Theory and Practice of Science and Technology
2025, VOL. 6, NO. 11, 49-54
DOI: 10.47297/taposatWSP2633-456910.20250611

A Comparative Review of Privacy Protection Techniques in Federated Learning

Jiayi Li

School of Data Science and Artificial Intelligence, Wenzhou Institute of Technology, Wenzhou, Zhejiang, 325000, China

ABSTRACT
This study addresses privacy protection challenges in federated learning by establishing a six-dimensional comparative framework encompassing "dataset type, accuracy loss, computational overhead, communication cost, privacy strength, and application domain." It systematically reviews representative works on differential privacy (DP), homomorphic encryption (HE), secure multi-party computation (SMPC), and their engineering variants MPHE-1, MHESA, and QSFL. Establishing a unified evaluation metric, we conclude: DP incurs minimal communication overhead but sacrifices accuracy; HE achieves near-cleartext accuracy with substantial computational overhead; SMPC offers strong security but is constrained by interaction rounds and network conditions; the three variants respectively address low-round scenarios, dropout scenarios, and edge efficiency scenarios. This paper provides contextualized selection recommendations and a reproducibility checklist, while proposing future research directions for large models and real-world network environments. It offers methodological and engineering references for compliant and implementable privacy-preserving federated learning.

KEYWORDS
Federated learning; Differential privacy; Homomorphic encryption; Secure multi-party computation; Secure aggregation; Quantization; System trade-offs

1 Introduction

Although federated learning mitigates compliance and leakage risks associated with centralized training, achieving a balance among privacy strength, model utility, and system cost remains a core challenge for researchers in real-world networks and large-scale collaborative environments. Existing studies often focus on single technical approaches, with inconsistent reporting dimensions and threat model descriptions, making cross-paper comparison and reproducibility difficult. simultaneously, systematic evidence for complex scenarios like healthcare, finance, and IoT remains insufficient, lacking unified benchmarks and auditable evidence tailored to large models and real-world network conditions. Therefore, this paper establishes a six-dimensional comparative framework based on a systematic literature review: without re-normalizing original hardware metrics, it uniformly presents utility loss as percentage points (pp), training overhead as multiples, and communication volume as MB/round. For unreported metrics, it adheres to the principle of "no fabrication of data." . Within this framework, we synthesize representative studies on DP, HE, SMPC, and their engineered variants (MPHE-1, MHESA, QSFL) to distill actionable insights from the "triadic trade-offs," propose scenario-based selection strategies, and provide comparative tables to enhance comparability and reproducibility. Finally, we outline future directions including unified benchmarks for large models and real-world networks, adaptive hybrid orchestration, auditable privacy evidence, and system automation.

2 Literature Review

2.1 Differential Privacy (DP)

Differential Privacy (DP) was first proposed by Cynthia Dwork in 2006 [1]. Its fundamental principle is to protect individual privacy by obscuring the influence of individual samples on the overall results while ensuring the accuracy of the analysis. Formally defined: A random algorithm M satisfies (ε,δ) if, for any two datasets D and D′ differing by only one sample, and any measurable set $S \subseteq \text{Range}(M)$, the following holds:

$$\Pr[M(D) \in S] \leq e^{\varepsilon} \cdot \Pr[M(D') \in S] + \delta \quad (1)$$

where ε quantifies privacy loss and δ denotes the tolerable probability of privacy leakage [1].

DP research has progressed through stages including theoretical foundations, mechanism optimization and application expansion, and paradigm evolution. In mechanism optimization, techniques like exponential mechanisms and sparse vector methods enhance data utility within a given ε threshold [1]. Application expansion saw Google's DP-SGD drive its adoption in deep learning. Paradigms transitioned from centralized DP to localized DP (LDP) [2]. DP's strengths lie in provable privacy guarantees and intuitive parameter design, while its limitations include utility degradation due to noise, complex budget management, and lower utility in LDP.

2.2 Homomorphic Encryption (HE)

Homomorphic Encryption (HE) is a cryptographic method enabling direct operations on ciphertext that yield results identical to plaintext computations. Based on supported operation types and frequencies, it is categorized into partial homomorphic encryption (PHE), semi-homomorphic encryption (SHE), and fully homomorphic encryption (FHE). The CKKS scheme proposed by Cheon et al. in 2017 achieves efficient addition and multiplication of encrypted real and complex numbers through approximate coding and numerical scaling, significantly enhancing HE's practicality in privacy-preserving numerical computation [3].

Building upon this foundation, subsequent research has focused on enhancing HE performance and applicability. This includes optimizing ciphertext structures, improving noise management, reducing computational and storage overhead, and exploring integration with technologies such as machine learning and cloud computing [4]. These advancements have amplified HE's potential in fields like medical data analysis, financial computing, and secure multi-party collaboration. Its key advantage lies in supporting complex computations while preserving data privacy, offering flexible trade-offs between security, performance, and accuracy. Nevertheless, HE still faces challenges such as high computational and storage overhead, complex parameter selection, and limited support for high-precision computations [3-4].

2.3 Secure Multi-Party Computation (SMPC)

Secure Multi-Party Computation (SMPC) is a critical research area in modern cryptography, aiming to enable multiple parties to jointly compute the result of a target function without revealing their respective inputs. This concept was first proposed in 1982, with the formalization of the secure two-party computation model in the paper *Protocols for Secure Computations*, laying the theoretical foundation for privacy-preserving computation [5].

Subsequently, the GMW protocol proposed by Goldreich, Micali, and Wigderson achieved secure computation of arbitrary functions under a semi-honest model [6]. Ben-Or, Goldwasser, and Wigderson's BGW protocol extended the applicability of multi-party computation, guaranteeing correctness and privacy even under a malicious adversary model [7]. While these studies demonstrated the feasibility of secure computation, their high communication and computational complexity limited early practical applications.

In recent years, researchers have significantly enhanced the efficiency and scalability of SMPC by integrating techniques such as secret sharing, homomorphic encryption, and hybrid computation frameworks. Nevertheless, existing solutions still suffer from high communication overhead and deployment complexity. Future research should focus on protocol optimization and cross-domain applications to further advance the implementation of SMPC in scenarios like federated learning, private data analysis, and blockchain computation.

2.4 Summary of Limitations and Motivation for Variants

In summary, current privacy-preserving techniques in federated learning primarily include differential privacy, homomorphic encryption, and secure multi-party computation. While these traditional methods have achieved significant theoretical and practical results, they still exhibit shortcomings in real-world applications: differential privacy inevitably reduces model accuracy while protecting privacy; single homomorphic encryption methods face excessive computational and communication overhead in multi-party scenarios and lack flexible distributed key management; while secure multi-party computation (SMP) suffers from high resource consumption and complex protocol implementation. To overcome these limitations, researchers have begun exploring fusion schemes combining multiple privacy-preserving techniques. For instance, regarding the three technologies mentioned in this paper, one study proposes a method seamlessly integrating multi-party homomorphic encryption with model aggregation to enhance both aggregation efficiency and security. For comparison purposes, we refer to this approach as MPHE-1 [8] in this paper. Additionally, MHESA [9] proposes an efficient and secure aggregation mechanism based on multi-homogeneous properties, which reduces communication costs while strengthening privacy protection. Furthermore, QSFL, a model training framework integrating integer quantization and secure multi-party computation, has been introduced to enhance computational efficiency and privacy in deep learning [10]. These variant techniques offer new approaches to address bottlenecks in existing methods and improve privacy protection and scalability.

3 Method

This study employs a systematic literature review methodology, systematically screening and comparing relevant literature according to PRISMA standards. The research focuses on privacy protection mechanisms in federated learning (FL), encompassing differential privacy (DP), homomorphic encryption (HE), secure multi-party computation (SMPC), and their variants. Qualitative and quantitative evaluations across six dimensions analyze the evolution and trade-offs of

privacy protection mechanisms.

3.1 Research Design and Comparison Dimensions

Building upon the original paper, this study constructs a unified comparison framework. It selects research directly related to federated learning training or aggregation that provides verifiable experimental data, clear metrics, and well-defined threat models. Key numerical values extracted from data are standardized in units and scope without renormalizing original hardware or implementations. When a method spans multiple datasets, the most representative experiments from the paper are prioritized. Missing entries are not fabricated but labeled as "Not reported (original paper) "; intervals or ranges are presented as stated in the original text.

Comparison dimensions are defined as follows: (1) Dataset Type: Public datasets or tasks used in the paper (e.g., MNIST, CIFAR-10, medical data). (2) Precision-loss (pp): Accuracy of plaintext − Accuracy of method. If only "comparable to plaintext" is stated without quantification, label as "Equivalent to plaintext (unquantified difference)". (3) Computational Overhead: Multiplier relative to plaintext. If only stage durations or ranges are provided, report as originally stated. (4) Communication Cost: Recorded in MB/round or relative multiples. If only protocol-level elements are available, provide textual metrics as originally stated. (5) Privacy Strength: Assign qualitative levels (High/Medium/Low) based on threat models and formal guarantees; details are elaborated in the respective method subsections. (6) Application Domains: Targeted at real-world business scenarios (e. g., medical imaging, financial risk control, encrypted queries), not broadly categorized as "federated learning."

3.2 Traditional Methods

3.2.1 Differential Privacy (DP)

Differential privacy in federated learning prunes client gradients or parameter updates while injecting noise, constraining the impact of individual samples on the output distribution within (ε, δ) to achieve measurable privacy guarantees [1]. Experiments commonly use DP-SGD and moments accountant to measure privacy loss. Under the unified MNIST benchmark, with $(\varepsilon, \delta) = (8, 10\text{^}-5)$, test accuracy is approximately 97%, representing a 1.3 percentage point loss compared to plaintext accuracy of 98.3%. When ε is further reduced, the losses increase to approximately 3.3 and 8.3 percentage points, respectively [11]. DP requires no additional encrypted transmission, with near-zero communication overhead, and effectively mitigates inference attacks like membership inference. Its limitations include sensitivity to batch size, learning rate, and clipping threshold—particularly in non-iID and small-sample client environments where convergence slowdown and accuracy degradation may occur, necessitating combined dynamic clipping and optimizer tuning.

3.2.2 Homomorphic Encryption (HE)

Homomorphic encryption enables direct addition and multiplication within the ciphertext domain: clients encrypt updates for upload, and servers perform aggregation in ciphertext, thereby avoiding plaintext exposure. To reduce computational cost, selective encryption strategies that encrypt only key parameter subsets can be employed [12]. Empirical tests on MNIST and CIFAR-10 show that full-encryption training slows down by approximately 3.1x and 17.1x compared to plaintext training, respectively. Communication volume decreased from 871.9MB to 209.83MB when encrypting only 20% of parameters, equivalent to a reduction of approximately 4.15 times. Encrypting just 5% of parameters suppressed the accuracy of membership inference attacks to near random levels (around 50%), while maintaining accuracy within 0.1% of plaintext results [12]. The underlying approach typically employs approximate numerical schemes like CKKS to support vectorized homomorphic operations for floating-point neural networks. This path offers end-to-end secretness and good utility preservation. Its drawbacks include significant computational and bandwidth overhead for encryption/decryption and homomorphic operations, particularly sensitive to large models and high-dimensional gradients, necessitating engineering trade-offs between encryption ratio, block-wise vectorization, and parameter sharing.

3.2.3 Secure Multi-Party Computation (SMPC)

Secure Multi-Party Computation (SMPC), built upon secret sharing and additive homomorphic encryption, enables participants to perform secure training and aggregation without revealing local updates [14]. In terms of systematic implementation: Tetrad (4PC, active security) achieves approximately 4x and 5x overall acceleration for training and inference respectively in wide-area network settings, compressing communication for multiplication gates to 5 ring elements—outperforming competing systems [13]. The advantage of this approach lies in providing formalized

confidentiality and correctness under adversarial models without relying on trusted third parties, making it suitable for multi-organization collaboration and privacy-preserving inference services. Its limitation is the high communication and latency costs due to multiple rounds, making it sensitive to bandwidth and round-trip delays. Deployment requires comprehensive reduction of system overhead through batch processing, operator replacement, and topology optimization.

3.3 Variant Methods

3.3.1 Multi-Party Homomorphic Encryption with Model Aggregation (MPHE-1)

MPHE-1 combines multi-party homomorphic encryption with threshold or subset decryption: clients encrypt and upload gradients or weights for selected coordinates, while servers perform linear summation within the ciphertext domain. Once the threshold is met, parties collaboratively decrypt the result. This approach requires only two communication rounds per aggregation and inherently tolerates client disconnections [8]. The paper details the protocol and interface specifications, reporting accuracy degradation ranges of approximately 1 to 3 percentage points on typical tasks, which correlate with homomorphic parameter settings and aggregation granularity [8]. This approach offers advantages including fewer communication rounds, a simple interface, and fully encrypted aggregation throughout, facilitating integration into existing federated learning frameworks. Additionally, the threshold mechanism supports dynamic joining and enhanced robustness. Its limitations include insufficient systematic quantification on public benchmarks and large models, lacking detailed metrics on per-round communication volume and end-to-end latency under plaintext settings. When security levels increase, the computational cost of homomorphic operations and ciphertext size rises, necessitating further optimization through parameter tuning, sparsity strategies, and parallel batch processing.

3.3.2 Multi-Key Homomorphic Encryption with Secure Aggregation (MHESA)

MHESA employs multi-key homomorphic encryption where each client independently encrypts local updates using their public key. The server aggregates results by summing ciphers, then restores the aggregated outcome through multi-key transformation and threshold decryption, achieving key independence and offline tolerance [9]. System experiments demonstrate: under no-dropout conditions, communication requires two rounds; an additional round is needed when dropouts occur. In a scenario with 500 clients and a 30% dropout rate, overall efficiency improves by approximately 24 times compared to the baseline scheme. Additionally, training time is reduced by about 15%, communication volume decreases by approximately 25%, and sub-second timing statistics for both client and server phases are provided [9]. The paper does not provide a quantified accuracy gap compared to plaintext training, but the authors claim accuracy is preserved. The approach's advantages include low iteration counts, strong fault tolerance, and robust key independence, making it suitable for highly heterogeneous and high-packet-loss environments such as wearable device aggregation, vehicle-to-everything (V2X), and industrial IoT. Its limitations lie in the implementation complexity introduced by threshold decryption and multi-key management; scaling up still requires engineering optimizations in homomorphic parameters, batch processing, and parallel communication.

3.3.3 Quantized Secure Federated Learning (QSFL)

QSFL quantizes gradients or weights into fixed-point integers (optionally with unbiased random rounding) at the client to reduce local computation and uplink bandwidth. It then employs secure multi-party aggregation to sum the quantized values and performs inverse quantization with a known step size to obtain approximate aggregated results, balancing privacy and efficiency [10]. Public experiments demonstrate an accuracy degradation range of approximately 0.5 to 1.2 percentage points, a reduction in end-to-end latency of about 30% to 40%, and a decrease in communication volume of approximately 40%, indicating excellent energy efficiency on resource-constrained terminals [10]. The method's strengths include simplicity, direct benefits, and seamless integration with existing secure aggregation protocols. Its limitations stem from quantization introducing controllable approximation errors and the non-smoothness of backpropagation. Training stability depends on joint parameter tuning of bit width, step size, and learning rate. Transferability across different tasks and models requires further validation. Engineering practices often combine bit width search, step size adaptation, and distillation to mitigate accuracy loss.

3.4 Comparative Analysis Across Six Dimensions

To facilitate readers' rapid comprehension across unified dimensions, Table 1 summarizes the six methods' performance across six dimensions: dataset type, precision-loss (pp), computational overhead, communication cost,

privacy strength, and application domain. All values are directly extracted from the original text, with standardized annotations referenced in Section 3.1. This table aims to provide a horizontally comparable perspective, facilitating an understanding of the focus and applicability boundaries of each method and laying the groundwork for subsequent analysis and discussion.

Table 1 Comparative Analysis of Privacy-Preserving Methods in Federated Learning Across Six Dimensions

Technology	Dataset Type	Accuracy Loss (pp)	Computational Overhead	Communication Cost	Privacy Strength	Application Domain
DP	MNIST, CIFAR-10	ε=0.5 → 90% (↓ 8.3) ε=2→95%(↓ 3.3) ε=8→97%(↓ 1.3)	Training time ≈ 1.5× non-DP model, PCA reduced to 0.15×	≈0 (local clipping + noise addition)	Medium	Input method: Advertising; Statistical release
HE	MNIST, CIFAR-10, Texas, Purchase	<0.1	MNIST 3.1×, CIFAR-10 17.1×	42.7 MB → 871.9 MB (↑ 20.4×); Encrypting only 20% of weights: 209.8 MB (↓ 4.15×)	High	Medical imaging; Financial risk control; Encrypted queries; IoT cloud integration
SMPC	MNIST, CIFAR-10	Equivalent to plaintext (unquantized difference)	Inference 6×–113×; Training 4×–5×	Multi-gate communication with 5 ring elements (4PC)	High	Cross-hospital medical care; Multi-party risk control; Privacy-preserving inference
MPHE-1	MNIST, CIFAR-10 (Aggregation Protocol)	1–3	Two-round communication O(L)	Unreported MB/round; supports disconnections	High	Edge/IoT aggregation; cross-campus updates; mobile device disconnect scenarios
MHESA	IoT, Healthcare	Equivalent to plaintext (unquantized difference)	Training time ↓ 15%	Communication volume ↓ 25%	High	Wearable convergence; V2X/ Industrial IoT; Remote monitoring
QSFL	MNIST, Fashion-MNIST	0.5–1.2	Computational latency ↓ 30–40%	Communication volume ↓ ≈ 40%	High	Edge devices; Mobile vision/ speech

4 Conclusion

4.1 Key Findings

Based on the comparisons in Section 3, privacy-preserving methods in federated learning can be summarized as a clear "triadic trade-off curve": privacy strength—model utility—system cost. Under moderate privacy budgets, DP achieves near-zero additional communication and explicit auditable privacy guarantees with only minor accuracy loss, making it suitable for resource-constrained edge scenarios. HE enables end-to-end encrypted protection on the server side while maintaining near-plaintext accuracy, but incurs significant training time and encryption overhead; selective encryption can substantially reduce communication and storage demands. SMPC provides formal security and protocol-level optimization under semi-honest and malicious adversary models, yet remains constrained by multi-round interactions and bandwidth limitations. Three engineered variants further refine the options: MPHE-1 aggregates multi-party homomorphic encryption with key decryption into a single round, sacrificing limited quantization proofs and high security parameter costs, making it suitable for low-round deployments with unstable networks; MHESA combines multi-key HE with threshold functions to maintain efficiency during disconnections, at the cost of complex key management, ideal for highly heterogeneous environments with frequent disconnections; QSFL employs integer quantization and secure aggregation to reduce latency and communication overhead, sacrificing minor precision loss and requiring meticulous parameter tuning, making it suitable for mobile and bandwidth-constrained environments. Overall, there is no single optimal standard for evaluating these methods; their suitability depends on factors such as bandwidth and round-trip latency, client stability, model scale, and compliance requirements.

4.2 Implications for Research and Practice

For practical deployment, contextual selection should be made based on constraints: When bandwidth is tight or

mobile devices dominate, prioritize QSFL or moderately budgeted DP to trade minor utility costs for significant communication and latency improvements. Strictly compliant or plaintext-prohibited services are better suited for HE or SMPC. For significant disconnections or heterogeneous devices, MHESA and MPHE-1 offer advantages in iteration rounds and robustness. Large models may adopt a hybrid approach of "selective HE + secure aggregation" to balance privacy. For research work, standardize reporting and reproducibility metrics: report accuracy differences in percentage points, training overhead in multiples, and communication in MB/round and total volume. Clearly define threat models, privacy parameters, client scale, and disconnect settings, while publishing scripts, random seeds, and configuration files to enhance cross-paper comparability and reproducibility. At the algorithmic and system levels, focus respectively on: - Coordinating DP pruning thresholds with noise scheduling - Balancing encryption ratios and vectorization in HE - Optimizing batch processing and operator substitution in SMPC - Implementing bitwidth joint search and error compensation in QSFL.

4.3 Future Directions

Future research may advance in the following areas: First, establish a unified benchmark for large-scale visual and language models to standardize reporting of accuracy, end-to-end latency, communication, and energy consumption, covering DP, HE, SMPC, QSFL, and their hybrid schemes. Second, constructing adaptive hybrid orchestration frameworks that dynamically select combinations based on bandwidth, dropout rates, and privacy budgets to achieve context-aware training and aggregation workflows. Third, conducting large-scale experiments under non-iid and real-world network conditions to systematically characterize stable operating regions balancing privacy, utility, and cost, thereby supporting large-scale deployment.

References

[1] Dwork, C. (2006). Differential privacy. In Automata, languages and programming: 33rd international colloquium, ICALP 2006 (Vol. 4052, pp. 1–12). Springer.

[2] Liu YX, Chen H, Liu YH, Li CP. Privacy-preserving Techniques in Federated Learning. Journal of Software, 2022, 33(3): 1057-1092(in Chinese).

[3] Cheon J H, Kim A, Kim M, Song Y. Homomorphic encryption for arithmetic of approximate numbers. In: Takagi T, Peyrin T, eds. Advances in Cryptology – ASIACRYPT 2017. Lecture Notes in Computer Science, vol. 10624. Cham: Springer, 409–437, 2017.

[4] Teng WANG, Zheng HUO, Yaxin HUANG, Yilin FAN. Review on privacy-preserving technologies in federated learning[J]. Journal of Computer Applications, 2023, 43(2): 437-449.

[5] Yao, A. C. (1982). Protocols for secure computations. In Proceedings of the 23rd Annual Symposium on Foundations of Computer Science (FOCS '82) (pp. 160–164). IEEE.

[6] Goldreich, O., Micali, S., & Wigderson, A. (1987). How to play any mental game—A completeness theorem for protocols with honest majority. In Proceedings of the 19th Annual ACM Symposium on Theory of Computing (STOC '87) (pp. 218–229). ACM.

[7] Ben-Or, M., Goldwasser, S., & Wigderson, A. (1988). Completeness theorems for non-cryptographic fault-tolerant distributed computation. In Proceedings of the 20th Annual ACM Symposium on Theory of Computing (STOC '88) (pp. 1–10). ACM.

[8] Hosseini, E., Chen, S., & Khisti, A. (2024). Secure Aggregation in Federated Learning using Multiparty Homomorphic Encryption. arXiv preprint.

[9] Gao, Q., Sun, Y., Chen, X., Yang, F., & Wang, Y. (2024). An efficient multi-party secure aggregation method based on multi-homomorphic attributes. Electronics, 13(4), Article 671.

[10] Tran, A. T., Luong, T. D., & Pham, X. S. (2024). Privacy-preserving deep learning model with integer quantization and secure multi-party computation. Annals of Operations Research.

[11] 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 (CCS '16) (pp. 308–318). ACM.

[12] Hu, C., & Li, B. (2024). MASKCRYPT: Federated learning with selective homomorphic encryption. IEEE Transactions on Dependable and Secure Computing. Advance online publication.

[13] Koti, N., Ranellucci, S., Srinivasan, A., Rosulek, M., Trieu, N., Mandal, A., & Wang, X. (2022). Tetrad: Actively secure 4PC for secure training and inference. Proceedings of the Network and Distributed System Security Symposium (NDSS 2022).

[14] Wagh, S., Gupta, D., & Chandran, N. (2019). SecureNN: 3-party secure computation for neural network training. Proceedings on Privacy Enhancing Technologies, 2019(3), 26–49.