise ratio, making compression more difficult because the noise component cannot be easily separated from the true gradient. System designers must therefore choose an operating point that respects the available bandwidth, latency constraints, and privacy requirements simultaneously. Cross-layer optimization that considers the entire protocol stack—from the network layer to the cryptographic primitives—is essential for achieving a feasible deployment.

A further trade-off involves the computational load on clients. Many federated devices, such as smartphones, sensors, and edge gateways, have limited processing power and battery life. Homomorphic encryption can increase the per-round computation time by orders of magnitude, rendering it unsuitable for resource-constrained clients unless the model size is

very small [10]. Differential privacy is computationally lighter because it only requires clipping and adding noise, but the clipping operation itself necessitates a pass over the local gradients, which may be costly for large models. Secure aggregation falls in between; its computational cost is dominated by the generation and reconstruction of secret shares, which scales linearly with the number of clients and the model size. To balance these demands, many production systems use a combination of local differential privacy on the client side and secure aggregation at the server, with the noise level calibrated to the expected number of clients in each round [8]. This hybrid approach provides a practical trade-off, but it still requires careful management of client churn and network asynchrony.

5. Robustness Against Adversarial Threats

Privacy-preserving optimization in distributed systems must be resilient not only to honest-but-curious adversaries but also to malicious participants who may attempt to corrupt the training process. Byzantine attacks, in which a compromised client sends arbitrary or carefully crafted updates, can cause the aggregated model to converge to a suboptimal solution or to include a backdoor that responds to specific triggers [14]. Byzantine-resilient aggregation rules, such as trimmed mean, median, and Krum, have been developed to detect and discard outlier updates, but they often rely on the assumption that the benign updates are statistically similar. When combined with differential privacy, the noise injected to protect privacy can mask the anomaly of a Byzantine attack, making detection harder [15]. Conversely, some Byzantine defenses require comparing individual updates, which contradicts the privacy goal of secure aggregation because the server would need to see per-client values. This tension between security and privacy is a central challenge in designing robust federated systems. One approach is to use secure aggregation in combination with verifiable computation, such as zero-knowledge proofs, so that the server can check that each client's update was computed correctly without revealing it [3]. However, the overhead of such proofs is currently too high for practical deployment on low-power devices.

Another adversarial threat is model poisoning, where a client contributes data that biases the global model toward a malicious objective. Unlike Byzantine attacks that aim to disrupt convergence, model poisoning can be subtle and persistent, gradually shifting the decision boundary. Privacy-preserving mechanisms that mask individual updates prevent the server from inspecting each contribution, thereby increasing the difficulty of detecting poisoning [14]. To mitigate this, the system can rely on server-side validation on a small public dataset, or employ reputation systems that penalize clients whose aggregated updates deviate from the expected pattern over rounds. Yet these solutions introduce their own privacy risks by potentially exposing client behavior patterns. The trade-off between privacy and robustness necessitates a careful risk assessment for each application domain. In high-stakes domains such as medical diagnosis, the cost of a successful poisoning attack may be life-threatening, justifying a reduction in privacy guarantee to allow for stronger anomaly detection. In other contexts, such as consumer analytics, privacy may take precedence over the low probability of a coordinated attack.

6. Fairness and Governance

The deployment of federated learning with privacy-preserving optimization raises important questions about algorithmic fairness, especially when the client population is heterogeneous in terms of data volume, quality, and representativeness. Differential privacy, by adding the same level of noise to all clients, may disadvantage clients with smaller or less diverse datasets because their contributions are more easily dominated by noise [5]. This can lead to a

global model that performs poorly on minority subgroups, exacerbating existing disparities in the system's performance. Fairness-aware privacy budgeting, where the noise scale is adjusted based on group membership, has been proposed but faces the challenge of defining groups without accessing sensitive attributes [16]. Furthermore, the choice of aggregation algorithm itself can introduce bias. For instance, federated averaging weights each client's update by the number of local samples, which gives more influence to clients with more data. If those clients are not representative of the overall population, the model will skew toward their distribution. Privacy-preserving mechanisms do not inherently correct this skew; they only obscure individual contributions. Governance frameworks must therefore specify how to define fairness metrics, how to audit the model after deployment, and how to redress inequities that emerge.

Governance also encompasses the allocation of trust among the parties involved in the federated system. In a typical setting, the server is assumed to be honest-but-curious, meaning it follows the protocol but may attempt to infer private information from the aggregated updates. Secure aggregation and homomorphic encryption shift the trust assumption by ensuring that even the server cannot learn individual updates [3]. However, these cryptographic guarantees rely on the correct implementation of the protocol and on the integrity of the underlying infrastructure. A malicious server could collude with some clients to bypass privacy protections, or could manipulate the aggregation result to favor a particular outcome. Auditable protocols that generate proofs of correct execution, such as verifiable secure aggregation, are an active area of research but are not yet mature enough for widespread adoption [12]. From a policy perspective, regulatory frameworks such as the GDPR require data controllers to implement appropriate technical and organizational measures. Federated learning with strong privacy guarantees can help meet these requirements, but the system deployer must also document the threat model, the privacy budget used, and the measures taken to ensure accountability. Transparency reports and third-party audits become crucial for building trust among end users and regulators.

7. Deployment and Sustainability

Scaling privacy-preserving federated learning from laboratory experiments to production environments involves overcoming numerous practical hurdles related to client availability, network heterogeneity, and energy consumption. In mobile deployments, clients frequently drop out of training rounds due to connectivity loss, battery depletion, or user activity. Secure aggregation protocols that require all clients to be online simultaneously are therefore impractical; threshold protocols that tolerate a certain fraction of dropouts are necessary, but they add complexity and reduce the effective privacy guarantee because the server learns that some updates are missing [3]. Adaptive client selection strategies that prioritize clients with stable connections and sufficient resources can improve convergence, but they may introduce bias if certain regions or device types are systematically excluded [4]. Sustainability also concerns the energy footprint of cryptographic operations. On-device encryption and decryption of large model parameters can drain batteries quickly, shortening the useful life of the device and increasing electronic waste. Trade-offs between privacy strength and energy consumption must be quantified during system design, and users should be informed about the impact of participating in federated training [13].

Another sustainability dimension is the long-term evolution of the model. As data distributions shift over time—a phenomenon known as concept drift—the global model must be continuously updated. Privacy-preserving mechanisms that rely on a fixed privacy budget

may exhaust that budget over many rounds, forcing the system to stop training or to relax privacy guarantees [11]. Differential privacy with a finite cumulative budget requires the system designer to plan for the model's lifecycle, deciding how many rounds can be executed before the privacy loss exceeds acceptable thresholds. In practice, many systems use a sliding window approach, where older updates are forgotten and the privacy budget is renewed periodically, but this introduces additional architectural complexity. Cross-domain comparisons reveal that healthcare federated networks, which often have stricter privacy regulations and longer operational lifespans, tend to adopt more conservative privacy budgets and invest in high-efficiency cryptographic accelerators [19]. In contrast, consumer-facing applications may prioritize fast iteration and accept higher privacy risks, provided the system is transparent about its practices.

8. Case Illustrations and Cross-Domain Comparisons

To ground the discussion of system-level trade-offs, it is instructive to examine how privacy-preserving federated learning has been deployed in different domains. In the healthcare sector, multi-institutional collaborations for medical imaging diagnosis have adopted federated learning with differential privacy to comply with patient data protection laws. A landmark study involving chest X-ray classification across multiple hospitals demonstrated that with a moderate privacy budget, the model achieved accuracy within a few percentage points of the centralized baseline, although the training required more rounds due to the added noise [19]. The infrastructure in healthcare settings often includes high-performance servers and dedicated networks, reducing the communication bottleneck, but the need for regulatory approval slowed the adoption of secure aggregation because of the difficulty in auditing the protocol. In the financial domain, fraud detection systems that operate across banks use federated learning with a combination of secure aggregation and differential privacy to protect transaction histories. Here, the non-IID nature of the data—each bank's fraud patterns are distinct—poses a challenge for convergence, and the use of differential privacy can amplify accuracy loss on rare fraudulent transactions [5]. Banks have responded by using larger clipping thresholds and more frequent rounds, which increases communication load but maintains acceptable model performance.

In the domain of smart infrastructure, such as traffic management systems that learn from distributed sensors, the devices are often resource-constrained and operate on battery power. Homomorphic encryption is generally ruled out due to computational demands, and secure aggregation is used only with small models and short training sessions. Researchers have demonstrated that lightweight differential privacy combined with gradient compression can achieve reasonable privacy guarantees while keeping energy consumption low [9]. The key trade-off in this domain is between privacy and real-time responsiveness—traffic predictions must be updated quickly to be useful, so the system often sacrifices a portion of the privacy budget to reduce the number of aggregation rounds. These cross-domain comparisons highlight that there is no one-size-fits-all solution; the optimal configuration depends on the regulatory environment, the available infrastructure, the cost of privacy breaches, and the tolerance for accuracy degradation. Future systems should be designed with modular privacy and optimization components that can be tuned to the specific context.

9. Future Directions and Policy Implications

Looking ahead, several research directions promise to improve the feasibility of privacy-preserving federated learning in distributed intelligent systems. One promising avenue is the development of adaptive privacy mechanisms that adjust the noise level based on real-time

measurements of model sensitivity and client heterogeneity. Such mechanisms could reduce the privacy-accuracy trade-off by allocating more noise only when necessary, analogous to adaptive clipping in differential privacy [11]. Another direction involves the integration of local and global privacy guarantees through hierarchical noise calibration, where clients can enforce their own privacy requirements while still contributing to the global model. This approach would align with the principle of data sovereignty, allowing each data owner to define its own privacy budget. Furthermore, advances in hardware security modules and trusted execution environments can offload cryptographic operations from the main processor, reducing both energy consumption and latency [13]. However, hardware-based solutions introduce supply chain trust issues and may not be accessible in all deployment contexts.

From a policy perspective, the deployment of privacy-preserving federated learning at scale necessitates clear guidelines about privacy budget accounting, transparency reporting, and auditability. Regulators will need to define acceptable thresholds for cumulative privacy loss in long-running systems, and certif ication frameworks for secure aggregation protocols should be established to prevent the misuse of cryptographic guarantees [5]. The interdisciplinary nature of these challenges calls for collaboration between computer scientists, legal scholars, and ethicists. As distributed intelligent systems become embedded in critical infrastructure, the responsibility to design systems that are both private and robust falls on the entire engineering community. Federated learning with privacy-preserving optimization is not merely a technical upgrade; it represents a shift toward a more decentralized and user-centered paradigm of machine learning, one that respects the fundamental rights of individuals while enabling collective intelligence.

10. Conclusion

This paper has presented a comprehensive system-level analysis of federated learning with privacy-preserving optimization for distributed intelligent systems. The discussion has highlighted the structural trade-offs among privacy, accuracy, communication efficiency, and computational cost that must be navigated when deploying such systems in real-world environments. Privacy-preserving techniques, including differential privacy, secure multi-party computation, and homomorphic encryption, each offer distinct protection guarantees but impose differing burdens on the infrastructure. The integration of these techniques requires careful consideration of client heterogeneity, adversarial threats, fairness, and long-term sustainability. Case illustrations from healthcare, finance, and smart infrastructure demonstrate that optimal configurations are highly context-dependent. Governance frameworks must evolve to provide accountability and transparency while respecting the privacy rights of data owners. As federated learning moves from research prototypes to production systems, the community must prioritize interdisciplinary collaboration to ensure that distributed intelligent systems are not only efficient and accurate but also private, fair, and trustworthy. The continued development of adaptive privacy mechanisms and hardware-assisted solutions will be instrumental in realizing this vision.

References

1. McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS).

2. Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016). Deep learning with differential privacy. Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security.

3. Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., ... & Seth, K. (2017). Practical secure aggregation for privacy-preserving machine learning. Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security.

4. Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., ... & Zhao, S. (2021). Advances and open problems in federated learning. Foundations and Trends in Machine Learning, 14(1-2), 1-210.

5. Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated learning: Challenges, methods, and future directions. IEEE Signal Processing Magazine, 37(3), 50-60.

6. Geyer, R. C., Klein, T., & Nabi, M. (2017). Differentially private federated learning: A client level perspective. arXiv preprint arXiv:1712.07557.

7. Shokri, R., & Shmatikov, V. (2015). Privacy-preserving deep learning. Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security.

8. Hard, A., Rao, K., Mathews, R., Ramaswamy, S., Beaufays, F., Augenstein, S., ... & Ramage, D. (2018). Federated learning for mobile keyboard prediction. arXiv preprint arXiv:1811.03604.

9. Smith, V., Chiang, C. K., Sanjabi, M., & Talwalkar, A. (2017). Federated multi-task learning. Advances in Neural Information Processing Systems, 30.

10. Li, X., Huang, K., Yang, W., Wang, S., & Zhang, Z. (2020). On the convergence of FedAvg on non-IID data. Proceedings of the International Conference on Machine Learning.

11. Sahu, A. K., Li, T., Sanjabi, M., Zaheer, M., Talwalkar, A., & Smith, V. (2018). On the convergence of federated optimization in heterogeneous networks. arXiv preprint arXiv:1812.06127.

12. Mohri, M., Sivek, G., & Suresh, A. T. (2019). Agnostic federated learning. Proceedings of the 36th International Conference on Machine Learning.

13. Wang, J., Charles, Z., Xu, Z., Joshi, G., McMahan, H. B., & others (2019). A field guide to federated optimization. arXiv preprint arXiv:1907.10995.

14. Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., & Shmatikov, V. (2020). How to backdoor federated learning. Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics.

15. Blanchard, P., El Mhamdi, E. M., Guerraoui, R., & Stainer, J. (2017). Machine learning with adversaries: Byzantine tolerant gradient descent. Advances in Neural Information Processing Systems, 30.

16. Zafar, M. B., Valera, I., Gomez Rodriguez, M., & Gummadi, K. P. (2017). Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. Proceedings of the 26th International Conference on World Wide Web.

17. Dwork, C., McSherry, F., Nissim, K., & Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. Theory of Cryptography Conference.

18. Caldas, S., Wu, J., Li, T., Konečný, J., McMahan, H. B., Smith, V., & Talwalkar, A. (2018). LEAF: A benchmark for federated settings. arXiv preprint arXiv:1812.01097.

19. Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., ... & Cardoso, M. J. (2020). The future of digital health with federated learning. NPJ Digital Medicine, 3(1), 119.

20. Liu, Y., Kang, Y., Xing, C., Chen, T., & Yang, Q. (2020). A secure federated transfer learning framework. IEEE Intelligent Systems, 35(4), 70-82.