ERIS: Enhancing the Privacy–Utility–Scalability Trade-Off in Federated Learning via Federated Shard Aggregation

Dario Fenoglio[1,2]* **Pasquale Polverino**[1] **Akash Dhasade**[3]
Jacopo Quizi[1] **Martin Gjoreski**[1] **Marc Langheinrich**[1]

[1]Università della Svizzera italiana, Lugano, Switzerland
[2]Massachusetts Institute of Technology, Cambridge, MA, USA
[3]Carnegie Mellon University, Pittsburgh, PA, USA

Abstract

Scaling Federated Learning (FL) to billion-parameter models forces a challenging trade-off between privacy, scalability, and model utility. Existing solutions often tackle these challenges in isolation, sacrificing accuracy, relying on costly cryptographic tools, or introducing communication and optimization inefficiencies that affect convergence. We introduce ERIS, an FL framework centered on *Federated Shard Aggregation* (FSA), a novel mechanism that partitions each client update into non-overlapping shards whose aggregation is distributed across multiple client-side aggregators. FSA removes the central aggregation bottleneck, limits the information visible to any single observer, and preserves the centralized FL update after reassembly. ERIS can further readily integrate Distributed Shifted Compression (DSC) to reduce transmitted payloads and exposed coordinates. We prove that ERIS preserves convergence under standard assumptions and bounds mutual information leakage by the observable fraction of each update, decreasing with the number of client-side aggregators, and with the compression level when DSC is enabled. Experiments across image and text tasks, including large language models, show that ERIS achieves FedAvg-level utility while substantially reducing communication bottlenecks and improving robustness to membership inference and reconstruction attacks, without relying on heavy cryptography or utility-degrading perturbations.

1 Introduction

The widespread digitalization has led to an unprecedented volume of data being continuously recorded. However, much of this data is sensitive, introducing privacy risks and regulatory constraints that limit its usability [1]. Federated Learning (FL) has emerged as a distributed learning paradigm that enables multiple clients to collaboratively train machine learning (ML) models without directly sharing their private data [2]. By keeping data local, FL can unlock sensitive distributed data from hospitals, corporations, vehicles, and personal devices that would otherwise remain inaccessible. At the same time, the growing demand for high-capacity models, including foundation models and large language models (LLMs), makes practical FL increasingly dependent on three fundamental requirements: preserving privacy, scaling efficiently, and maintaining model utility [3–5].

Although FL avoids direct data sharing, the exchanged updates can still reveal sensitive information about the underlying training data [6]. Adversaries can exploit gradients or model updates to reconstruct private samples or infer whether specific records were used during training [7–17].

*Corresponding author: dario.fenoglio@usi.ch

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Existing privacy-preserving approaches mainly follow two directions. Cryptographic protocols can hide client updates from the server but introduce additional communication, computation, or hardware requirements, thereby hurting scalability [18–27]. Perturbation-based mechanisms, such as differential privacy (DP), or pruning, reduce leakage by modifying the transmitted updates, but often degrade model utility, especially in large models or low-data regimes [28–43]. Thus, effective privacy protection in FL remains tightly coupled with costly system overheads or accuracy loss.

Scaling FL to modern models further exposes the limits of centralized aggregation. In traditional FL, a central server collects client updates and redistributes the updated model every round, creating severe network-utilization imbalance and a single aggregation bottleneck as the number of clients grows [44]. This issue is amplified by billion-parameter models [45, 46], where transmitting full updates becomes prohibitively expensive. Fully decentralized architectures distribute communication across nodes, but rely on neighborhood-only aggregation, often reducing utility or slowing convergence depending on the topology [47–52]. Compression techniques, such as quantization or sparsification, reduce payloads, but aggressive compression can also affect convergence or degrade accuracy [53–56]. Similar limitations hold for parameter-efficient fine-tuning, which often remains outperformed by full-parameter fine-tuning [57, 58]. As a result, existing scalable FL methods address the central bottleneck or communication overhead at the cost of reduced model utility.

For FL to be practically useful, privacy and scalability should not come at the expense of the learning. However, existing methods introduce approximations, perturbations, or deviations from centralized aggregation that alter the optimization trajectory. DP and pruning can suppress informative gradient components [29–33, 38–41]; compression can introduce additional variance or require more rounds [54–56, 59]; and fully decentralized schemes often aggregate only within client-local neighborhoods, affecting convergence [47–50]. This creates a persistent trade-off: methods that improve privacy or scalability often fail to preserve the utility of centralized FedAvg-style collaboration.

To address these limitations, we propose ERIS, an FL framework designed to jointly satisfy privacy, scalability, and utility. At its core, ERIS introduces *Federated Shard Aggregation* (FSA), which partitions each client update into non-overlapping shards and distributes their aggregation across multiple client-side aggregators. Because the shards are disjoint and complete, clients recover after reassembly the same global update as induced by centralized aggregation, while no single aggregator ever observes a full client update. Furthermore, we show that FSA readily integrates with *Distributed Shifted Compression* (DSC) [60], a key component of modern systems that effectively reduces the number of transmitted parameters but does not, by itself, guarantee privacy. To the best of our knowledge, ERIS is the first FL framework that simultaneously provides FedAvg-equivalent utility, information-theoretic privacy amplification, and scalable distributed aggregation without relying on heavy cryptography or utility-degrading perturbations. Our key contributions are:

- **Federated Shard Aggregation.** We introduce FSA, a distributed aggregation mechanism that shards client updates across multiple aggregators. FSA removes the central aggregation bottleneck, limits the information visible to any single observer, and preserves the centralized FL update after reassembly.
- **Seamless integration with Distributed Shifted Compression.** We show that FSA naturally supports DSC as a pre-processing layer, reducing transmitted and exposed coordinates. This further enhances scalability and privacy while retaining the strong utility provided by FSA.
- **Theoretical guarantees and large-scale validation.** We provide convergence guarantees and information-theoretic privacy bounds showing that leakage decreases with the number of aggregators and, when DSC is enabled, with the compression level. Extensive experiments on four image and two text datasets—from small models to modern LLMs—and under two threat models against six SOTA baselines confirm ERIS's strong privacy–utility–scalability trade-off.

With ERIS, we hope to provide a foundation for next-generation FL systems where privacy and scalability can be strengthened without sacrificing model utility.

2 Background and Related Work

Federated Learning (FL). Traditional FL systems [2] consist of $K \in \mathbb{N}$ clients, denoted by $\mathcal{K} = \{1, 2, \ldots, K\}$, coordinated by a central server to collaboratively train an ML model over a distributed dataset D. Each client $k \in \mathcal{K}$ holds a private dataset $D_k = \{d_{k,s}\}_{s=1}^{S_k}$ with S_k samples. During each round, clients independently update model parameters $\mathbf{x}^t \in \mathbb{R}^n$ by minimizing a

nonconvex local loss $f(D_k; \mathbf{x}^t)$, producing stochastic gradients $\tilde{\mathbf{g}}_k^t$. After local training, each client transmits its gradients to the server, which aggregates them using a permutation-invariant operation, and updates the global model as $\mathbf{x}^{t+1} = \mathbf{x}^t - \lambda_t \tilde{\mathbf{g}}^t$, where λ_t is the learning rate. It then broadcasts $\mathbf{x}^{t+1}$ back to the clients for next round. In general, FL aims to minimize:

$$\arg\min_{\mathbf{x}} \frac{1}{K} \sum_{k=1}^{K} f(D_k; \mathbf{x}), \quad \text{where} \quad f(D_k; \mathbf{x}) := \frac{1}{S_k} \sum_{s=1}^{S_k} f(d_{k,s}; \mathbf{x}). \tag{1}$$

For brevity, we denote the loss function of the current model as $f(\mathbf{x}^t)$ for the entire dataset D, $f_k(\mathbf{x}^t)$ for the local dataset D_k, and $f_{k,s}(\mathbf{x}^t)$ for a single sample $d_{k,s}$, respectively.

Privacy-preserving FL. Although FL avoids direct data sharing, transmitted updates (*e.g.*, gradients) can still leak sensitive information [6–14, 61–69]. The central server, which collects full client updates, is therefore a primary vulnerability. Two dominant families of privacy-preserving methods are *cryptographic techniques* and *perturbation-based mechanisms*. Cryptographic techniques, such as secure aggregation [18–24] and trusted execution environments [25–27], mask client updates from the server, but introduce significant computational overhead or require specialized hardware. Perturbation-based mechanisms, such as local differential privacy (LDP) [29–37, 70–75] and gradient pruning [38–43, 53], reduce leakage by modifying updates: LDP clips and adds noise to provide formal guarantees, while pruning removes informative gradient components. However, both approaches often incur substantial utility degradation, especially for large models [76]. Recent works such as LotteryFL [77] and PriPrune [41] attenuate this trade-off through personalized pruning schemes, while other methods combine LDP with compression to balance privacy and communication efficiency [33, 34, 36, 60, 75], often at the cost of increased algorithmic complexity or slower convergence. A complementary line of work explores *partial-update schemes* that reduce exposure by transmitting or exchanging only subsets of model updates. Methods such as Ako [51] and Shatter [78] partition model updates to limit what any single participant observes. However, their partial or neighborhood-based exchanges do not recover the exact centralized aggregate at each round, potentially altering the optimization trajectory; in the case of Shatter, this also incurs more than twice the communication overhead. These limitations motivate alternative privacy mechanisms that can limit information exposure while preserving the efficiency and utility of centralized aggregation.

Scalable FL. Scalability is a central limitation of traditional FL systems, arising from single server bottleneck and high per-round communication cost. As the number of participating clients grows, the central server must collect and redistribute updates from all participants, creating severe network-utilization imbalance and server-side congestion. *Decentralized architectures* alleviate this bottleneck by distributing communication and computation across multiple nodes [49–52, 55, 79–85]. These methods commonly rely on peer-to-peer or gossip-based communication, where clients exchange updates with selected neighbors [51, 52, 79–84]. However, such neighborhood-based aggregation generally does not recover the exact centralized aggregate at each round, leading to client-specific models and making optimization dependent on the communication topology. *Multi-server FL* instead distributes coordination across multiple servers [86, 87], but typically partitions clients rather than individual updates, such that each server may still observe complete updates from its assigned clients unless additional cryptographic mechanisms are employed [88]. Another line of work employs *model and parameter partitioning*, together with collective communication primitives such as reduce-scatter and all-gather, to improve throughput, memory efficiency, or straggler tolerance [51, 89, 90]. These mechanisms primarily use partitioning for systems or communication efficiency rather than as an information-exposure boundary. As model sizes increase—reaching billions of parameters—the amount of data transmitted per round grows substantially, making large-scale FL increasingly impractical. *Compression techniques* reduce this payload via quantization [56, 59, 91–99] and sparsification [55, 56, 59, 60, 92, 96, 98, 100–102], but naive or aggressive compression can degrade utility or slow convergence, undermining scalability gains [59, 91, 102]. These limitations motivate distributed aggregation mechanisms that can alleviate central bottlenecks while preserving the exact centralized update and limiting information exposure.

3 ERIS

In this work, we propose ERIS, a novel FL framework designed to address key limitations in privacy, scalability, and utility. This section states the design goals (Section 3.1), describes the

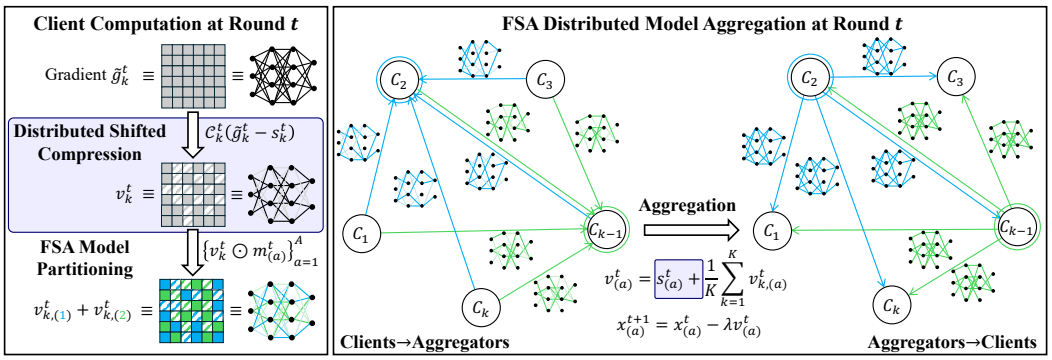

Figure 1: Illustration of ERIS at training round t for two aggregators ($A=2$). **Left:** each client optionally applies DSC and then performs FSA partitioning, generating shards $\mathbf{v}_{k,(a)}^t$ sent to aggregators C_2 and C_{k-1}. **Right:** each aggregator collects and aggregates the corresponding shards across clients to produce partial updated models $\mathbf{x}_{(a)}^{t+1}$, which are sent back to the clients.

pipeline (Section 3.2), provides theoretical foundations for the convergence of the learning process (Section 3.3), and establishes an information-theoretic upper bound on privacy leakage (Section 3.4).

3.1 Design Goals

We aim to collaboratively train a global model under three requirements: (i) *privacy*: the information available to any single honest-but-curious observer should be limited, preventing direct access to complete client updates; (ii) *scalability*: the protocol should remove the central aggregation bottleneck and balance communication and computation across multiple nodes; and (iii) *utility*: the distributed protocol should preserve the collaborative update induced by centralized aggregation, avoiding approximation errors or perturbations that degrade the learning process.

3.2 The ERIS Pipeline

The ERIS pipeline is detailed in Algorithm 1 and shown in Figure 1. In each round t, every client k computes a local update $\tilde{\mathbf{g}}_k^t$ on its private dataset D_k. Now, instead of sending the update directly to a central server, ERIS leverages *Federated Shard Aggregation* (FSA) to distribute the aggregation task across multiple client-side aggregators. We first present our core FSA mechanism applied to original gradient updates *i.e.*, uncompressed updates. We then present the variant with Distributed Shifted Compression (DSC) which further enhances scalability and privacy within FSA. The lines corresponding to DSC are uniquely highlighted in Algorithm 1.

3.2.1 Federated Shard Aggregation (FSA)

FSA is the main building block of ERIS. Instead of sending a complete client update to a central server, each client partitions its update into A disjoint shards and sends each shard to a different aggregator. The aggregators independently combine their assigned shards and return the corresponding model segments to the clients, which then reassemble the global model.

Shard-wise model partitioning. More formally, let $\mathbf{v}_k^t$ denote the pre-partition update of client k at round t. In the basic FSA variant, no compression is applied and $\mathbf{v}_k^t = \tilde{\mathbf{g}}_k^t$. FSA partitions $\mathbf{v}_k^t$ into A disjoint shards using a set of binary masks $\{\mathbf{m}_{(a)}^t\}_{a=1}^A \subset \{0,1\}^n$, where each mask selects the coordinates assigned to aggregator a. The masks satisfy:

$$\textit{Disjointness:} \quad \mathbf{m}_{(a)}^t \odot \mathbf{m}_{(a')}^t = \mathbf{0} \quad \forall a \neq a',$$

$$\textit{Completeness:} \quad \sum_{a=1}^{A} \mathbf{m}_{(a)}^t = \mathbf{1}_n,$$

where $\mathbf{1}_n$ is the all-ones vector and $\odot$ denotes element-wise multiplication. Client k then constructs the shard assigned to aggregator a as $\mathbf{v}_{k,(a)}^t = \mathbf{v}_k^t \odot \mathbf{m}_{(a)}^t$, where $a \in \{1,\ldots,A\}$, and transmits each shard $\mathbf{v}_{k,(a)}^t$ to its corresponding aggregator. Because the masks are disjoint and complete, the

Algorithm 1: ERIS: Federated Shard Aggregation with optional Distributed Shifted Compression

Input: Initial global model $\mathbf{x}^0$, number of aggregators A, learning rate λ_t, number of clients K, number of rounds T, reference vectors $\mathbf{s}_k^0 = \mathbf{0}$ and $\mathbf{s}_{(a)}^0 = \mathbf{0}$ for DSC.

Output: Final global model $\mathbf{x}^T$

for $t = 0, 1, \ldots, T-1$ **do**

  `// Client-side operations`

  for ***each client*** $k \in \{1, \ldots, K\}$ ***in parallel*** **do**

    Compute local stochastic gradient $\tilde{\mathbf{g}}_k^t$;

    Shifted compression: $\mathbf{v}_k^t = \mathcal{C}_k^t\left(\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t\right)$; `// without compression:` $\mathbf{v}_k^t = \tilde{\mathbf{g}}_k^t$

    Shard-wise partitioning: $\{\mathbf{v}_{k,(a)}^t\}_{a=1}^A = \{\mathbf{v}_k^t \odot \mathbf{m}_{(a)}^t\}_{a=1}^A$;

    Send each shard $\mathbf{v}_{k,(a)}^t$ to aggregator a, for $a = 1, \ldots, A$;

    Update client reference vector: $\mathbf{s}_k^{t+1} = \mathbf{s}_k^t + \gamma_t \mathbf{v}_k^t$

  `// Aggregator-side operations`

  for ***each aggregator*** $a \in \{1, \ldots, A\}$ ***in parallel*** **do**

    Aggregate assigned shard: $\mathbf{v}_{(a)}^t = \frac{1}{K}\sum_{k=1}^K \mathbf{v}_{k,(a)}^t$;

    Compensate global shift: $\mathbf{v}_{(a)}^t \leftarrow \mathbf{s}_{(a)}^t + \mathbf{v}_{(a)}^t$

    Update shard of the global model: $\mathbf{x}_{(a)}^{t+1} = \mathbf{x}_{(a)}^t - \lambda_t \mathbf{v}_{(a)}^t$;

    Update aggregator reference: $\mathbf{s}_{(a)}^{t+1} = \mathbf{s}_{(a)}^t + \gamma_t \frac{1}{K}\sum_{k=1}^K \mathbf{v}_{k,(a)}^t$

    Broadcast updated shard $\mathbf{x}_{(a)}^{t+1}$ to all clients ;

  Each client k reassembles the global model: $\mathbf{x}_k^{t+1} = \sum_{a=1}^A \mathbf{m}_{(a)}^t \odot \mathbf{x}_{(a)}^{t+1}$;

original update can be recovered by summing the shards as $\mathbf{v}_k^t = \sum_{a=1}^A \mathbf{v}_{k,(a)}^t$. Thus, FSA distributes each update across aggregators without discarding any coordinate.

The A aggregators can be selected before training, e.g., through decentralized peer-sampling or gossip protocols [103], or consensus-based committee-selection mechanisms [104]. To limit the coordinate coverage accumulated by any individual non-colluding aggregator over training, we consider a deployment in which the aggregator set and shard assignments remain fixed across rounds, except when an aggregator becomes unavailable and reconfiguration is required.

Distributed model aggregation. Each aggregator a receives the shards $\{\mathbf{v}_{k,(a)}^t\}_{k=1}^K$ from the participating clients and uses a permutation-invariant aggregation over its assigned coordinates, such as:

$$\mathbf{v}_{(a)}^t = \frac{1}{K}\sum_{k=1}^K \mathbf{v}_{k,(a)}^t. \tag{2}$$

The aggregator then updates the corresponding shard of the global model as $\mathbf{x}_{(a)}^{t+1} = \mathbf{x}_{(a)}^t - \lambda_t \mathbf{v}_{(a)}^t$. Finally, each aggregator broadcasts its updated shard $\mathbf{x}_{(a)}^{t+1}$ to the clients, which reassemble the full model as $\mathbf{x}_k^{t+1} = \sum_{a=1}^A \mathbf{m}_{(a)}^t \odot \mathbf{x}_{(a)}^{t+1}$. By the disjointness and completeness of the masks, this reassembled model is equivalent to the model obtained by aggregating the full client updates centrally.

Collectively, FSA eliminates the single server bottleneck while still preserving the exact update induced by centralized FL, thereby maintaining strong *utility*. By distributing shards across multiple aggregators, it balances communication and computation, improving *scalability*. At the same time, each aggregator observes only a subset of every client update, limiting the information accessible to any single honest-but-curious observer and thus enhancing *privacy*. This combination distinguishes ERIS from decentralized methods that offer scalability but sacrifice centralized aggregation and from perturbation based methods that offer privacy but compromise utility.

3.2.2 Amplifying FSA with Distributed Shifted Compression

FSA already provides the core privacy and scalability benefits of ERIS. However, the same architecture can naturally support additional privacy- and communication-enhancing mechanisms. In this work, we instantiate this idea with DSC, an optional layer that further reduces both transmitted and exposed coordinates while preserving the FSA structure and utility. We use the standard unbiased compressor definition adopted in communication-efficient FL [59, 92, 96].

Definition 3.1 (Compression operator)**.** A randomized map $\mathcal{C} : \mathbb{R}^n \to \mathbb{R}^n$ is an ω-compression operator if, for all $\mathbf{x} \in \mathbb{R}^n$, it satisfies $\omega \geq 0$ and:

$$\mathbb{E}[\mathcal{C}(\mathbf{x})] = \mathbf{x}, \qquad \mathbb{E}\big[\|\mathcal{C}(\mathbf{x}) - \mathbf{x}\|^2\big] \leq \omega\|\mathbf{x}\|^2. \tag{3}$$

Def. 3.1 covers common randomized compressors such as quantization and sparsification [45, 60, 93, 59, 92]. For instance, random sparsification can be written as $\mathcal{C}_k^t(\mathbf{x}) = \mathbf{x} \odot \mathbf{m}_{\mathcal{C}_k^t}$, where each entry of $\mathbf{m}_{\mathcal{C}_k^t}$ equals $1/p_k$ with probability p_k and 0 otherwise, giving $\mathbb{E}[\mathbf{m}_{\mathcal{C}_k^t}] = \mathbf{1}_n$ and $\omega = (1 - p_k)/p_k$.

To reduce communication while preserving favorable convergence behavior, we extend shifted compression [60] to the distributed FSA setting. Each client maintains a reference vector $\mathbf{s}_k^t$ and sends the compressed shifted update $\mathbf{v}_k^t = \mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t)$, then updates $\mathbf{s}_k^{t+1} = \mathbf{s}_k^t + \gamma_t \mathbf{v}_k^t$ to track the local update direction over time. The vector $\mathbf{v}_k^t$ is then passed to FSA, which shards and distributes it as before. Hence, DSC augments FSA by simply changing the vector that FSA aggregates. Since compression is applied before partitioning, aggregator a receives only a compressed shard. To compensate for the shift, it maintains a shard-level reference $\mathbf{s}_{(a)}^t$ and replaces Equation (2) with:

$$\mathbf{v}_{(a)}^t = \mathbf{s}_{(a)}^t + \frac{1}{K}\sum_{k=1}^{K} \mathbf{v}_{k,(a)}^t, \qquad \mathbf{s}_{(a)}^{t+1} = \mathbf{s}_{(a)}^t + \gamma_t \frac{1}{K}\sum_{k=1}^{K} \mathbf{v}_{k,(a)}^t. \tag{4}$$

DSC amplifies both scalability and privacy. By reducing the number of transmitted parameters before FSA partitioning, it decreases client upload and per-aggregator traffic while also reducing the coordinates exposed in each shard. Importantly, privacy in ERIS does not rely on aggressive compression, which can otherwise harm utility: FSA already limits each aggregator to a subset of the update, while DSC can further shrink this observable subset. This modularity also shows that ERIS can incorporate other update transformations beyond DSC before the FSA partitioning, including DP (see Section 4.2), with the corresponding aggregation-side correction, when needed.

3.3 Theoretical Analysis of Convergence and Utility

FSA across A aggregators preserves the exact update induced by centralized FL (Section 3.2.1). In Appendix B, we further formally establish that FSA incurs no information loss or deviation from the centralized aggregation trajectory. Building on this, we analyze the optional DSC layer on top of FSA, showing that ERIS preserves favorable convergence behavior under compressed updates.

Theorem 3.2 (Utility of ERIS with DSC)**.** *Consider* ERIS *under the standard smoothness and unbiased estimator assumptions (Appendix A), with compression operators $\mathcal{C}_k^t$ satisfying Definition 3.1. Let L denote the smoothness constant, and let $\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4, \theta$ and Δ^t be as defined in Assumption A.2. Let $\alpha = \frac{3\beta\Gamma_1}{2(1+\omega)L^2\theta}$, for any $\beta > 0$, and let the learning rate be defined as:*

$$\lambda_t \equiv \lambda \leq \min\Big\{\frac{\sqrt{\beta K}}{\sqrt{1 + 2\alpha\Gamma_4 + 4\beta(1+\omega)}(1+\omega)L}, \frac{1}{\big(1 + 2\alpha\Gamma_4 + 4\beta(1+\omega) + 2\alpha\Gamma_3/\lambda^2\big)L}\Big\} \tag{5}$$

with shift stepsize $\gamma_t = \sqrt{(1+2\omega)/(2(1+\omega)^3)}$. Then, ERIS *satisfies the utility bound:*

$$\frac{1}{T}\sum_{t=0}^{T-1} \|\nabla f(\mathbf{x}^t)\|^2 \leq \frac{2\Phi_0}{\lambda T} + \frac{3\beta\Gamma_2}{(1+\omega)L\lambda}, \tag{6}$$

where $\Phi_0 := f(\mathbf{x}^0) - f^* + \alpha L\Delta^0 + \frac{\beta}{KL}\sum_{k=1}^{K} ||\nabla f_k(\mathbf{x}^0) - \mathbf{s}_k^0||^2$. Equation 6 implies that the asymptotic utility of ERIS *is governed by the gradient estimator variance Γ_2, which vanishes for lower-variance estimators such as SVRG/SAGA [105, 106]. Similarly, a larger local batch size reduces gradient variance, leading to improved convergence, with $\Gamma_2 = 0$ when full local gradients are used.*

In contrast to prior communication-efficient privacy-preserving FL methods [36, 60, 107], the bound in (6) depends primarily on the gradient-estimator variance Γ_2 and is independent of the specific privacy-preserving mechanism applied; notably, it contains no term that grows with T. The proof and a comparison with existing DP and compressed FL bounds are provided in Appendix C.

3.4 Theoretical Analysis of Privacy Guarantees

We analyze ERIS under the standard *honest-but-curious* threat model [108–110], where an adversary observes and stores transmitted updates, e.g. via eavesdropping or a compromised aggregator/server,

and attempts to infer sensitive information about clients' private data D_k. FSA limits the adversary's view by ensuring that any fixed aggregator observes only one shard of each client update. DSC further reduces this view by retaining only a fraction p of the coordinates before partitioning. To quantify privacy, we bound the mutual information between D_k and the adversary's view $\mathbf{v}_{k,(a)}^t$ over rounds.

Theorem 3.3 (Privacy guarantee of ERIS with DSC)**.** *Let $\mathbf{v}_{k,(a)}^t = (\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t) \odot \mathbf{m}_{\mathcal{C}_k^t} \odot \mathbf{m}_{(a)}^t$ denote the a-th transmitted shard of client k at round t, where $\mathbf{m}_{\mathcal{C}_k^t}$ is a compression mask satisfying Definition 3.1 with probability p, and $\mathbf{m}_{(a)}^t$ selects one of A disjoint shards. Write $\mathbf{U}_k^t := \tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t$. Conditional on the history $\mathcal{H}_t$ of past shards and masks, assume the masks are independent of $(D_k, \mathbf{U}_k^t)$ and expose np/A coordinates in expectation. Assume $I(D_k; U_{k,i}^t \mid \mathcal{H}_t = h, \mathbf{U}_{k,J}^t) \le C_{\max} < \infty$ uniformly over i, t, $J \subseteq \{1, \dots, n\} \setminus \{i\}$, and almost every h. Then, under the honest-but-curious model, the mutual information over T rounds satisfies:*

$$I_k = I\big(D_k; \{\mathbf{v}_{k,(a)}^t\}_{t=0}^{T-1}\big) \le n\, T\, \frac{p}{A}\, C_{\max}, \tag{7}$$

where n is the model size and $C_{\max}$ bounds the per-coordinate mutual information conditional also on previously revealed coordinates within the round.

Theorem 3.3 shows that the privacy-leakage bound scales with the number of observable coordinates over time, nTp/A. For a common $C_{\max}$, FSA reduces this upper bound by a factor $1/A$, while DSC contributes the retention factor p. This captures the most general setting: when DSC is disabled, $p = 1$ and the bound reduces to the FSA-only case $I_k \le nTC_{\max}/A$. The absolute leakage depends on $C_{\max}$, which is model-, data-, and training-dependent. Appendix D.1 provides an operational Gaussian-channel instantiation of $C_{\max}$, while Appendix F.1 reports a corresponding empirical audit. Full proofs and the extension to colluding adversaries are provided in Appendix D. These theoretical findings are corroborated by our experiments.

4 Results

We now evaluate ERIS along all three dimensions: privacy, scalability, and utility. We detail the experimental setup in Section 4.1 and present the results in Section 4.2.

4.1 Experimental Setup

Datasets. We evaluate ERIS on six publicly available datasets spanning image classification and text generation. For image classification, we use MNIST [111] and CIFAR-10 [112]; for text classification, IMDB Reviews [113]; and for text generation, CNN/DailyMail [114]. To evaluate data reconstruction attacks, we additionally use LFW [115] and ImageNet [116]. Datasets are randomly partitioned among K clients (K=10 for CNN/DailyMail, K=25 for IMDB, and K=50 for the others); while non-IID scenarios are generated using a Dirichlet distribution with $\alpha \in \{0.2, 0.5\}$. We initialize GPT-Neo [117] (1.3B) and DistilBERT [118] (67M) from pre-trained checkpoints for CNN/DailyMail and IMDB, respectively, and subsequently perform full-parameter federated fine-tuning. ResNet-9 (1.65M) for CIFAR-10, ResNet-18 [119] (11.7M) for ImageNet, and LeNet-5 [120] (62K) for MNIST and LFW are instead trained from scratch. Unless otherwise specified, all experiments use 5-fold cross-validation, and reported results are averaged across folds. Training hyperparameters are detailed in Appendix E.1.

Baselines. We compare ERIS against several SOTA methods targeting privacy, scalability, or their trade-off in FL: *Ako* [51] and *Shatter* [78], decentralized approaches with partial gradient exchange; *SoteriaFL* [60], which combines centralized shifted compression with DP; *PriPrune* [41], a pruning strategy that withholds the most informative gradient components from communication; and *LDP* [35]. We also include *FedAvg* [2] as the standard baseline with no defenses or compression, and report results for an idealized upper bound (*Min. Leakage*), where clients transmit no gradients and the attack is applied only to the last-round global model.

Privacy Attacks. Under the standard honest-but-curious model, we assume the attacker is a compromised aggregator or server observing client-transmitted updates at every round. All methods use the same round budget T, capped at FedAvg convergence; full exposure details are provided in Appendix E. We evaluate six representative attacks across two widely studied categories: *Membership*

Table 1: Mean test performance (ROUGE-1 for CNN/DailyMail, accuracy for others) and MIA accuracy, averaged over varying local sample sizes. Configuration: ϵ=10; pruning rate $p \in \{0.1, 0.2, 0.3\}$ on IMDB/CNN-DailyMail, and $p \in \{0.01, 0.05, 0.1\}$ on others.

Method	CNN/DailyMail – GPT-Neo		IMDB – DistilBERT		CIFAR-10 – ResNet9		MNIST – LeNet5	
	R-1 (↑)	MIA Acc. (↓)	Acc. (↑)	MIA Acc. (↓)	Acc. (↑)	MIA Acc. (↓)	Acc. (↑)	MIA Acc. (↓)
FedAvg	33.22±0.99	97.94±0.63	79.60±0.83	68.21±1.36	34.86±0.31	68.46±0.96	88.91±0.35	65.11±0.78
FedAvg (ϵ, δ)-LDP	26.00±0.28	51.98±3.13	53.97±0.04	50.55±1.18	19.00±0.47	63.35±0.85	61.03±1.03	57.24±0.59
SoteriaFL (ϵ, δ)	25.40±0.70	52.14±2.97	54.24±0.15	51.25±1.19	17.18±0.24	58.83±0.56	57.27±0.88	57.13±0.56
PriPrune (p_1)	24.67±4.64	71.35±2.83	74.15±1.00	66.36±1.13	26.30±0.39	65.67±0.84	77.41±1.52	62.21±1.01
PriPrune (p_2)	24.67±4.64	71.35±2.83	66.30±2.14	63.61±1.11	11.24±0.71	56.55±0.78	27.36±1.04	52.69±0.83
PriPrune (p_3)	24.67±4.64	71.35±2.83	60.32±1.98	60.54±1.03	10.01±0.01	54.86±0.77	17.83±0.60	52.01±0.80
Shatter	31.95±0.71	70.67±4.03	76.94±1.40	57.41±2.01	12.40±1.85	63.00±2.01	15.86±4.82	56.23±1.50
ERIS	33.22±1.01	70.55±3.95	79.30±0.91	57.40±1.99	34.84±0.64	63.02±1.98	88.87±0.67	56.14±1.47
ERIS (+DSC)	32.83±0.78	69.55±3.94	79.07±0.80	56.31±0.81	34.68±0.48	60.48±0.91	89.00±0.23	55.97±0.77
Min. Leakage	33.23±0.99	60.53±4.83	79.68±0.36	55.58±0.76	34.92±0.29	58.85±0.93	88.90±0.40	55.22±0.64

Inference Attacks (MIA) and *Data Reconstruction Attacks (DRA)*. For MIA, we adopt the privacy auditing framework of Steinke et al. [121], repeating the evaluation at each round for every client; for text generation, we adapt the SPV-MIA of Fu et al. [122] to our auditing setting. Reported results correspond to the maximum, over all T rounds, of the average MIA accuracy across K clients. Representative per-round privacy–utility trajectories are reported in App. F.11. For DRA, we consider the strongest white-box threat model, which assumes access to the gradient of a single training sample, and implement DLG [12], iDLG [13], and ROG [8], with the latter specifically designed for reconstruction from obfuscated gradients. We further include GGL [7], an adaptive gradient inversion attack that accounts for the gradient transformation induced by the privacy defense.

4.2 Numerical Experiments

Effect of FSA and DSC on privacy. We first isolate the two mechanisms that reduce the adversary's view in ERIS. Figure 2 (left) evaluates FSA by varying the number of aggregators A on MNIST. Consistent with Theorem 3.3, increasing A reduces privacy leakage by limiting each aggregator to a smaller shard of the client update, without affecting model accuracy. The observed trend follows the $1/A$ factor in the mutual-information bound. Figure 2 (right) evaluates DSC with $A=50$ fixed: stronger compression (higher ω, i.e., lower retention probability p) further reduces MIA accuracy toward the idealized minimum-leakage baseline. These results validate the decomposition in Theorem 3.3: FSA provides the core privacy amplification through sharding, while DSC further shrinks the observable subset. Appendix F.3 quantifies the corresponding effect of DSC on model utility. For DRA, we find that compression alone is insufficient, especially against ROG (Table 7), whereas FSA is highly effective: even with $A=2$ (i.e., half of the update exposed), reconstructions are highly distorted and no longer preserve meaningful features of the original.

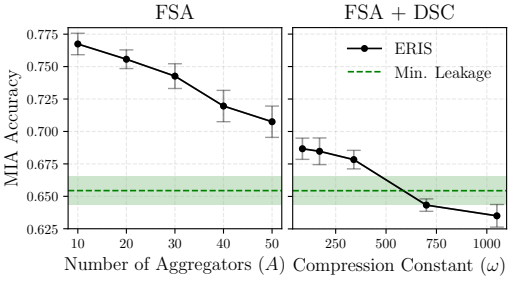

Figure 2: Effect of FSA (**left**) and DSC (**right**).

Balancing Utility and Privacy. To evaluate the utility–privacy trade-off, we benchmark ERIS with DSC against SOTA baselines across settings that influence memorization and overfitting. First, we vary model capacity, a key factor in memorization, spanning from large-scale 1.3B-parameter model on CNN/DailyMail to a lightweight 62K-parameter model on MNIST. Second, we control overfitting by varying the number of local training samples, from 4 to 128. Figure 3 shows that ERIS consistently maintains high utility, on par with non-private FedAvg, while significantly reducing privacy leakage—approaching the idealized *Min. Leakage* scenario. In contrast, privacy-preserving methods such as FedAvg-LDP, PriPrune, and SoteriaFL reduce leakage only at the cost of degraded performance. This confirms prior findings [76] that DP can substantially impair utility, particularly for large models, where low leakage may largely reflect the model's inability to learn the task. Notably, while Shatter's partial gradient exchange offers privacy protection comparable to or weaker than ERIS, its fragmented collaboration substantially slows convergence, especially when models are trained from scratch. Table 1 summarizes mean utility and MIA accuracy over client training-sample

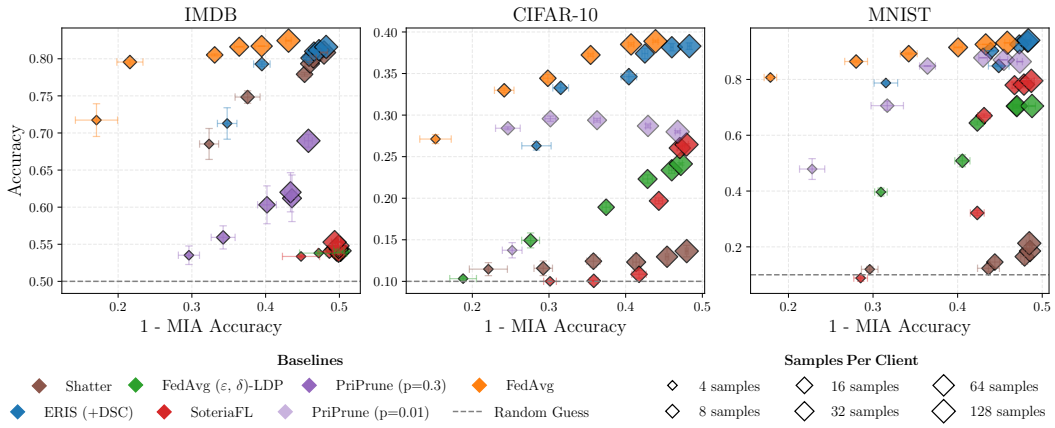

Figure 3: Comparison of test accuracy and (1- MIA) accuracy across varying model capacities (one per dataset) and client-side overfitting levels, controlled via the number of training samples per client.

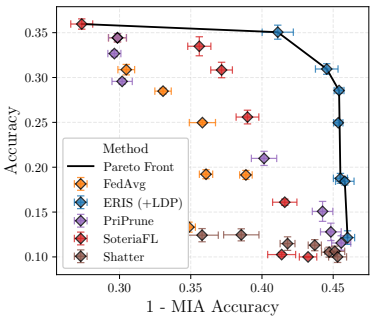

Figure 4: Pareto analysis of utility and privacy on CIFAR-10.

Table 2: Scalability comparison: per-client upload and minimum distribution time per round.

Method	CNN/DailyMail		CIFAR-10	
	Upload	Dist. Time	Upload	Dist. Time
FedAvg (-LDP)	5.2GB (100%)	5200s	6.6MB (100%)	33s
Shatter	5.2GB (100%)	780s	6.6MB (100%)	1.32s
PriPrune (p_1)	4.68GB (90%)	4940s	6.53MB (99%)	32.84s
PriPrune (p_2)	4.16GB (80%)	4680s	6.27MB (95%)	32.17s
PriPrune (p_3)	3.64GB (70%)	4420s	5.9MB (90%)	31.35s
SoteriaFL	0.26GB (5%)	2730s	0.33MB (5%)	17.32s
ERIS	4.68GB (90%)	468s	6.47MB (98%)	0.65s
ERIS (+DSC)	46.8MB (1%)	236s	0.04MB (0.6%)	0.33s

sizes, confirming that ERIS achieves the strongest utility–privacy trade-off among all baselines. Appendix F.8 reports the same experiments under non-IID settings, with consistent conclusions.

Pareto Analysis under Varying Privacy Constraints. To further investigate the utility–privacy trade-off, we evaluate each method under varying strengths of its privacy mechanism: for DP-based methods, we vary ϵ and clipping thresholds; for PriPrune, the pruning rate; and for ERIS and Shatter, we add LDP on top of their masking mechanisms. Full settings are provided in Appendix F.10. Figure 4 plots accuracy against privacy (1−MIA accuracy) on CIFAR-10 with 16 training samples per client. The Pareto front contains solutions for which no method improves utility without increasing leakage, or reduces leakage without sacrificing utility. ERIS contributes a majority of the Pareto-optimal points, confirming its ability to balance privacy and utility more effectively than the baselines.

Scalability. Table 2 evaluates scalability in terms of per-client upload size and minimum distribution time per round, assuming 20MB/s bandwidth and using the same experimental setting as Table 1. The results show that ERIS improves scalability through two complementary mechanisms. First, FSA removes the central aggregation bottleneck by distributing communication across aggregators. Even without DSC, ERIS substantially reduces distribution time, from 5200s to 468s on CNN/DailyMail and from 33s to 0.65s on CIFAR-10, with larger gains as the number of aggregators increases (CNN/DailyMail: $A = 10$; CIFAR-10: $A = 50$). Second, DSC further reduces the transmitted payload, yielding the full ERIS gains: on CNN/DailyMail, upload size drops from 5.2GB in FedAvg to 46.8MB (1%) and distribution time to 236s; on CIFAR-10, communication drops to 0.6% of the full update and distribution time to 0.33s. Unlike prior decentralized methods, ERIS preserves the full collaborative update after reassembly, retaining the utility of centralized aggregation while scaling to billion-parameter models. A full comparison across additional FL and decentralized FL methods, including per-client upload/download, total communication, and distribution time for each dataset/model pair, is reported in Appendix F.2.

5 Discussion

Limitations. While ERIS achieves strong empirical and theoretical performance, it also introduces system-level trade-offs. First, distributing aggregation shifts coordination to clients, making reliability dependent on aggregator availability, client–aggregator connectivity, and client-side computational capacity. To assess robustness to aggregator and link failures, Appendix F.5 shows that accuracy remains nearly unchanged up to 70% aggregator dropout and 50% link failures, with degradation mainly due to slower convergence. In addition, to support heterogeneous deployments, FSA can allocate larger shards to stronger aggregators and smaller shards to weaker or bandwidth-constrained nodes, at the cost of worst-case leakage being determined by the largest shard rather than n/A. Our scalability analysis focuses primarily on communication costs, since payload transmission increasingly dominates end-to-end latency as model sizes and client populations grow, and abstracts from connection-establishment and synchronization overheads. A full end-to-end systems evaluation is left to future work. Second, ERIS's privacy amplification degrades under aggregator collusion: leakage increases with coalition size, as analyzed in Corollary D.2 and Figure 5, while remaining substantially lower than centralized FL in the evaluated settings. Importantly, this can be compensated by increasing A proportionally to the anticipated coalition size. We also remark that ERIS's information-theoretic guarantee is distinct from record-level DP: it bounds the information revealed about a client's dataset through the observed shards, but does not provide an (ϵ, δ)-DP guarantee for individual records. DP can be integrated with FSA when such guarantees are required.

Benefits. As alluded to in Section 3.3, FSA supports the execution of any centralized FL algorithm beyond FedAvg, such as FedAdam [123], FedYogi [123], FedNova [124], *etc*. since it exhibits no deviation from the centralized aggregation trajectory. Furthermore, FSA acts as a modular aggregation layer: transformations such as compression [92], DP [35, 60], or robust aggregations [125, 126] can be readily integrated while preserving FSA's benefits. Similarly, secure aggregation [20, 21] can be executed independently within each shard, limiting colluding aggregators to shard-level aggregates rather than individual client shards, without altering the resulting global update but at the cost of additional cryptographic overhead. We view this flexibility and modularity as key advantages of FSA, enabling ERIS to serve as a general foundation for scalable and privacy-preserving FL.

6 Conclusion

We introduced ERIS, an FL framework that jointly addresses privacy, scalability, and utility through Federated Shard Aggregation (FSA). By sharding client updates across multiple aggregators, FSA removes the central server bottleneck, limits the information visible to any single observer, and preserves the centralized FL update after reassembly. We further instantiate Distributed Shifted Compression (DSC) on top of FSA to reduce transmitted and exposed coordinates while preserving favorable convergence behavior. We provide convergence bounds and information-theoretic privacy guarantees, and validate them through extensive experiments across diverse datasets and model scales. Our results show that privacy and scalability in FL can be enhanced without sacrificing utility.

Acknowledgments and Disclosure of Funding

This study was funded by the Swiss National Science Foundation, and the European Union's Horizon Europe research initiative through the following projects: SmartCHANGE (Project No. 101080965), XAI-PAC (Grant No. Z00P2_216405), and starting grant "Multiresolution methods for unstructured data" (TMSGI2_211684). AD was supported in part by the Swiss National Science Foundation's Postdoc.Mobility fellowship (Grant No. P500-2_239026). The authors declare no competing interests.

References

[1] EU. Regulation - 2016/679 - EN - gdpr - EUR-Lex. https://eur-lex.europa.eu/eli/reg/2016/679/oj, 2024.

[2] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, April 2017.

[3] Sixing Yu, Juan Pablo Munoz, and Ali Jannesari. Federated Foundation Models: Privacy-Preserving and Collaborative Learning for Large Models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7174–7184. ELRA and ICCL, May 2024.

[4] Herbert Woisetschläger, Alexander Isenko, Shiqiang Wang, Ruben Mayer, and Hans-Arno Jacobsen. A Survey on Efficient Federated Learning Methods for Foundation Model Training. *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24) - Survey Track*, 2024.

[5] Yujun Cheng, Weiting Zhang, Zhewei Zhang, Chuan Zhang, Shengjin Wang, and Shiwen Mao. Toward Federated Large Language Models: Motivations, Methods, and Future Directions. *IEEE Communications Surveys & Tutorials*, 27(4):2733–2764, August 2025. ISSN 1553-877X. doi: 10.1109/COMST.2024.3503680.

[6] Dario Fenoglio, Martin Gjoreski, Edoardo Gabrielli, and Marc Langheinrich. Security and Privacy in Federated Learning: A Survey and Unified Taxonomy of Threat Models, Attacks, and Defenses, May 2026.

[7] Zhuohang Li, Jiaxin Zhang, Luyang Liu, and Jian Liu. Auditing Privacy Defenses in Federated Learning via Generative Gradient Leakage. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10122–10132. IEEE, June 2022. ISBN 978-1-6654-6946-3. doi: 10.1109/CVPR52688.2022.00989.

[8] Kai Yue, Richeng Jin, Chau-Wai Wong, Dror Baron, and Huaiyu Dai. Gradient obfuscation gives a false sense of security in federated learning. In *Proceedings of the 32nd USENIX Conference on Security Symposium*, SEC '23, pages 6381–6398, August 2023.

[9] Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, and Xuyun Zhang. Source Inference Attacks in Federated Learning. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 1102–1107, December 2021. doi: 10.1109/ICDM51629.2021.00129.

[10] Li Bai, Haibo Hu, Qingqing Ye, Haoyang Li, Leixia Wang, and Jianliang Xu. Membership Inference Attacks and Defenses in Federated Learning: A Survey. *ACM Comput. Surv.*, 57(4): 89:1–89:35, December 2024. ISSN 0360-0300. doi: 10.1145/3704633.

[11] Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. Deep Models Under the GAN: Information Leakage from Collaborative Deep Learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS '17, pages 603–618, October 2017. ISBN 978-1-4503-4946-8. doi: 10.1145/3133956.3134012.

[12] Ligeng Zhu, Zhijian Liu, and Song Han. Deep Leakage from Gradients. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[13] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. iDLG: Improved Deep Leakage from Gradients, January 2020.

[14] Hongxu Yin, Arun Mallya, Arash Vahdat, Jose M. Alvarez, Jan Kautz, and Pavlo Molchanov. See Through Gradients: Image Batch Recovery via GradInversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16337–16346, 2021.

[15] Hanchi Ren, Jingjing Deng, and Xianghua Xie. GRNN: Generative Regression Neural Network—A Data Leakage Attack for Federated Learning. *ACM Trans. Intell. Syst. Technol.*, 13(4):65:1–65:24, May 2022. ISSN 2157-6904. doi: 10.1145/3510032.

[16] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership Inference Attacks Against Machine Learning Models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18, May 2017. doi: 10.1109/SP.2017.41.

[17] Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S. Yu, and Xuyun Zhang. Membership Inference Attacks on Machine Learning: A Survey. *ACM Comput. Surv.*, 54(11s): 235:1–235:37, September 2022. ISSN 0360-0300. doi: 10.1145/3523273.

[18] Valerie Chen, Valerio Pastro, and Mariana Raykova. Secure Computation for Machine Learning With SPDZ, January 2019. Workshop at 32nd Conference on Neural Information Processing Systems (NIPS 2018).

[19] Brandon Reagen, Woo-Seok Choi, Yeongil Ko, Vincent T. Lee, Hsien-Hsin S. Lee, Gu-Yeon Wei, and David Brooks. Cheetah: Optimizing and Accelerating Homomorphic Encryption for Private Inference. In *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, pages 26–39, February 2021. doi: 10.1109/HPCA51647.2021.00013.

[20] Vaikkunth Mugunthan, Antigoni Polychroniadou, David Byrd, and Tucker Hybinette Balch. SMPAI: Secure Multi-Party Computation for Federated Learning. In *Proceedings of the NeurIPS 2019 Workshop on Robust AI in Financial Services*, volume 21, 2019.

[21] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical Secure Aggregation for Privacy-Preserving Machine Learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS '17, pages 1175–1191, October 2017.

[22] Le Trieu Phong, Yoshinori Aono, Takuya Hayashi, Lihua Wang, and Shiho Moriai. Privacy-Preserving Deep Learning via Additively Homomorphic Encryption. *IEEE Transactions on Information Forensics and Security*, 13(5):1333–1345, May 2018. ISSN 1556-6021.

[23] Haokun Fang and Quan Qian. Privacy Preserving Machine Learning with Homomorphic Encryption and Federated Learning. *Future Internet*, 13(4):94, April 2021. ISSN 1999-5903. doi: 10.3390/fi13040094.

[24] Peter Kairouz, Ziyu Liu, and Thomas Steinke. The Distributed Discrete Gaussian Mechanism for Federated Learning with Secure Aggregation. In *Proceedings of the 38th International Conference on Machine Learning*, pages 5201–5212. PMLR, July 2021.

[25] Lingchen Zhao, Jianlin Jiang, Bo Feng, Qian Wang, Chao Shen, and Qi Li. SEAR: Secure and Efficient Aggregation for Byzantine-Robust Federated Learning. *IEEE Transactions on Dependable and Secure Computing*, 19(5):3329–3342, September 2022. ISSN 1941-0018. doi: 10.1109/TDSC.2021.3093711.

[26] Abbas Yazdinejad, Ali Dehghantanha, and Gautam Srivastava. AP2FL: Auditable Privacy-Preserving Federated Learning Framework for Electronics in Healthcare. *IEEE Transactions on Consumer Electronics*, 70(1):2527–2535, February 2024. ISSN 1558-4127. doi: 10.1109/TCE.2023.3318509.

[27] Hanieh Hashemi, Yongqin Wang, Chuan Guo, and Murali Annavaram. Byzantine-Robust and Privacy-Preserving Framework for FedML, May 2021.

[28] Changshi Zhou, Nirwan Ansari, et al. Securing Federated Learning Enabled NWDAF Architecture With Partial Homomorphic Encryption. *IEEE Networking Letters*, 5(4):299–303, December 2023. ISSN 2576-3156. doi: 10.1109/LNET.2023.3294497.

[29] Robin C. Geyer, Tassilo Klein, and Moin Nabi. Differentially Private Federated Learning: A Client Level Perspective, March 2018.

[30] Yuanyuan Xie, Bing Chen, Jiale Zhang, and Di Wu. Defending against Membership Inference Attacks in Federated learning via Adversarial Example. In *2021 17th International Conference on Mobility, Sensing and Networking (MSN)*, pages 153–160, December 2021. doi: 10.1109/MSN53354.2021.00036.

[31] Joceline Ziegler, Bjarne Pfitzner, Heinrich Schulz, Axel Saalbach, and Bert Arnrich. Defending against Reconstruction Attacks through Differentially Private Federated Learning for Classification of Heterogeneous Chest X-ray Data. *Sensors*, 22(14):5195, January 2022. ISSN 1424-8220. doi: 10.3390/s22145195.

[32] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep Learning with Differential Privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, pages 308–318, October 2016. ISBN 978-1-4503-4139-4. doi: 10.1145/2976749.2978318.

[33] Naman Agarwal, Ananda Theertha Suresh, Felix Xinnan X Yu, Sanjiv Kumar, and Brendan McMahan. cpSGD: Communication-efficient and differentially-private distributed SGD. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[34] Huixuan Zong, Qing Wang, Xiaofeng Liu, Yinchuan Li, and Yunfeng Shao. Communication Reducing Quantization for Federated Learning with Local Differential Privacy Mechanism. In *2021 IEEE/CIC International Conference on Communications in China (ICCC)*, pages 75–80, July 2021. doi: 10.1109/ICCC52777.2021.9580315.

[35] Lichao Sun, Jianwei Qian, and Xun Chen. LDP-FL: Practical Private Aggregation in Federated Learning with Local Differential Privacy. In *Twenty-Ninth International Joint Conference on Artificial Intelligence*, volume 2, pages 1571–1578, 2021. doi: 10.24963/ijcai.2021/217.

[36] Jiahao Ding, Guannan Liang, Jinbo Bi, and Miao Pan. Differentially Private and Communication Efficient Collaborative Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(8):7219–7227, May 2021. ISSN 2374-3468. doi: 10.1609/aaai.v35i8.16887.

[37] Antonious Girgis, Deepesh Data, Suhas Diggavi, Peter Kairouz, and Ananda Theertha Suresh. Shuffled Model of Differential Privacy in Federated Learning. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, pages 2521–2529. PMLR, March 2021.

[38] Meng Shen, Jin Meng, Ke Xu, Shui Yu, and Liehuang Zhu. MemDefense: Defending against Membership Inference Attacks in IoT-based Federated Learning via Pruning Perturbations. *IEEE Transactions on Big Data*, pages 1–13, 2024. ISSN 2332-7790. doi: 10.1109/TBDATA.2024.3403388.

[39] Zhiqiu Zhang, Zhu Tianqing, Wei Ren, Ping Xiong, and Kim-Kwang Raymond Choo. Preserving data privacy in federated learning through large gradient pruning. *Computers & Security*, 125:103039, February 2023. ISSN 0167-4048. doi: 10.1016/j.cose.2022.103039.

[40] Sameer Bibikar, Haris Vikalo, Zhangyang Wang, and Xiaohan Chen. Federated Dynamic Sparse Training: Computing Less, Communicating Less, Yet Learning Better. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(6):6080–6088, June 2022. ISSN 2374-3468, 2159-5399. doi: 10.1609/aaai.v36i6.20555.

[41] Tianyue Chu, Mengwei Yang, Nikolaos Laoutaris, and Athina Markopoulou. PriPrune: Quantifying and Preserving Privacy in Pruned Federated Learning. *ACM Trans. Model. Perform. Eval. Comput. Syst.*, November 2024. ISSN 2376-3639. doi: 10.1145/3702241.

[42] Dan Alistarh, Torsten Hoefler, Mikael Johansson, Nikola Konstantinov, Sarit Khirirat, and Cedric Renggli. The Convergence of Sparsified Gradient Methods. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[43] Jingwei Sun, Ang Li, Binghui Wang, Huanrui Yang, Hai Li, and Yiran Chen. Soteria: Provable Defense against Privacy Leakage in Federated Learning from Representation Perspective. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9307–9315. IEEE, June 2021. ISBN 978-1-66544-509-2. doi: 10.1109/CVPR46437.2021.00919.

[44] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloé Kiddon, Jakub Konecný, Stefano Mazzocchi, H Brendan McMahan, Timon Van Overveldt, David Petrou, Daniel Ramage, and Jason Roselander. Towards Federated Learning at Scale: System Design. *Proceedings of the 2nd SysML Conference*, 2019.

[45] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American*, pages 4171–4186. Association for Computational Linguistics, June 2019. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423/.

[46] OpenAI. GPT-4 Technical Report, March 2023.

[47] Aurélien Bellet, Anne-Marie Kermarrec, and Erick Lavoie. D-cliques: Compensating for data heterogeneity with topology in decentralized federated learning. In *2022 41st International Symposium on Reliable Distributed Systems (SRDS)*, pages 1–11. IEEE, 2022.

[48] Batiste Le Bars, Aurélien Bellet, Marc Tommasi, Erick Lavoie, and Anne-Marie Kermarrec. Refined convergence and topology learning for decentralized sgd with heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, pages 1672–1702. PMLR, 2023.

[49] Shivam Kalra, Junfeng Wen, Jesse C. Cresswell, Maksims Volkovs, and H. R. Tizhoosh. Decentralized federated learning through proxy model sharing. *Nature Communications*, 14 (1):2899, May 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-38569-4.

[50] Qian Chen, Zilong Wang, Hongbo Wang, and Xiaodong Lin. FedDual: Pair-Wise Gossip Helps Federated Learning in Large Decentralized Networks. *IEEE Trans. Inf. Forensics Secur.*, 18:335–350, 2023. doi: 10.1109/TIFS.2022.3222935.

[51] Pijika Watcharapichat, Victoria Lopez Morales, Raul Castro Fernandez, and Peter Pietzuch. Ako: Decentralised Deep Learning with Partial Gradient Exchange. In *Proceedings of the Seventh ACM Symposium on Cloud Computing*, SoCC '16, pages 84–97, October 2016. ISBN 978-1-4503-4525-5. doi: 10.1145/2987550.2987586.

[52] Chenghao Hu, Jingyan Jiang, and Zhi Wang. Decentralized Federated Learning: A Segmented Gossip Approach, August 2019. In 1st International Workshop on Federated Machine Learning for User Privacy and Data Confidentiality (FML'19).

[53] Yuang Jiang, Shiqiang Wang, Víctor Valls, Bong Jun Ko, Wei-Han Lee, Kin K. Leung, and Leandros Tassiulas. Model Pruning Enables Efficient Federated Learning on Edge Devices. *IEEE Transactions on Neural Networks and Learning Systems*, 34(12):10374–10386, December 2023. ISSN 2162-2388. doi: 10.1109/TNNLS.2022.3166101.

[54] Farzin Haddadpour, Mohammad Mahdi Kamani, Aryan Mokhtari, and Mehrdad Mahdavi. Federated Learning with Compression: Unified Analysis and Sharp Guarantees. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, pages 2350–2358. PMLR, March 2021.

[55] Wei Liu, Li Chen, and Wenyi Zhang. Decentralized Federated Learning: Balancing Communication and Computing Costs. *IEEE Transactions on Signal and Information Processing over Networks*, 8:131–143, 2022. ISSN 2373-776X. doi: 10.1109/TSIPN.2022.3151242.

[56] Haoyu Zhao, Boyue Li, Zhize Li, Peter Richtarik, and Yuejie Chi. BEER: Fast $O(1/T)$ Rate for Decentralized Nonconvex Optimization with Communication Compression. *Advances in Neural Information Processing Systems*, 35:31653–31667, December 2022.

[57] Arian Raje, Baris Askin, Divyansh Jhunjhunwala, and Gauri Joshi. Ravan: Multi-Head Low-Rank Adaptation for Federated Fine-Tuning, June 2025.

[58] Youbang Sun, Zitao Li, Yaliang Li, and Bolin Ding. Improving LoRA in Privacy-preserving Federated Learning. *International Conference on Representation Learning*, 2024:17978–17994, May 2024.

[59] Zhize Li, Dmitry Kovalev, Xun Qian, and Peter Richtarik. Acceleration for Compressed Gradient Descent in Distributed and Federated Optimization. In *Proceedings of the 37th International Conference on Machine Learning*, pages 5895–5904. PMLR, November 2020.

[60] Zhize Li, Haoyu Zhao, Boyue Li, and Yuejie Chi. SoteriaFL: A Unified Framework for Private Federated Learning with Communication Compression. *Advances in Neural Information Processing Systems*, 35:4285–4300, December 2022.

[61] Liwei Zhang, Linghui Li, Xiaoyong Li, Binsi Cai, Yali Gao, Ruobin Dou, and Luying Chen. Efficient Membership Inference Attacks against Federated Learning via Bias Differences. In *Proceedings of the 26th International Symposium on Research in Attacks, Intrusions and Defenses*, RAID '23, pages 222–235. Association for Computing Machinery, October 2023. ISBN 9798400707650. doi: 10.1145/3607199.3607204.

[62] Xinlong He, Yang Xu, Sicong Zhang, Weida Xu, and Jiale Yan. Enhance membership inference attacks in federated learning. *Computers & Security*, 136:103535, January 2024. ISSN 0167-4048. doi: 10.1016/j.cose.2023.103535.

[63] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Exploiting Unintended Feature Leakage in Collaborative Learning. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 691–706, May 2019. doi: 10.1109/SP.2019.00029.

[64] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting Gradients - How easy is it to break privacy in federated learning? In *Advances in Neural Information Processing Systems*, volume 33, pages 16937–16947. Curran Associates, Inc., 2020.

[65] Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. The Secret Revealer: Generative Model-Inversion Attacks Against Deep Neural Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 253–261, 2020.

[66] Dimitar Iliev Dimitrov, Mislav Balunovic, Nikola Konstantinov, and Martin Vechev. Data Leakage in Federated Averaging. *Transactions on Machine Learning Research*, June 2022. ISSN 2835-8856.

[67] Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive Privacy Analysis of Deep Learning: Passive and Active White-box Inference Attacks against Centralized and Federated Learning. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 739–753, May 2019. doi: 10.1109/SP.2019.00065.

[68] Oualid Zari, Chuan Xu, and Giovanni Neglia. Efficient passive membership inference attack in federated learning, October 2021. At NeurIPS 2021 Workshop Privacy in Machine Learning.

[69] Jiacheng Li, Ninghui Li, and Bruno Ribeiro. Effective passive membership inference attacks in federated learning against overparameterized models. In *The Eleventh International Conference on Learning Representations*, September 2022.

[70] Yinbin Miao, Rongpeng Xie, Xinghua Li, Ximeng Liu, Zhuo Ma, and Robert H. Deng. Compressed Federated Learning Based on Adaptive Local Differential Privacy. In *Proceedings of the 38th Annual Computer Security Applications Conference*, ACSAC '22, pages 159–170. Association for Computing Machinery, December 2022. ISBN 978-1-4503-9759-9. doi: 10.1145/3564625.3567973.

[71] Mengmeng Yang, Taolin Guo, Tianqing Zhu, Ivan Tjuawinata, Jun Zhao, and Kwok-Yan Lam. Local differential privacy and its applications: A comprehensive survey. *Computer Standards & Interfaces*, 89:103827, April 2024. ISSN 0920-5489. doi: 10.1016/j.csi.2023.103827.

[72] Mohammed Adnan, Shivam Kalra, Jesse C. Cresswell, Graham W. Taylor, and Hamid R. Tizhoosh. Federated learning and differential privacy for medical image analysis. *Scientific Reports*, 12(1):1953, February 2022. ISSN 2045-2322. doi: 10.1038/s41598-022-05539-7.

[73] Bargav Jayaraman, Lingxiao Wang, David Evans, and Quanquan Gu. Distributed Learning without Distress: Privacy-Preserving Empirical Risk Minimization. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[74] Stacey Truex, Ling Liu, Ka-Ho Chow, Mehmet Emre Gursoy, and Wenqi Wei. LDP-Fed: Federated learning with local differential privacy. In *Proceedings of the Third ACM International Workshop on Edge Systems, Analytics and Networking*, EdgeSys '20, pages 61–66, May 2020. ISBN 978-1-4503-7132-2. doi: 10.1145/3378679.3394533.

[75] Richeng Jin, Zhonggen Su, Caijun Zhong, Zhaoyang Zhang, Tony Quek, and Huaiyu Dai. Breaking the Communication-Privacy-Accuracy Tradeoff with f-Differential Privacy. *Advances in Neural Information Processing Systems*, 36:74103–74126, December 2023.

[76] Xuechen Li, Florian Tramèr, Percy Liang, and Tatsunori Hashimoto. Large language models can be strong differentially private learners. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=bVuP3ltATMz.

[77] Ang Li, Jingwei Sun, Binghui Wang, Lin Duan, Sicheng Li, Yiran Chen, and Hai Li. LotteryFL: Empower Edge Intelligence with Personalized and Communication-Efficient Federated Learning. In *2021 IEEE/ACM Symposium on Edge Computing (SEC)*, pages 68–79, December 2021. doi: 10.1145/3453142.3492909.

[78] Sayan Biswas, Mathieu Even, Anne-Marie Kermarrec, Laurent Massoulié, Rafael Pires, Rishi Sharma, and Martijn De Vos. Noiseless Privacy-Preserving Decentralized Learning. *Proceedings on Privacy Enhancing Technologies*, 2025(1):824–844, January 2025. ISSN 2299-0984. doi: 10.56553/popets-2025-0043.

[79] Abhijit Guha Roy, Shayan Siddiqui, Sebastian Pölsterl, Nassir Navab, and Christian Wachinger. BrainTorrent: A Peer-to-Peer Environment for Decentralized Federated Learning, May 2019.

[80] Christodoulos Pappas, Dimitris Chatzopoulos, Spyros Lalis, and Manolis Vavalis. IPLS: A Framework for Decentralized Federated Learning. In *2021 IFIP Networking Conference (IFIP Networking)*, pages 1–6, June 2021. doi: 10.23919/IFIPNetworking52078.2021.9472790.

[81] Marco Bornstein, Tahseen Rabbani, Evan Z. Wang, Amrit Bedi, and Furong Huang. SWIFT: Rapid Decentralized Federated Learning via Wait-Free Model Communication. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=jh1nCir1R3d.

[82] D. Kempe, A. Dobra, and J. Gehrke. Gossip-based computation of aggregate information. In *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.*, pages 482–491, October 2003. doi: 10.1109/SFCS.2003.1238221.

[83] Shahryar Zehtabi, Dong-Jun Han, Rohit Parasnis, Seyyedali Hosseinalipour, and Christopher Brinton. Decentralized Sporadic Federated Learning: A Unified Algorithmic Framework with Convergence Guarantees. In *The Thirteenth International Conference on Learning Representations*, October 2024.

[84] Yifan Shi, Li Shen, Kang Wei, Yan Sun, Bo Yuan, Xueqian Wang, and Dacheng Tao. Improving the Model Consistency of Decentralized Federated Learning. In *Proceedings of the 40th International Conference on Machine Learning*, pages 31269–31291. PMLR, July 2023.

[85] Edoardo Gabrielli, Anthony Di Pietro, Dario Fenoglio, Giovanni Pica, and Gabriele Tolomei. A Survey on Decentralized Federated Learning, March 2026.

[86] Zhe Qu, Xingyu Li, Jie Xu, Bo Tang, Zhuo Lu, and Yao Liu. On the Convergence of Multi-Server Federated Learning With Overlapping Area. *IEEE Transactions on Mobile Computing*, 22(11):6647–6662, November 2023. ISSN 1536-1233. doi: 10.1109/TMC.2022.3200016.

[87] Bin Wang, Jun Fang, Hongbin Li, Xiaojun Yuan, and Qing Ling. Confederated Learning: Federated Learning With Decentralized Edge Servers. *IEEE Transactions on Signal Processing*, 71:248–263, 2023. ISSN 1941-0476. doi: 10.1109/TSP.2023.3241768.

[88] Mayank Rathee, Conghao Shen, Sameer Wagh, and Raluca Ada Popa. ELSA: Secure Aggregation for Federated Learning with Malicious Actors. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 1961–1979, May 2023. doi: 10.1109/SP46215.2023.10179468.

[89] Sayan Biswas, Anne-Marie Kermarrec, Alexis Marouani, Rafael Pires, Rishi Sharma, and Martijn de Vos. Boosting Asynchronous Decentralized Learning with Model Fragmentation. In *Proceedings of the ACM on Web Conference 2025*, WWW '25, pages 685–696, New York, NY, USA, April 2025. Association for Computing Machinery. ISBN 979-8-4007-1274-6. doi: 10.1145/3696410.3714872.

[90] Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. ZeRO: Memory optimizations Toward Training Trillion Parameter Models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16, November 2020. doi: 10.1109/SC41405.2020.00024.

[91] Nicolò Michelusi, Gesualdo Scutari, and Chang-Shen Lee. Finite-Bit Quantization for Distributed Algorithms With Linear Convergence. *IEEE Transactions on Information Theory*, 68 (11):7254–7280, November 2022. ISSN 1557-9654. doi: 10.1109/TIT.2022.3176253.

[92] Zhize Li and Peter Richtarik. CANITA: Faster Rates for Distributed Convex Optimization with Communication Compression. In *Advances in Neural Information Processing Systems*, volume 34, pages 13770–13781. Curran Associates, Inc., 2021.

[93] Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. QSGD: Communication-Efficient SGD via Gradient Quantization and Encoding. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[94] Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian Stich, and Martin Jaggi. Error Feedback Fixes SignSGD and other Gradient Compression Schemes. In *Proceedings of the 36th International Conference on Machine Learning*, pages 3252–3261. PMLR, May 2019.

[95] Amirhossein Reisizadeh, Aryan Mokhtari, Hamed Hassani, Ali Jadbabaie, and Ramtin Pedarsani. FedPAQ: A Communication-Efficient Federated Learning Method with Periodic Averaging and Quantization. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, pages 2021–2031. PMLR, June 2020.

[96] Eduard Gorbunov, Konstantin P. Burlachenko, Zhize Li, and Peter Richtarik. MARINA: Faster Non-Convex Distributed Learning with Compression. In *Proceedings of the 38th International Conference on Machine Learning*, pages 3788–3798. PMLR, July 2021.

[97] Konstantin Mishchenko, Eduard Gorbunov, Martin Takáč, and Peter Richtárik. Distributed Learning with Compressed Gradient Differences, December 2023.

[98] Sarit Khirirat, Hamid Reza Feyzmahdavian, and Mikael Johansson. Distributed learning with compressed gradients, November 2018.

[99] Anastasia Koloskova, Sebastian Stich, and Martin Jaggi. Decentralized Stochastic Optimization and Gossip Algorithms with Compressed Communication. In *Proceedings of the 36th International Conference on Machine Learning*, pages 3478–3487. PMLR, May 2019.

[100] Peter Richtarik, Igor Sokolov, Elnur Gasanov, Ilyas Fatkhullin, Zhize Li, and Eduard Gorbunov. 3PC: Three Point Compressors for Communication-Efficient Distributed Training and a Better Theory for Lazy Aggregation. In *Proceedings of the 39th International Conference on Machine Learning*, pages 18596–18648. PMLR, June 2022.

[101] Hongyi Wang, Scott Sievert, Shengchao Liu, Zachary Charles, Dimitris Papailiopoulos, and Stephen Wright. ATOMO: Communication-efficient Learning via Atomic Sparsification. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[102] Nikita Ivkin, Daniel Rothchild, Enayat Ullah, Vladimir braverman, Ion Stoica, and Raman Arora. Communication-efficient Distributed SGD with Sketching. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[103] Márk Jelasity, Spyros Voulgaris, Rachid Guerraoui, Anne-Marie Kermarrec, and Maarten van Steen. Gossip-based peer sampling. *ACM Transactions on Computer Systems (TOCS)*, 25(3): 8–es, August 2007. ISSN 0734-2071. doi: 10.1145/1275517.1275520.

[104] Yossi Gilad, Rotem Hemo, Silvio Micali, Georgios Vlachos, and Nickolai Zeldovich. Algorand: Scaling Byzantine Agreements for Cryptocurrencies. In *Proceedings of the 26th Symposium on Operating Systems Principles*, SOSP '17, pages 51–68. Association for Computing Machinery, October 2017. ISBN 978-1-4503-5085-3. doi: 10.1145/3132747.3132757.

[105] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[106] Rie Johnson and Tong Zhang. Accelerating Stochastic Gradient Descent using Predictive Variance Reduction. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

[107] Andrew Lowy, Ali Ghafelebashi, and Meisam Razaviyayn. Private Non-Convex Federated Learning Without a Trusted Server. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, pages 5749–5786. PMLR, April 2023.

[108] Yangsibo Huang, Samyak Gupta, Zhao Song, Kai Li, and Sanjeev Arora. Evaluating Gradient Inversion Attacks and Defenses in Federated Learning. In *Advances in Neural Information Processing Systems*, volume 34, pages 7232–7241. Curran Associates, Inc., 2021.

[109] Samyak Gupta, Yangsibo Huang, Zexuan Zhong, Tianyu Gao, Kai Li, and Danqi Chen. Recovering Private Text in Federated Learning of Language Models. *Advances in Neural Information Processing Systems*, 35:8130–8143, December 2022.

[110] Caridad Arroyo Arevalo, Sayedeh Leila Noorbakhsh, Yun Dong, Yuan Hong, and Binghui Wang. Task-Agnostic Privacy-Preserving Representation Learning for Federated Learning against Attribute Inference Attacks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(10):10909–10917, March 2024. ISSN 2374-3468. doi: 10.1609/aaai.v38i10.28965.

[111] Yann LeCun, Corinna Cortes, et al. The mnist database of handwritten digits. 2005.

[112] Alex Krizhevsky, Geoffrey Hinton, et al. Learning Multiple Layers of Features from Tiny Images. 2009.

[113] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning Word Vectors for Sentiment Analysis. In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea, editors, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, June 2011.

[114] Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, July 2017. doi: 10.18653/v1/P17-1099. URL `https://www.aclweb.org/anthology/P17-1099`.

[115] Gary B. Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled Faces in the Wild: A Database forStudying Face Recognition in Unconstrained Environments. In *Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition*, 2008.

[116] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, June 2009. doi: 10.1109/CVPR.2009.5206848.

[117] Sid Black, Gao Leo, Phil Wang, Connor Leahy, and Stella Biderman. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow, March 2021. URL `https://doi.org/10.5281/zenodo.5297715`.

[118] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108, 2019.

[119] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778. IEEE, June 2016. ISBN 978-1-4673-8851-1. doi: 10.1109/CVPR.2016.90.

[120] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, November 1998. ISSN 1558-2256. doi: 10.1109/5.726791. URL https://ieeexplore.ieee.org/abstract/document/726791. Conference Name: Proceedings of the IEEE.

[121] Thomas Steinke, Milad Nasr, and Matthew Jagielski. Privacy Auditing with One (1) Training Run. *Advances in Neural Information Processing Systems*, 36:49268–49280, December 2023.

[122] Wenjie Fu, Huandong Wang, Chen Gao, Guanghua Liu, Yong Li, and Tao Jiang. Membership Inference Attacks against Fine-tuned Large Language Models via Self-prompt Calibration. *Advances in Neural Information Processing Systems*, 37:134981–135010, December 2024.

[123] Sashank J. Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and Hugh Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=LkFG3lB13U5.

[124] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33:7611–7623, 2020.

[125] Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[126] Dario Fenoglio, Gabriele Dominici, Pietro Barbiero, Alberto Tonda, Martin Gjoreski, and Marc Langheinrich. Federated Behavioural Planes: Explaining the Evolution of Client Behaviour in Federated Learning. *Advances in Neural Information Processing Systems*, 37:112777–112813, 2024.

[127] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the Convergence of FedAvg on Non-IID Data. In *International Conference on Learning Representations*, September 2019.

[128] Xiaoxiao Li, Zhao Song, Runzhou Tao, and Guangyi Zhang. A Convergence Theory for Federated Average: Beyond Smoothness. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 1292–1297, December 2022. doi: 10.1109/BigData55660.2022.10020426.

[129] Lingxiao Wang, Bargav Jayaraman, David Evans, and Quanquan Gu. Efficient privacy-preserving stochastic nonconvex optimization. In *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, volume 216 of *UAI '23*, pages 2203–2213. JMLR.org, July 2023.

[130] Xin Zhang, Minghong Fang, Jia Liu, and Zhengyuan Zhu. Private and communication-efficient edge learning: A sparse differential gaussian-masking distributed SGD approach. In *Proceedings of the Twenty-First International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*, Mobihoc '20, pages 261–270. Association for Computing Machinery, October 2020. ISBN 978-1-4503-8015-7.

[131] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, January 2017.

[132] Herbert Robbins and Sutton Monro. A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, 22(3):400–407, September 1951. ISSN 0003-4851, 2168-8990. doi: 10.1214/aoms/1177729586.

[133] Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, Graham Cormode, and Ilya Mironov. Opacus: User-friendly differential privacy library in PyTorch. *arXiv preprint arXiv:2109.12298*, 2021.

[134] Chi Zhang, Xiaoman Zhang, Ekanut Sotthiwat, Yanyu Xu, Ping Liu, Liangli Zhen, and Yong Liu. Generative Gradient Inversion via Over-Parameterized Networks in Federated Learning. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5103–5112. IEEE, October 2023. ISBN 9798350307184. doi: 10.1109/ICCV51070.2023.00473.

[135] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in Neural Information Processing Systems*, 32, 2019.

[136] Daniel J. Beutel, Taner Topal, Akhil Mathur, Xinchi Qiu, Javier Fernandez-Marques, Yan Gao, Lorenzo Sani, Kwing Hei Li, Titouan Parcollet, Pedro Porto Buarque de Gusmão, and Nicholas D. Lane. Flower: A Friendly Federated Learning Research Framework, March 2022.

[137] J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi: 10.1109/MCSE.2007.55.

[138] Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 56 – 61, 2010. doi: 10.25080/Majora-92bf1922-00a.

[139] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 586–595. IEEE, June 2018. ISBN 978-1-5386-6420-9. doi: 10.1109/CVPR.2018.00068.

[140] Dario Fenoglio, Mohan Li, Pietro Barbiero, Nicholas D. Lane, Marc Langheinrich, and Martin Gjoreski. FLUX: Efficient Descriptor-Driven Clustered Federated Learning under Arbitrary Distribution Shifts. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, October 2025.

Table of Contents

A Assumptions **22**

B Convergence Equivalence of FSA **22**

C Convergence and Utility for ERIS with DSC **23**
C.1 Proof of Theorem 3.2 23
C.2 Proof of Lemma C.1 26
C.3 Proof of Lemma C.2 27
C.4 Proof of Lemma C.3 29
C.5 Utility Comparison 31

D Privacy Guarantees for ERIS with DSC **32**
D.1 Proof of Theorem 3.3 32
D.2 Privacy Under Colluding Aggregators 34

E Experimental Setup **36**
E.1 Models and Hyperparameter Settings 36
E.2 Implementation Details of Privacy Attacks 36
E.3 Licenses and Hardware 37

F Additional Experiments and Analysis **37**
F.1 Empirical Validation and Operational Estimation of $C_{\max}$ 37
F.2 Scalability and Efficiency of ERIS 38
F.3 Effect of Distributed Shifted Compression on Model Utility 42
F.4 Effect of Local Data Size on the Utility–Privacy Trade-off 42
F.5 Robustness to Aggregator and Link Failures 43
F.6 Data Reconstruction Attacks 45
F.7 Balancing Utility and Privacy - IID Setting 46
F.8 Balancing Utility and Privacy - non-IID Setting 46
F.9 Balancing Utility and Privacy - Biased Gradient Estimator 47
F.10 Pareto Analysis under Varying Privacy Constraints 51
F.11 Temporal Evolution of Privacy Leakage and Utility 52

Appendix

This appendix is organised as follows. Section A restates the main assumptions used in our theoretical analysis. Section B proves that FSA maintains the same convergence behavior as centralized FedAvg in the absence of compression. Section C presents the convergence and communication analysis for ERIS with DSC. Section D provides the privacy guarantees and includes an extension of Theorem 3.3 to colluding aggregators. Section E details our experimental setup, including models, hyperparameters, privacy attacks, datasets, licenses, and hardware for full reproducibility. Finally, Sections F.1–F.11 report additional experimental results supporting our claims, including evaluations of scalability, compression, data reconstruction attacks, privacy–utility trade-offs under varying heterogeneity conditions (IID and non-IID) and both biased and unbiased gradient estimators, and the temporal evolution of privacy leakage throughout training.

A Assumptions

For clarity and completeness of the Appendix, we restate the core assumptions used in the main theorems—Theorem 3.2 and Theorem 3.3. These include smoothness and unbiased local estimator conditions commonly adopted in the FL literature.

Assumption A.1 (Smoothness)**.** There exists some $L \geq 0$, such that for all local functions $f_{i,j}$ (indexed by $i \in [n]$ and $j \in [m]$), we have

$$\|\nabla f_{i,j}(x_1) - \nabla f_{i,j}(x_2)\| \;\leq\; L\,\|x_1 - x_2\|, \quad \forall\, x_1, x_2 \in \mathbb{R}^d, \tag{8}$$

or equivalently expressed with the following general bound:

$$f_{i,j}(x_1) \leq f_{i,j}(x_2) + \langle \nabla f_{i,j}(x_2), x_1 - x_2 \rangle + \frac{L}{2}\|x_1 - x_2\|^2. \tag{9}$$

Assumption A.2 (Unbiased local estimator)**.** The gradient estimator $\tilde{\mathbf{g}}_k^t$ is unbiased $\mathbb{E}_t[\tilde{\mathbf{g}}_k^t] = \nabla f_k(\mathbf{x^t})$ for $k \in \mathcal{K}$, where $\mathbb{E}_t$ takes the expectation conditioned on all history before round t. Moreover, there exist constants Γ_1, Γ_2, Γ_3, Γ_4, and θ such that:

$$\mathbb{E}_t\big[\frac{1}{K}\sum_{k=1}^{K} ||\tilde{\mathbf{g}}_k^t - \nabla f_k(\mathbf{x^t})||^2\big] \leq \Gamma_1 \Delta^t + \Gamma_2 \tag{11a}$$

$$\mathbb{E}_t[\Delta^{t+1}] \leq (1-\theta)\Delta^t + \Gamma_3 ||\nabla f(\mathbf{x}^t)||^2 + \Gamma_4 \mathbb{E}_t[||\mathbf{x}^{t+1} - \mathbf{x}^t||^2] \tag{11b}$$

Remark A.3. The parameters Γ_1 and Γ_2 capture the variance of the gradient estimators, e.g., $\Gamma_1 = \Gamma_2 = 0$ if the client computes local full gradient $\tilde{\mathbf{g}}_i^t = \nabla f_i(\mathbf{x^t})$, and $\Gamma_1 \neq 0$ and $\Gamma_2 = 0$ if the client uses variance-reduced gradient estimators such as SVRG/SAGA.

B Convergence Equivalence of FSA

This section shows that Federated Shard Aggregation (FSA)—Algorithm 1 instantiated without DSC, i.e., with the identity compressor $\mathcal{C}_k^t = \mathrm{Id}$ and reference vectors fixed to zero—produces exactly the same global iterate sequence as the standard single-server algorithm (e.g., *FedAvg*). Consequently, every convergence guarantee proved for FedAvg carries over verbatim. The proof is algebraic and does not rely on additional smoothness or convexity assumptions beyond those stated in Appendix A.

Notation. Recall that client k holds S_k data points and that $S := \sum_{k=1}^K S_k$. Let $\tilde{\mathbf{g}}_k^t$ denote the (possibly stochastic) gradient that client k transmits at communication round t and write $\tilde{\mathbf{g}}^t = \frac{1}{S}\sum_{k=1}^K S_k\, \tilde{\mathbf{g}}_k^t$ for the sample–weighted mean gradient.

Theorem B.1 (Convergence equivalence of FSA)**.** *Run Algorithm 1 with $A \geq 1$ aggregators, $\mathcal{C}_k^t = \mathrm{Id}$, and $\mathbf{s}_k^t = \mathbf{0}$ for all k, t. Let $\mathbf{x}^t$ with $t \geq 0$ be the resulting iterates and let $\tilde{\mathbf{x}}^t$ be the iterates obtained by FedAvg ($A = 1$) using the same initialization, learning rates λ_t, and client gradients $\tilde{\mathbf{g}}_k^t$. Then for every round $t \geq 0$*

$$\mathbf{x}^t = \tilde{\mathbf{x}}^t. \tag{11}$$

Hence all convergence bounds that hold for FedAvg under Assumptions A.1 and A.2 (with $\omega = 0$) apply unchanged to ERIS *with FSA only.*

Sketch. Partition each client gradient into A disjoint coordinate shards using the categorical masks $\{\mathbf{m}_{(a)}^t\}_{a=1}^A$ introduced in Section 3.2: $\tilde{\mathbf{g}}_{k,(a)}^t = \tilde{\mathbf{g}}_k^t \odot \mathbf{m}_{(a)}^t$. Because the masks are disjoint and sum to the all–ones vector, the original gradient decomposes exactly as $\tilde{\mathbf{g}}_k^t = \sum_{a=1}^A \tilde{\mathbf{g}}_{k,(a)}^t$. Aggregator a forms the weighted average of its shard

$$\bar{\mathbf{g}}_{(a)}^t \;:=\; \frac{1}{S}\sum_{k=1}^K S_k\, \tilde{\mathbf{g}}_{k,(a)}^t. \tag{12}$$

Summing over all aggregators and swapping summation order yields

$$\sum_{a=1}^A \bar{\mathbf{g}}_{(a)}^t = \frac{1}{S}\sum_{k=1}^K S_k \sum_{a=1}^A \tilde{\mathbf{g}}_{k,(a)}^t = \frac{1}{S}\sum_{k=1}^K S_k\, \tilde{\mathbf{g}}_k^t = \tilde{\mathbf{g}}^t. \tag{13}$$

FSA therefore updates the global model via $\mathbf{x}^{t+1} = \mathbf{x}^t - \lambda_t \sum_{a=1}^A \bar{\mathbf{g}}_{(a)}^t = \mathbf{x}^t - \lambda_t \tilde{\mathbf{g}}^t$, which is exactly the FedAvg rule. By induction on t the iterates coincide. □

Remark B.2. The identity above is purely algebraic, hence it remains valid when clients perform multiple local SGD steps, when the data are non IID, or when the global objective is nonconvex (e.g., [2, 127, 128]). The key insight is that splitting the gradient vector dimension-wise introduces no additional approximation error; the final aggregated gradient is mathematically identical to that obtained by a single server aggregating all client gradients in one place. This ensures that the convergence behavior of FSA matches that of traditional federated learning approaches, while its sole effect is to distribute network load.

C Convergence and Utility for ERIS with DSC

In this section, we present the proof of Theorem 3.2 (Utility of ERIS with DSC), modifying the general proof strategy of [60] to accommodate our decentralized setting, model partitioning, and the absence of differential privacy.

C.1 Proof of Theorem 3.2

Proof. Let $\mathbb{E}_t$ denote the expectation conditioned on the full history up to round t. By invoking Theorem B.1, we simplify the analysis by omitting model partitioning and treating $\mathbf{v}^t$ as the aggregated update. Thus, the update rule becomes $\mathbf{x}^{t+1} = \mathbf{x}^t - \lambda_t \mathbf{v}^t$. We now apply this rule within the smoothness inequality (9):

$$\mathbb{E}_t[f(\mathbf{x}^{t+1})] \leq \mathbb{E}_t\left[f(\mathbf{x}^t) - \lambda_t \langle \nabla f(\mathbf{x}^t), \mathbf{v}^t\rangle + \frac{\lambda^2 L}{2}\|\mathbf{v}^t\|^2\right] \tag{14}$$

First, to verify the unbiased nature of $\mathbf{v}^t$, we consider:

$$\begin{aligned}
\mathbb{E}_t[\mathbf{v}^t] &= \mathbb{E}_t\left[\mathbf{s}^t + \frac{1}{K}\sum_{k=1}^K \mathbf{v}_k^t\right] \\
&= \mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^K \mathbf{s}_k^t + \frac{1}{K}\sum_{k=1}^K \mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t)\right] \\
&\overset{(3)}{=} \mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^K \tilde{\mathbf{g}}_k^t\right] = \frac{1}{K}\sum_{k=1}^K \mathbb{E}_t[\tilde{\mathbf{g}}_k^t] \overset{(a)}{=} \frac{1}{K}\sum_{k=1}^K \nabla f_k(\mathbf{x}^t) = \nabla f(\mathbf{x}^t)
\end{aligned} \tag{15}$$

where (a) due to Assumption A.2, which states that each $\tilde{\mathbf{g}}_k^t$ is an unbiased estimator of $\nabla f_k(\mathbf{x}^t)$ (i.e., $\mathbb{E}_t[\tilde{\mathbf{g}}_k^t] = \nabla f_k(\mathbf{x}^t)$).

Substituting Equation (15) into (14), we obtain:

$$\mathbb{E}_t[f(\mathbf{x}^{t+1})] \leq \mathbb{E}_t\left[f(\mathbf{x}^t) - \lambda_t||\nabla f(\mathbf{x}^t)||^2 + \frac{\lambda^2 L}{2}\|\mathbf{v}^t\|^2\right] \tag{16}$$

We further bound the term $\mathbb{E}_t[||\mathbf{v}^t||^2]$ in Lemma C.1, whose proof is available in the Appendix C.2.

Lemma C.1. *Consider that $\mathbf{v}^t$ is constructed according to Algorithm 1, it holds that*

$$\mathbb{E}_t[\|\mathbf{v}^t\|^2] \leq \mathbb{E}_t\left[\frac{(1+\omega)}{K^2}\sum_{k=1}^{K}\|\tilde{\mathbf{g}}_k^t - \nabla f_k(\mathbf{x}^t)\|^2\right] + \frac{\omega}{K^2}\sum_{k=1}^{K}\|\nabla f_k(\mathbf{x}^t) - \mathbf{s}_k^t\|^2 + \|\nabla f(\mathbf{x}^t)\|^2. \quad (17)$$

To further our analysis, we now derive upper bounds for the first two terms on the right-hand side of Equation (17). The first term can be controlled using Equation (11a) from Assumption A.2, yielding the bound $\Gamma_1\Delta^t + \Gamma_2$. Next, we establish that the second term decreases over time, as formalized in the following lemma (proof available in Appendix C.3).

Lemma C.2. *Let Assumption A.1 hold, and let the shift $\mathbf{s}_k^{t+1}$ be updated according to Algorithm 1. Then, for $\gamma_t = \sqrt{\frac{1+2\omega}{2(1+\omega)^3}}$, we have:*

$$\begin{aligned}\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\|\nabla f_k(\mathbf{x}^{t+1}) - \mathbf{s}_k^{t+1}\|^2\right] \leq \mathbb{E}_t\Bigg[&\left(1 - \frac{1}{2(1+\omega)}\right)\frac{1}{K}\sum_{k=1}^{K}\|\nabla f_k(\mathbf{x}^t) - \mathbf{s}_k^t\|^2 \\ &+ \frac{1}{(1+\omega)K}\sum_{k=1}^{K}\|\tilde{\mathbf{g}}_k^t - \nabla f_k(\mathbf{x}^t)\|^2 \\ &+ 2(1+\omega)L^2\|\mathbf{x}^{t+1} - \mathbf{x}^t\|^2\Big]. \end{aligned} \quad (18)$$

For clarity, we introduce the notation: $\mathcal{S}^t := \frac{1}{K}\sum_{k=1}^{K}||\nabla f_k(\mathbf{x}^t) - \mathbf{s}_k^t||^2$. For some $\alpha \geq 0, \beta \geq 0$, we now define a potential function to analyze the convergence behavior:

$$\Phi_t := f(\mathbf{x}^t) - f^* + \alpha L\Delta^t + \frac{\beta}{L}\mathcal{S}^t, \quad (19)$$

Using Lemmas C.1 and C.2, we demonstrate in Lemma C.3 that this potential function decreases in expectation at each iteration (proof provided in Appendix C.4).

Lemma C.3. *Under Assumptions A.1 and A.2, if the learning rate is chosen as*

$$\lambda_t \triangleq \lambda \leq \min\left(\frac{1}{(1 + 2\alpha\Gamma_4 + 4\beta(1+\omega) + 2\alpha\Gamma_3/\lambda^2)L}, \frac{\sqrt{\beta K}}{\sqrt{1 + 2\alpha\Gamma_4 + 4\beta(1+\omega)}(1+\omega)L}\right), \quad (20)$$

where $\alpha = \frac{3\beta\Gamma_1}{2(1+\omega)\theta L^2}$ for any $\beta > 0$, and the shift step size γ_t is defined as in Lemma C.2, it follows that for every round $t \geq 0$, the expected potential function satisfies the following bound:

$$\mathbb{E}_t[\Phi_{t+1}] \leq \Phi_t - \frac{\lambda_t}{2}\|\nabla f(\mathbf{x}^t)\|^2 + \frac{3\beta\Gamma_2}{2(1+\omega)L}. \quad (21)$$

Remark C.4. Since the last term is generally a small constant during time (see Assumption A.2) and $\frac{\lambda_t}{2}\|\nabla f(\mathbf{x}^t)\|^2$ is positive, Equation (21) indicates that the potential decrease over the time.

With Lemma C.3 established, we now proceed to the proof of Theorem 3.2, which characterizes the utility and the number of communication rounds required for ERIS to reach a given accuracy level. We begin by summing Equation (21) from rounds $t = 0$ to $T - 1$:

$$\begin{aligned}\sum_{t=0}^{T-1}\mathbb{E}[\Phi_{t+1}] &\leq \sum_{t=0}^{T-1}\mathbb{E}[\Phi_t] - \sum_{t=0}^{T-1}\left(\frac{\lambda_t}{2}\|\nabla f(\mathbf{x}^t)\|^2 + \frac{3\beta\Gamma_2}{2(1+\omega)L}\right) \\ \mathbb{E}[\Phi_T] - \mathbb{E}[\Phi_0] &\leq -\sum_{t=0}^{T-1}\frac{\lambda_t}{2}\|\nabla f(\mathbf{x}^t)\|^2 + \frac{3\beta\Gamma_2 T}{2(1+\omega)L}\end{aligned}$$

Since by construction, we typically have $\mathbb{E}[\Phi_t] \geq 0$, by choosing the learning rate λ_t as in Lemma C.3, we finally obtain

$$\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla f(\mathbf{x}^t)\|^2 \leq \frac{2\Phi_0}{\lambda T} + \frac{3\beta\Gamma_2}{(1+\omega)L\lambda}, \tag{22}$$

which proves that per $T \to \infty$

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla f(\mathbf{x}^t)\|^2 \leq \frac{3\beta\Gamma_2}{(1+\omega)L\lambda}. \tag{23}$$

While to achieve a predefined utility level $\epsilon \geq \frac{1}{T}\sum_{t=0}^{T-1}||\nabla f(\mathbf{x}^t)||^2$, the total rounds T must satisfy:

$$T \;\geq\; \frac{2\,\Phi_0}{\lambda\left(\epsilon - \frac{3\beta\,\Gamma_2}{(1+\omega)\,L\,\lambda}\right)}. \tag{24}$$

If ϵ is strictly less than the residual $\frac{3\beta\,\Gamma_2}{(1+\omega)\,L\,\lambda}$, no finite T can achieve the utility ϵ in an average sense, therefore conditions on the adopted estimator need to be changed.

Remark C.5. ERIS with DSC utility is asymptotically governed by the variance in $\tilde{\mathbf{g}}$, which directly depends on the used estimator (e.g., $\Gamma_2 = 0$ with SVRG/SAGA) or on the dimension of the batch size (e.g., $\Gamma_2 = 0$ with local full gradients). Compared to SoteriaFL [60], the upper bound of ERIS with DSC utility does not have a component growing with T, limiting the convergence.

□

Corollary C.6 (Utility of ERIS–SGD with DSC)**.** *Consider the FL setting in* (1) *with K clients, where each client $k \in \mathcal{K}$ holds a local dataset $D_k = \{d_{k,s}\}_{s=1}^{S_k}$, and let Assumptions A.1 and A.2 hold. Assume that, at each round t, client k uses a mini-batch SGD estimator*

$$\tilde{\mathbf{g}}_k^t \;=\; \frac{1}{b_k}\sum_{s\in\mathcal{B}_k^t}\nabla f_{k,s}(\mathbf{x}^t),$$

where $\mathcal{B}_k^t \subseteq \{1,\ldots,S_k\}$ is a uniformly sampled mini-batch of size b_k, and that stochastic gradients are uniformly bounded as $\|\nabla f_{k,s}(\mathbf{x})\| \leq G$ for all $k, s, \mathbf{x}$.

For notational simplicity, suppose that all clients share the same dataset size and batch size, i.e., $S_k \equiv m$ and $b_k \equiv b$ for all k. Then the constants in Assumption A.2 are

$$\Gamma_1 = \Gamma_3 = \Gamma_4 = 0, \qquad \Gamma_2 = \frac{(m-b)G^2}{mb}, \qquad \theta = 1.$$

Let the compression operators $\mathcal{C}_k^t$ satisfy Definition 3.1 with parameter $\omega \geq 0$, and run ERIS *with constant learning rate $\lambda_t \equiv \lambda$ and shift stepsize $\gamma_t \equiv \gamma$ as in Theorem 3.2, with*

$$\lambda \;\leq\; \frac{1}{\big(1+4\beta(1+\omega)\big)L}, \qquad \gamma \;=\; \sqrt{\frac{1+2\omega}{2(1+\omega)^3}}, \tag{25}$$

for some fixed $\beta > 0$. Then ERIS *with DSC–SGD satisfies*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\big\|\nabla f(\mathbf{x}^t)\big\|^2 \;\leq\; \frac{2\Phi_0}{\lambda T} \;+\; \frac{3\beta(m-b)G^2}{(1+\omega)L\,\lambda\,mb}, \tag{26}$$

where

$$\Phi_0 := f(\mathbf{x}^0) - f^* + \alpha L\Delta^0 + \frac{\beta}{KL}\sum_{k=1}^{K}\big\|\nabla f_k(\mathbf{x}^0) - \mathbf{s}_k^0\big\|^2,$$

with $\alpha = \frac{3\beta\Gamma_1}{2(1+\omega)L^2\theta}$, $\Delta^t := \frac{1}{K}\sum_{k=1}^{K}\|\mathbf{s}_k^t - \nabla f_k(\mathbf{x}^t)\|^2$, and the stepsize λ is constrained by Theorem 3.2 as

$$\lambda \;\leq\; \lambda_{\max} := \min\{\lambda_1, \lambda_2\}, \quad \lambda_1 = \frac{\sqrt{\beta K}}{\sqrt{1+4\beta(1+\omega)}(1+\omega)L}, \quad \lambda_2 = \frac{1}{(1+4\beta(1+\omega))L}.$$

For any fixed K, ω we can choose $\beta > 0$ sufficiently small so that $\lambda_1 \leq \lambda_2$, and hence $\lambda_{\max} = \lambda_1$. We then set $\lambda = \lambda_1$ and let $T \to \infty$, so that the term $\frac{2\Phi_0}{\lambda T}$ vanishes. Substituting λ_1 into (26) *yields*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\big\|\nabla f(\mathbf{x}^t)\big\|^2 \;\leq\; \frac{3\beta(m-b)G^2}{(1+\omega)L\,mb}\cdot\frac{\sqrt{1+4\beta(1+\omega)}(1+\omega)L}{\sqrt{\beta K}}, \tag{27}$$

$$\leq\; \frac{3\sqrt{\beta}(m-b)G^2}{mb}\,\frac{\sqrt{1+4\beta(1+\omega)}}{\sqrt{K}}. \tag{28}$$

Since β is a fixed constant and $\sqrt{1+4\beta(1+\omega)} = \Theta(\sqrt{1+\omega})$, we obtain the asymptotic bound

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\big\|\nabla f(\mathbf{x}^t)\big\|^2 \;=\; \mathcal{O}\!\left(\frac{(m-b)G^2}{mb}\,\frac{\sqrt{1+\omega}}{\sqrt{K}}\right). \tag{29}$$

Finally, for $b = \Theta(m)$, we can remove the explicit dependence on b and write

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\big\|\nabla f(\mathbf{x}^t)\big\|^2 \;=\; \mathcal{O}\!\left(\frac{G^2\sqrt{1+\omega}}{\sqrt{K}m}\right). \tag{30}$$

Consequently, in this regime the stationarity error of ERIS *with DSC–SGD is controlled solely by the variance of the local mini-batch gradients and grows with the compression variance $(1+\omega)$ while decreasing with the square root of the number of clients K and the amount of local data m.*

C.2 Proof of Lemma C.1

Proof. By the definition of $\mathbf{v}^t$, we derive the following expression:

$$\begin{aligned}
\mathbb{E}_t[\|\mathbf{v}^t\|^2] &= \mathbb{E}_t\left[\left\|\frac{1}{K}\sum_{k=1}^K \mathbf{s}_k^t + \frac{1}{K}\sum_{k=1}^K \mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t)\right\|^2\right] \\
&= \mathbb{E}_t\left[\left\|\frac{1}{K}\sum_{k=1}^K \mathbf{s}_k^t + \frac{1}{K}\sum_{k=1}^K \mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t) + \frac{1}{K}\sum_{k=1}^K \tilde{\mathbf{g}}_k^t - \frac{1}{K}\sum_{k=1}^K \tilde{\mathbf{g}}_k^t\right\|^2\right] \\
&= \mathbb{E}_t\left[\left\|\frac{1}{K}\sum_{k=1}^K \mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t) - \frac{1}{K}\sum_{k=1}^K (\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t) + \frac{1}{K}\sum_{k=1}^K \tilde{\mathbf{g}}_k^t\right\|^2\right] \\
&= \mathbb{E}_t\left[\left\|\frac{1}{K}\sum_{k=1}^K \mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t) - \frac{1}{K}\sum_{k=1}^K (\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t)\right\|^2\right] + \mathbb{E}_t\left[\left\|\frac{1}{K}\sum_{k=1}^K \tilde{\mathbf{g}}_k^t\right\|^2\right] \\
&\quad + 2\left\langle \mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^K \mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t) - \frac{1}{K}\sum_{k=1}^K (\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t)\right], \mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^K \tilde{\mathbf{g}}_k^t\right]\right\rangle \\
&\overset{(3)}{\leq} \mathbb{E}_t\left[\frac{\omega}{K^2}\sum_{k=1}^K \|\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t\|^2\right] + \mathbb{E}_t\left[\left\|\frac{1}{K}\sum_{k=1}^K \tilde{\mathbf{g}}_k^t\right\|^2\right],
\end{aligned} \tag{31}$$

where (31) follows because the cross term vanishes: the compression error has zero mean ($\mathbb{E}_t[\mathcal{C}_k^t(\cdot)] = \cdot$), making their inner product zero in expectation.

Next, we establish upper bounds for each term in Equation (31).

- For the first term, we expand and decompose it as follows:

$$\mathbb{E}_t\left[\frac{\omega}{K^2}\sum_{k=1}^{K}\|\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t\|^2\right]=\mathbb{E}_t\left[\frac{\omega}{K^2}\sum_{k=1}^{K}\|(\tilde{\mathbf{g}}_k^t-\nabla f_k(\mathbf{x}^t))+(\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t)\|^2\right]$$

$$=\mathbb{E}_t\left[\frac{\omega}{K^2}\sum_{k=1}^{K}\big(\|\tilde{\mathbf{g}}_k^t-\nabla f_k(\mathbf{x}^t)\|^2+\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t\|^2\right.$$

$$\left.+2(\tilde{\mathbf{g}}_k^t-\nabla f_k(\mathbf{x}^t))^\top(\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t)\big)\right] \tag{32}$$

$$=\mathbb{E}_t\left[\frac{\omega}{K^2}\sum_{k=1}^{K}\|\tilde{\mathbf{g}}_k^t-\nabla f_k(\mathbf{x}^t)\|^2\right]$$

$$+\frac{\omega}{K^2}\sum_{k=1}^{K}\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t\|^2, \tag{33}$$

  where the last equality holds because the expectation of the cross-term vanishes due to the unbiased estimator assumption, i.e., $\mathbb{E}_t[\tilde{\mathbf{g}}_k^t]=\nabla f_k(\mathbf{x}^t)$, as specified in Assumption A.2.

- Similarly, for the second term, we proceed as follows:

$$\mathbb{E}^t\left[\left\|\frac{1}{K}\sum_{k=1}^{K}\tilde{\mathbf{g}}_k^t\right\|^2\right]=\mathbb{E}^t\left[\frac{1}{K^2}\sum_{k=1}^{K}\big\|(\tilde{\mathbf{g}}_k^t-\nabla f_k(\mathbf{x}^t))+\nabla f_k(\mathbf{x}^t)\big\|^2\right]$$

$$=\mathbb{E}^t\left[\frac{1}{K^2}\sum_{k=1}^{K}\big(\|\tilde{\mathbf{g}}_k^t-\nabla f_k(\mathbf{x}^t)\|^2+\|\nabla f_k(\mathbf{x}^t)\|^2\right.$$

$$\left.+2(\tilde{\mathbf{g}}_k^t-\nabla f_k(\mathbf{x}^t))^\top\nabla f_k(\mathbf{x}^t)\big)\right]$$

$$=\mathbb{E}^t\left[\frac{1}{K^2}\sum_{k=1}^{K}\|\tilde{\mathbf{g}}_k^t-\nabla f_k(\mathbf{x}^t)\|^2\right]+\|\nabla f(\mathbf{x}^t)\|^2, \tag{34}$$

The proof concludes by substituting (33) and (34) into (31). □

C.3 Proof of Lemma C.2

Proof. By the definition of the shift update $\mathbf{s}_k^{t+1}=\mathbf{s}_k^t+\gamma^t\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)$, we have:

$$\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\|\nabla f_k(\mathbf{x}^{t+1})-\mathbf{s}_k^{t+1}\|^2\right]=\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\big\|\nabla f_k(\mathbf{x}^{t+1})-\mathbf{s}_k^t-\gamma_t\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\big\|^2\right]$$

$$=\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\big\|\big(\nabla f_k(\mathbf{x}^{t+1})-\nabla f_k(\mathbf{x}^t)\big)+\big(\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t-\gamma_t\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\big)\big\|^2\right]$$

$$\le\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\Big((1+\frac{1}{\beta_t})\|\nabla f_k(\mathbf{x}^{t+1})-\nabla f_k(\mathbf{x}^t)\|^2\right.$$

$$\left.+(1+\beta_t)\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t-\gamma_t\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\|^2\Big)\right] \tag{35}$$

$$\overset{(8)}{\le}\mathbb{E}_t\left[(1+\frac{1}{\beta_t})L^2\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2+(1+\beta_t)\frac{1}{K}\sum_{k=1}^{K}\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t-\gamma_t\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\|^2\right], \tag{36}$$

where the Equation (35) is obtained from Young's inequality $\|\mathbf{a}+\mathbf{b}\|^2\le(1+\frac{1}{\beta})\|\mathbf{a}\|^2+(1+\beta)\|\mathbf{b}\|^2$ with any $\beta_t>0$.

To further bound the second term in (36), we expand the squared norm:

$$\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t-\gamma_t\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\right\|^2\right]$$
$$=\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\left(\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t\|^2+\gamma_t^2\|\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\|^2\right.\right.$$
$$\left.\left.-2\gamma_t\langle\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t,\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\rangle\right)\right] \tag{37}$$

Since the expectation of the inner product term satisfies:

$$\mathbb{E}_t[\langle\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t,\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\rangle]=\mathbb{E}_t[||\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t||^2],$$

Equation (37) simplifies to:

$$\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t-\gamma_t\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\right\|^2\right]$$
$$=\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\left((1-2\gamma_t)\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t\|^2+\gamma_t^2\|\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\|^2\right)\right] \tag{38}$$

Then, applying Definition 3.1 to the term $(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)$, we derive the following inequality:

$$\begin{aligned}\mathbb{E}_t\left[\|\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\|^2\right]&=\mathbb{E}_t\left[\left\|\left(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t\right)+\left(\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)-(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\right)\right\|^2\right]\\&=\mathbb{E}_t\left[\|\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t\|^2\right]+2\mathbb{E}_t\left[\left\langle\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t,\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)-(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\right\rangle\right]\\&\quad+\mathbb{E}_t\left[\left\|\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)-(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\right\|^2\right]\\&=\mathbb{E}_t\left[\|\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t\|^2\right]+\mathbb{E}_t\left[\left\|\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)-(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\right\|^2\right]\\&\leq\left(1+\omega\right)\mathbb{E}_t\left[\|\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t\|^2\right].\end{aligned} \tag{39}$$

Substituting Equation (39) into (38), we simplify the second term as follows:

$$\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t-\gamma_t\mathcal{C}_k^t(\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t)\right\|^2\right]$$
$$\leq\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\left((1-2\gamma_t)\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t\|^2+\gamma_t^2(1+\omega)\|\tilde{\mathbf{g}}_k^t-\mathbf{s}_k^t\|^2\right)\right]$$
$$\overset{(33)}{=}\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\left((1-2\gamma_t+\gamma_t^2(1+\omega))\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t\|^2\right.\right.$$
$$\left.\left.+\gamma_t^2(1+\omega)\|\tilde{\mathbf{g}}_k^t-\nabla f_k(\mathbf{x}^t)\|^2\right)\right]. \tag{40}$$

By plugging (40) into (36), we obtain:

$$\mathbb{E}_t\left[\frac{1}{K}\sum_{k=1}^{K}\|\nabla f_k(\mathbf{x}^{t+1})-\mathbf{s}_k^{t+1}\|^2\right] \leq \mathbb{E}_t\Bigg[(1+\frac{1}{\beta_t})L^2\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2 + (1+\beta_t)\frac{1}{K}\sum_{k=1}^{K}(1-2\gamma_t+\gamma_t^2(1+\omega))\|\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t\|^2 + (1+\beta_t)\frac{1}{K}\sum_{k=1}^{K}\gamma_t^2(1+\omega)\|\tilde{\mathbf{g}}_k^t-\nabla f_k(\mathbf{x}^t)\|^2\Bigg] \tag{41}$$

Finally, setting $\beta_t = \frac{1}{1+2\omega}$, and $\gamma_t = \sqrt{\frac{1+2\omega}{2(1+\omega)^3}}$, we approximate the second term in Equation (41) with the following upper bound:

$$\begin{aligned}(1+\beta_t)(1-2\gamma_t+\gamma_t^2(1+\omega)) &= \frac{2(1+\omega)}{(1+2\omega}\left(1-2\sqrt{\frac{1+2\omega}{2(1+\omega)^3}}+\frac{(1+2\omega)}{2(1+\omega)^2}\right)\\ &\leq 1-\frac{1}{2(1+\omega)} \qquad \forall\omega\geq 0.\end{aligned}$$

Substituting this bound into Equation (41), we obtain Equation (18), thereby completing the proof of Lemma C.2. □

C.4 Proof of Lemma C.3

Proof. Given the definition $\mathcal{S}^t := \frac{1}{K}\sum_{k=1}^{K}||\nabla f_k(\mathbf{x}^t)-\mathbf{s}_k^t||^2$, we can derive a recursive bound for $\mathcal{S}^{t+1}$ using Lemma C.2:

$$\begin{aligned}\mathbb{E}_t\left[\mathcal{S}^{t+1}\right] &\leq \mathbb{E}_t\left[\left(1-\frac{1}{2(1+\omega)}\right)\mathcal{S}^t+\frac{1}{(1+\omega)K}\sum_{k=1}^{K}\|\tilde{\mathbf{g}}_k^t-\nabla f_k(\mathbf{x}^t)\|^2+2(1+\omega)L^2\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2\right]\\ &\overset{(11a)}{\leq} \mathbb{E}_t\left[\left(1-\frac{1}{2(1+\omega)}\right)\mathcal{S}^t+\frac{\Gamma_1\Delta^t+\Gamma_2}{(1+\omega)}+2(1+\omega)L^2\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2\right]\end{aligned} \tag{42}$$

We now use Equation (42) along with (16) to bound the potential function Φ_{t+1}, as defined in Equation (19):

$$\begin{aligned}
\mathbb{E}_t[\Phi_{t+1}] &:= \mathbb{E}_t\left[f(\mathbf{x}^{t+1}) - f^* + \alpha L\Delta^{t+1} + \frac{\beta}{L}\mathcal{S}^{t+1}\right] \\
&\leq \mathbb{E}_t\Bigg[f(\mathbf{x}^t) - \lambda_t||\nabla f(\mathbf{x}^t)||^2 + \frac{\lambda^2 L}{2}\|\mathbf{v}^t\|^2 - f^* + \alpha L\Delta^{t+1} \\
&\quad + \frac{\beta}{L}\left(\left(1 - \frac{1}{2(1+\omega)}\right)\mathcal{S}^t + \frac{\Gamma_1\Delta^t + \Gamma_2}{(1+\omega)} + 2(1+\omega)L^2\|\mathbf{x}^{t+1} - \mathbf{x}^t\|^2\right)\Bigg] \\
&\overset{(11b)}{\leq} \mathbb{E}_t\Bigg[f(\mathbf{x}^t) - f^* - \lambda_t||\nabla f(\mathbf{x}^t)||^2 + \frac{\lambda^2 L}{2}\|\mathbf{v}^t\|^2 \\
&\quad + \alpha L\Big((1-\theta)\Delta^t + \Gamma_3||\nabla f(\mathbf{x}^t)||^2 + \Gamma_4||\mathbf{x}^{t+1} - \mathbf{x}^t||^2\Big) \\
&\quad + \frac{\beta}{L}\left(\left(1 - \frac{1}{2(1+\omega)}\right)\mathcal{S}^t + \frac{\Gamma_1\Delta^t + \Gamma_2}{(1+\omega)} + 2(1+\omega)L^2\|\mathbf{x}^{t+1} - \mathbf{x}^t\|^2\right)\Bigg] \\
&= \mathbb{E}_t\Bigg[f(\mathbf{x}^t) - f^* - \lambda_t||\nabla f(\mathbf{x}^t)||^2 + \left(\frac{1}{2} + \alpha\Gamma_4 + \beta(1+\omega)\right)L\lambda_t^2\|\mathbf{v}^t\|^2 \\
&\quad + \alpha L\Big((1-\theta)\Delta^t + \Gamma_3||\nabla f(\mathbf{x}^t)||^2\Big) \\
&\quad + \frac{\beta}{L}\left(\left(1 - \frac{1}{2(1+\omega)}\right)\mathcal{S}^t + \frac{\Gamma_1\Delta^t + \Gamma_2}{(1+\omega)}\right)\Bigg] \qquad (43)
\end{aligned}$$

where the last equality follows the adopted update rule $\mathbf{x}^{t+1} = \mathbf{x}^t - \lambda_t\mathbf{v}^t$. Now adopting $\mathcal{S}^t$ into the definition of $\mathbb{E}_t[||\mathbf{v}^t||^2]$ provided in Lemma C.1, we obtain:

$$\begin{aligned}
\mathbb{E}_t[\|\mathbf{v}^t\|^2] &\leq \mathbb{E}_t\left[\frac{(1+\omega)}{K^2}\sum_{k=1}^{K}\|\tilde{\mathbf{g}}_k^t - \nabla f_k(\mathbf{x}^t)\|^2 + \frac{\omega}{K}\mathcal{S}^t + \|\nabla f(\mathbf{x}^t)\|^2\right] \\
&\overset{(11a)}{\leq} \mathbb{E}_t\left[\frac{(1+\omega)}{K}(\Gamma_1\Delta^t + \Gamma_2) + \frac{\omega}{K}\mathcal{S}^t + \|\nabla f(\mathbf{x}^t)\|^2\right] \qquad (44)
\end{aligned}$$

Substituting Equation (44) into (43), we obtain as follows:

$$\begin{aligned}
\mathbb{E}_t[\Phi_{t+1}] &\leq f(\mathbf{x}^{t+1}) - f^* \\
&\quad + \left[\left(\frac{1}{2} + \alpha\Gamma_4 + 2\beta(1+\omega)\right)\frac{(1+\omega)\Gamma_1\lambda_t^2}{K} + \alpha(1-\omega) + \frac{\beta\Gamma_1}{(1+\omega)L^2}\right]L\Delta^t \\
&\quad + \left[\left(\frac{1}{2} + \alpha\Gamma_4 + 2\beta(1+\omega)\right)\frac{\omega L^2\lambda_t^2}{K} + \beta\big(1 - \frac{1}{2(1+\omega)}\big)\right]\frac{\mathcal{S}^t}{L} \\
&\quad - \left[\lambda_t - \left(\frac{1}{2} + \alpha\Gamma_4 + 2\beta(1+\omega)\right)L\lambda_t^2 - \alpha L\Gamma_3\right]||\nabla f(x^t)||^2 \\
&\quad + \left[\left(\frac{1}{2} + \alpha\Gamma_4 + 2\beta(1+\omega)\right)\frac{(1+\omega)L\lambda_t^2}{K} + \frac{\beta}{(1+\omega)L}\right]\Gamma_2 \qquad (45)
\end{aligned}$$

To ensure that the right-hand side of Equation (45) remains consistent with the potential function $\Phi_t := f(\mathbf{x}^t) - f^* + \alpha L\Delta^t + \frac{\beta}{L}\mathcal{S}_t$, we select the parameters α, β, and λ_t to satisfy the following constraints:

$$\left(\frac{1}{2} + \alpha\Gamma_4 + 2\beta(1+\omega)\right)\frac{(1+\omega)\Gamma_1\lambda_t^2}{K} + \alpha(1-\omega) + \frac{\beta\Gamma_1}{(1+\omega)L^2} \leq \alpha \qquad (46)$$

Table 3: Asymptotic utility / accuracy bounds (average squared gradient norm after T rounds) for different (local) differentially-private FL algorithms for the nonconvex problem in (1), compared to the non-DP utility bound of ERIS-SGD with DSC. Here K is the number of clients, m the number of samples per client, n the model dimension, ω the compressor variance parameter, and (ε, δ) the privacy parameters. All bounds hide absolute constants and, where standard, additional logarithmic factors. For SoteriaFL, $\tau := (1+\omega)^{3/2}/\sqrt{K}$. Note that smaller values of the bound correspond to better utility / accuracy.

Algorithm	**Privacy**	**Utility / Accuracy**
Distributed DP-SRM [129]	(ε, δ)-DP	$\tilde{\mathcal{O}}\left(\frac{\sqrt{n\log(1/\delta)}}{Km\varepsilon}\right)$
SDM-DSGD [130]	(ε, δ)-LDP	$\tilde{\mathcal{O}}\left(\frac{\sqrt{n\log(1/\delta)}}{\sqrt{K}m\varepsilon}\right)$
Q-DPSGD-1 [36]	(ε, δ)-LDP	$\tilde{\mathcal{O}}\left(\frac{\left(\frac{\tilde{\nu}^2}{K}+\frac{1}{m}\right)^{2/3}\left(n\log(1/\delta)\right)^{1/3}}{m^{2/3}\varepsilon^{2/3}}\right)$
CDP-SGD [60]	(ε, δ)-LDP	$\tilde{\mathcal{O}}\left(\frac{\sqrt{(1+\omega)\,n\log(1/\delta)}}{\sqrt{K}m\varepsilon}\right)$
SoteriaFL-SGD [60]	(ε, δ)-LDP	$\tilde{\mathcal{O}}\left(\frac{\sqrt{(1+\omega)\,n\log(1/\delta)}}{\sqrt{K}m\varepsilon}\,(1+\sqrt{\tau})\right)$
ERIS-SGD (+DSC)	—	$\tilde{\mathcal{O}}\left(\frac{\sqrt{1+\omega}}{\sqrt{K}m}\right)$

$$\left(\frac{1}{2}+\alpha\Gamma_4+2\beta(1+\omega)\right)\frac{\omega L^2\lambda_t^2}{K}+\beta\big(1-\frac{1}{2(1+\omega)}\big)\leq\beta \tag{47}$$

Although these are not the strictest possible bounds for a fair comparison with the utility results of SoteriaFL, we adopt the same choices for α, β, and λ_t, ensuring they satisfy conditions (46) and (47):

$$\alpha \geq \frac{3\beta\Gamma_1}{2(1+\omega)L^2\theta} \qquad \forall\beta>0 \tag{48}$$

$$\lambda_t \equiv \lambda \leq \frac{\sqrt{\beta K}}{\sqrt{1+2\alpha\Gamma_4+4\beta(1+\omega)}(1+\omega)L} \tag{49}$$

Here, Equation (48) follows from the constraint in (46), while (49) ensures compatibility with the potential function definition in (19). Additionally, we impose a further bound on λ_t to guarantee that the negative gradient squared term remains sufficiently large (i.e., $\geq \frac{\lambda_t}{2}\left\|\nabla f(\mathbf{x}^t)\right\|^2$), obtaining:

$$\lambda_t \equiv \lambda \leq \frac{1}{(1+2\alpha\Gamma_4+4\beta(1+\omega)+2\alpha\Gamma_3/\lambda^2)L} \tag{50}$$

Finally, substituting the conditions (48)–(50) into Equation (45), we obtain:

$$\mathbb{E}_t[\Phi_{t+1}] \leq \Phi_t - \frac{\lambda_t}{2}||\nabla f(x^t)||^2 + \frac{3\beta}{2(1+\omega)L}\Gamma_2 \tag{51}$$

The last term is obtained directly by applying the bound from Equation (50), completing the proof.

□

C.5 Utility Comparison

Table 3 summarizes the asymptotic utility/accuracy guarantees of existing differentially-private FL algorithms, most of them with communication compression, and compares them to our non-DP utility bound for ERIS-SGD with DSC. Distributed DP-SRM [129] provides a global (ε, δ)-DP baseline

without compression: its utility improves linearly in the number of clients K and in the number of samples per client m, but it does not consider LDP and therefore is not directly comparable to the LDP-based protocols in the rest of the table.

SDM-DSGD [130] and Q-DPSGD-1 [36] are early attempts to combine local DP with compressed communication. However, SDM-DSGD assumes random-k sparsification and requires $1+\omega \ll \log T$ (i.e., communicating at least $k \gtrsim n/\log T$ coordinates per round), and its bound hides logarithmic factors that grow faster than $(1+\omega)$. Q-DPSGD-1 relies on a different compression assumption ($\mathbb{E}\big[\|\mathcal{C}(\mathbf{x})-\mathbf{x}\|^2\big] \le \tilde{\nu}^2$, with parameter $\tilde{\nu}^2$ playing a similar role to our $1+\omega$) and incurs a strictly worse utility by a factor $T^{1/6}$ compared to later methods, as already observed in [60].

CDP-SGD [60] can be seen as a direct compressed analogue of DP-SGD: it achieves (ε,δ)-LDP and a utility that degrades with $\sqrt{(1+\omega)n}/(\sqrt{K}\,m\varepsilon)$, but still requires $\mathcal{O}(m^2)$ communication rounds when the local dataset size m is large. SoteriaFL-SGD/GD improves upon CDP-SGD via shifted compression: it preserves the same dependence on $(1+\omega), K, m, n$ up to a mild factor $(1+\sqrt{\tau})$, where $\tau = (1+\omega)^{3/2}/\sqrt{K}$ becomes negligible as $K \gg (1+\omega)^3$, while reducing the total communication to $\mathcal{O}(m)$ rounds.

In contrast, ERIS-SGD with DSC does not inject any differentially-private noise to ensure formal (ε,δ)-DP (although standard LDP mechanisms can be applied on top of it as shown in Figure 4), and thus its bound cannot be directly compared in terms of privacy guarantees. Nevertheless, once privacy noise is removed, our analysis shows that ERIS achieves a dimension-free non-private utility bound that scales as $\tilde{\mathcal{O}}\big(\sqrt{1+\omega}/(m\sqrt{K})\big)$, yielding faster convergence under the same optimization assumptions. Moreover, ERIS exhibits the same favorable dependence on the number of clients K and on the compression variance $(1+\omega)$ as SoteriaFL-style methods, while operating in a fully serverless, sharded architecture. Empirically (see Figure 4), when we add the *same* LDP mechanism to both methods, ERIS and SoteriaFL achieve comparable utility for a given (ε,δ), but ERIS requires less injected noise to reach this regime due to the privacy amplification inherent in its decentralized, sharded aggregation architecture.

D Privacy Guarantees for ERIS with DSC

In this section, we present the detailed proof of Theorem 3.3, which establishes an upper bound on the information leakage incurred by ERIS with DSC under the honest-but-curious threat model. The analysis follows an information-theoretic approach by bounding the mutual information between a client's local dataset D_k and the adversary's partial view of the transmitted model updates $\mathbf{v}_{k,(a)}^t = (\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t) \odot \mathbf{m}_{\mathcal{C}_k^t} \odot \mathbf{m}_{(a)}^t$ over T communication rounds. We then extend the result to colluding adversaries, who may share observations to amplify their attack.

D.1 Proof of Theorem 3.3

Proof. For lighter notation, define the combined mask $\mathbf{m}_k^t := \mathbf{m}_{\mathcal{C}_k^t} \odot \mathbf{m}_{(a)}^t$, so that $\mathbf{v}_{k,(a)}^t = \mathbf{U}_k^t \odot \mathbf{m}_k^t$. We work directly with the pre-mask update $\mathbf{U}_k^t$ and write

$$\mathcal{H}_t := \sigma\Big(\{(\mathbf{v}_{k,(a)}^\ell, \mathbf{m}_k^\ell) : 0 \le \ell < t\}\Big)$$

for this observer's history before round t. Revealing the masks can only increase information. The chain rule therefore gives

$$\begin{aligned} I\big(D_k; \{\mathbf{v}_{k,(a)}^t\}_{t=0}^{T-1}\big) &\le I\big(D_k; \{(\mathbf{v}_{k,(a)}^t, \mathbf{m}_k^t)\}_{t=0}^{T-1}\big) \\ &= \sum_{t=0}^{T-1} I\big(D_k; \mathbf{v}_{k,(a)}^t, \mathbf{m}_k^t \mid \mathcal{H}_t\big) \\ &= \sum_{t=0}^{T-1} I\big(D_k; \mathbf{U}_k^t \odot \mathbf{m}_k^t \mid \mathcal{H}_t, \mathbf{m}_k^t\big), \end{aligned}$$

where the last equality uses $I(D_k; \mathbf{m}_k^t \mid \mathcal{H}_t) = 0$.

Fix a history h and a mask realization $\mathbf{m}$, and order the revealed coordinates as

$$S(\mathbf{m}) := \{i : m_i \neq 0\} = \{i_1, \dots, i_s\}, \qquad J_j := \{i_1, \dots, i_{j-1}\}.$$

The nonzero mask values are known invertible scalings. Conditional mask independence and the coordinate-wise chain rule imply

$$\begin{aligned} &I\big(D_k; \mathbf{U}_k^t \odot \mathbf{m}_k^t \mid \mathcal{H}_t = h, \mathbf{m}_k^t = \mathbf{m}\big) \\ &\quad = I\big(D_k; \{U_{k,i}^t\}_{i \in S(\mathbf{m})} \mid \mathcal{H}_t = h\big) \\ &\quad = \sum_{j=1}^{s} I\big(D_k; U_{k,i_j}^t \mid \mathcal{H}_t = h, \mathbf{U}_{k,J_j}^t\big) \;\le\; |S(\mathbf{m})|\, C_{\max}. \end{aligned}$$

In particular, the constant in the theorem can be chosen as

$$C_{\max} := \sup_{t,i,J} \operatorname*{ess\,sup}_{h} I\big(D_k; U_{k,i}^t \mid \mathcal{H}_t = h, \mathbf{U}_{k,J}^t\big), \qquad J \subseteq \{1, \dots, n\} \setminus \{i\}.$$

Here the conditional mutual information averages over $\mathbf{U}_{k,J}^t$; the essential supremum is over the pre-round history h. By the expected-exposure assumption, $\mathbb{E}[|S(\mathbf{m}_k^t)|] = np/A$. Taking expectations gives

$$I\big(D_k; \mathbf{U}_k^t \odot \mathbf{m}_k^t \mid \mathcal{H}_t, \mathbf{m}_k^t\big) \le \frac{n}{A}\, p\, C_{\max},$$

and summing over $t = 0, \dots, T-1$ yields

$$I\big(D_k; \{\mathbf{v}_{k,(a)}^t\}_{t=0}^{T-1}\big) \le T\, \frac{n}{A}\, p\, C_{\max}.$$

□

Remark D.1 (Gaussian-channel instantiation of $C_{\max}$). For a fixed client k and history $\mathcal{H}_t = h_t$, let $U_{k,i}^t := \widetilde{g}_{k,i}^t - s_{k,i}^t$ denote coordinate i of the pre-mask update. Consider the Gaussian-channel instantiation

$$U_{k,i}^t = \mu_{i,t}(D_k, h_t) + Z_{i,t}, \qquad Z_{i,t} \perp D_k \mid \mathcal{H}_t = h_t, \qquad Z_{i,t} \sim \mathcal{N}\big(0, \sigma_{w,i,t}^2\big),$$

where, conditional on $\mathcal{H}_t = h_t$, the noise vector is independent of D_k, its coordinates are mutually independent, and $\sigma_{w,i,t}^2 > 0$. The dataset-dependent means may be correlated across coordinates. Define the finite between-dataset variance $\sigma_{b,i,t}^2 := \operatorname{Var}_{D_k \mid \mathcal{H}_t = h_t}[\mu_{i,t}(D_k, h_t)]$. Then

$$\operatorname{Var}\big(U_{k,i}^t \mid \mathcal{H}_t = h_t\big) = \sigma_{b,i,t}^2 + \sigma_{w,i,t}^2.$$

Using the Gaussian maximum-entropy property, and reporting mutual information in bits, gives

$$I\big(D_k; U_{k,i}^t \mid \mathcal{H}_t = h_t\big) \le \frac{1}{2} \log_2 \left(1 + \frac{\sigma_{b,i,t}^2}{\sigma_{w,i,t}^2}\right).$$

Moreover, $U_{k,i}^t \perp \mathbf{U}_{k,J}^t \mid (D_k, \mathcal{H}_t = h_t)$ for $i \notin J$, so

$$\begin{aligned} I(D_k; U_{k,i}^t \mid \mathcal{H}_t = h_t, \mathbf{U}_{k,J}^t) &= I(D_k; U_{k,i}^t \mid \mathcal{H}_t = h_t) - I(U_{k,i}^t; \mathbf{U}_{k,J}^t \mid \mathcal{H}_t = h_t) \\ &\le \frac{1}{2} \log_2 \left(1 + \frac{\sigma_{b,i,t}^2}{\sigma_{w,i,t}^2}\right). \end{aligned}$$

A uniform supremum of this variance-ratio bound over coordinates, rounds, and histories therefore bounds $C_{\max}$ in Theorem 3.3. The scalar bound does not require the dataset-dependent means $\mu_{i,t}(D_k, h_t)$ to be Gaussian; equality holds when these means follow a Gaussian random-effects model. Consequently, the ratio $\sigma_{b,i,t}^2/\sigma_{w,i,t}^2$ provides an operational signal-to-noise characterization of the per-coordinate quantity entering $C_{\max}$.

Operational estimation of $C_{\max}$. At a fixed update state, consider M independently sampled client datasets and R repeated stochastic updates for each dataset. Under the conditional independent-noise model of Remark D.1, the following marginal variance estimates give a plug-in estimate of their conditional information bound. For the theorem, the dataset law at a realized history is $D_k \mid \mathcal{H}_t = h_t$; resampling datasets at a fixed checkpoint instead audits the chosen resampling law. Let $U_{i,t}^{(m,r)}$ denote coordinate i of repetition r for dataset m, and define

$$\overline{U}_{i,t}^{(m)} = \frac{1}{R}\sum_{r=1}^{R} U_{i,t}^{(m,r)}, \qquad \overline{U}_{i,t} = \frac{1}{M}\sum_{m=1}^{M} \overline{U}_{i,t}^{(m)}.$$

The within-dataset variance can be estimated as

$$\widehat{\sigma}_{w,i,t}^2 = \frac{1}{M}\sum_{m=1}^{M} \frac{1}{R-1}\sum_{r=1}^{R} \left(U_{i,t}^{(m,r)} - \overline{U}_{i,t}^{(m)}\right)^2,$$

while the between-dataset component is estimated by correcting the variance of the dataset-specific means for finite R:

$$\widehat{\sigma}_{b,i,t}^2 = \left[\frac{1}{M-1}\sum_{m=1}^{M} \left(\overline{U}_{i,t}^{(m)} - \overline{U}_{i,t}\right)^2 - \frac{\widehat{\sigma}_{w,i,t}^2}{R}\right]_+,$$

where $[z]_+ := \max\{z, 0\}$. This gives the empirical per-coordinate estimate

$$\widehat{C}_{i,t} = \frac{1}{2}\log_2\left(1 + \frac{\widehat{\sigma}_{b,i,t}^2}{\widehat{\sigma}_{w,i,t}^2}\right).$$

For a finite audit set $\mathcal{A}$ of coordinate–round pairs, we denote

$$\widehat{C}_{\max}^{\text{obs}} := \max_{(i,t)\in\mathcal{A}} \widehat{C}_{i,t}.$$

Unlike $C_{\max}$ in Theorem 3.3, $\widehat{C}_{\max}^{\text{obs}}$ is an observed maximum over the audited states and should therefore not be interpreted as a certified supremum over all possible datasets and training histories. Marginal variance estimates alone do not verify the independent-noise assumption. If $\sigma_{w,i,t}^2 = 0$, this Gaussian-channel argument gives no finite bound.

D.2 Privacy Under Colluding Aggregators

We now extend our analysis to a coalition of aggregators that share their shards before attempting the attack. Let $\mathcal{C} \subseteq \{1, \ldots, A\}$ denote the set of colluding aggregators with cardinality $A_c := |\mathcal{C}|$.

Corollary D.2 (Colluding–aggregator privacy bound)**.** *Assume the setting of Theorem 3.3. For every communication round $t \in \{0, \ldots, T-1\}$ let the union mask*

$$\mathbf{m}_{\text{col}}^t := \bigvee_{a\in\mathcal{C}} \mathbf{m}_{(a)}^t \quad \big(\vee \textit{ denotes the element-wise logical } \text{OR}\big)$$

select the coordinates revealed to the colluding coalition. Define the coalition's view of client k at round t as

$$\mathbf{v}_{k,\text{col}}^t := (\tilde{\mathbf{g}}_k^t - \mathbf{s}_k^t) \;\odot\; \mathbf{m}_{\mathcal{C}_k^t} \;\odot\; \mathbf{m}_{\text{col}}^t.$$

Assuming the mask-independence and uniform conditional-information assumptions of Theorem 3.3 also hold with $\mathcal{H}_t$ denoting the coalition's past shards and masks, under the honest-but-curious *threat model, the mutual information between the client's private dataset D_k and the coalition's transcript over T rounds satisfies*

$$I\big(D_k; \{\mathbf{v}_{k,\text{col}}^t\}_{t=0}^{T-1}\big) \;\le\; n\,T\,\frac{pA_c}{A}\,C_{\max},$$

where $C_{\max}$ bounds the same conditional per-coordinate terms for the coalition history.

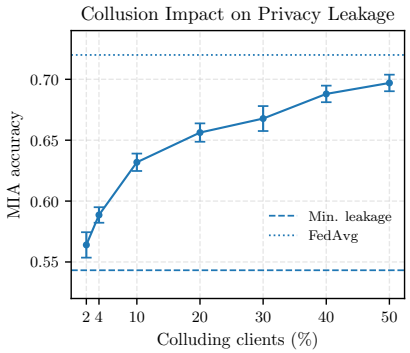

Figure 5: Impact of honest-but-curious client collusion in ERIS.

Proof. The extension follows the same steps as in Appendix D, using the coalition history and replacing the per-shard mask $\mathbf{m}_{(a)}^t$ by the union mask

$$\mathbf{m}_{\text{col}}^t \;=\; \bigvee_{a\in\mathcal{C}} \mathbf{m}_{(a)}^t,$$

where $\mathcal{C}$ is the set of colluding shards of size A_c. Since the shards are disjoint, the retained support $S_{\text{col}}^t := \{i : m_{\mathcal{C}_k^t,i} m_{\text{col},i}^t \neq 0\}$ satisfies

$$\mathbb{E}|S_{\text{col}}^t| = \sum_{a\in\mathcal{C}} \mathbb{E}|S(\mathbf{m}_{\mathcal{C}_k^t} \odot \mathbf{m}_{(a)}^t)| = A_c \,\frac{np}{A}.$$

The masks remain conditionally independent of $(D_k, \mathbf{U}_k^t)$ given the coalition history. Apply the same coordinate-wise chain rule with that history and the stated uniform bound $C_{\max}$.

Substituting this modification into the rest of the derivation yields

$$I\big(D_k; \{\mathbf{v}_{k,\text{col}}^t\}_{t=0}^{T-1}\big) \;\le\; T\, n\, \frac{p\, A_c}{A}\, C_{\max},$$

as claimed. In particular:

- If $A_c = 1$, this reduces to Theorem 3.3.
- If $A_c = A$, the sharding protection vanishes and the bound becomes $I \le n\,T\,p\,C_{\max}$, governed solely by the compression mechanism.

□

Remark D.3. For a common $C_{\max}$ across coalition histories, Corollary D.2 shows that the leakage bound grows *linearly* with the coalition size A_c: increasing collusion progressively weakens the privacy amplification provided by FSA, rather than causing an abrupt loss of protection at a fixed coalition size. Consequently, anticipating up to $A_c^{\max}$ colluding aggregators, one can retain the original leakage bound of Theorem 3.3 by increasing the shard count to $A \mapsto A \cdot A_c^{\max}$ or, equivalently, by decreasing the retention probability to $p \mapsto p/A_c^{\max}$, thereby preserving the product $\frac{pA_c}{A}$.

Empirical Validation To assess the robustness of ERIS against coordinated leakage attempts, we evaluate how MIA accuracy evolves as multiple honest-but-curious clients collude by sharing their received shards. Figure 5 reports the resulting leakage curve. As the collusion group grows, MIA accuracy increases smoothly but remains consistently below the FedAvg baseline and close to the minimum achievable leakage, even when 50% of clients collude. These results confirm that the shard-based decomposition in ERIS meaningfully amplifies privacy, limiting the adversary's advantage even under strong collusion scenarios.

E Experimental Setup

This section provides additional details on the experimental configuration used throughout the paper, including model architectures, training protocols, and hardware resources. We also describe the software libraries, dataset licenses, and implementation details to ensure full reproducibility.

E.1 Models and Hyperparameter Settings

We use 5-fold cross-validation across all experiments, varying the random seed for both data generation and model initialization to ensure reproducibility. Each dataset is paired with an appropriate architecture: GPT-Neo [117] (`EleutherAI/gpt-neo-1.3B`, 1.3B parameters) from HuggingFace for CNN/DailyMail, DistilBERT [118] (`distilbert-base-uncased`, 67M parameters) for IMDB, ResNet-9 [119] (1.65M parameters) for CIFAR-10, and LeNet-5 [120] (62K parameters) for MNIST. For both IID and non-IID settings, we use one local update per client per round (i.e., unbiased gradient estimator), except for GPT-Neo, where memory constraints require two local epochs with a batch size of 8. In the biased setting (multiple local updates per round), we use a batch size of 16 for IMDB and 64 for CIFAR-10 and MNIST under IID conditions. In all settings, each client reserves 30% of its local data for evaluation. To ensure fair comparison of communication costs—which directly depend on the number of rounds—we cap the total rounds for all baselines at the point where FedAvg converges, determined by the minimum validation loss (generally the first to converge). This results in 2-4 rounds for CNN/DailyMail, 14–22 for IMDB, 80–140 for CIFAR-10, and 120–250 for MNIST in the unbiased setting. In the biased setting (two local epochs per round), the ranges are 4–16 for IMDB, 60–140 for CIFAR-10, and 80–200 for MNIST. Unless otherwise specified, we set $A = K$. Consequently, a fixed non-colluding aggregator observes one shard from each client per round, corresponding to $1/A = 10\%$, 4%, 2%, and 2% of the uncompressed update for CNN/DailyMail, IMDB, CIFAR-10, and MNIST, respectively. With DSC, the observable fraction becomes p/A; under the compression rates used in the main experiments (Appendix F.2), this corresponds to approximately 0.1%, 0.0005%, 0.012%, and 0.066%, respectively. We use a learning rate of $5\text{e}{-5}$ for CNN/DailyMail and IMDB, and 0.01 for CIFAR-10 and MNIST. For optimization, we adopt Adam [131] (with `weight_decay` $= 0.0$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1\text{e}{-8}$) on CNN/DailyMail and IMDB, and SGD [132] with momentum 0.9 for CIFAR-10 and MNIST. For experiments involving differential privacy, we use the Opacus library [133].

E.2 Implementation Details of Privacy Attacks

We evaluate privacy leakage under the standard *honest-but-curious* threat model, where an adversary (e.g., a compromised aggregator or server) can observe all transmitted model updates derived from each client's private dataset. We implement two widely studied categories of attacks: *Membership Inference Attacks (MIA)* [16, 61, 62, 68, 69] and *Data Reconstruction Attacks (DRA)* [11, 13–15, 65, 66, 134].

Membership Inference Attacks. We adopt a distributed variant of the privacy auditing framework of Steinke et al. [121]. For each client, 50% of the local data pool is designated as *canary* samples and randomly split equally between samples included in and excluded from local training. After training, canaries are ranked by model confidence or gradient alignment; the top third are labeled as "in," the bottom third as "out," while the middle third are discarded to mitigate uncertainty bias. Evaluation is repeated on the same canary sets across all methods and folds of the cross-validation. To capture privacy leakage throughout training, MIA accuracy is computed at each round and for each client; the reported score corresponds to the maximum, over all T rounds, of the average accuracy across K clients. This ensures comparability across methods with different convergence speeds.

Data Reconstruction Attacks. For DRA, we adopt the strongest white-box threat model, where the adversary is assumed to access the gradient of a single training sample. We implement four representative gradient inversion methods: DLG [12], iDLG [13], ROG [8], and GGL [7]. ROG is specifically tailored to reconstruct images from obfuscated gradients, while GGL further strengthens gradient inversion by using a generative prior and an adaptive loss that accounts for the gradient transformation induced by privacy defenses. All methods are evaluated on the same subset of 200 randomly sampled data points within each cross-validation fold to ensure fairness. For GGL, we use a fixed optimization budget of 300 steps per reconstruction trial. Reconstruction quality is assessed

with LPIPS, capturing perceptual similarity. Further algorithmic descriptions on each attack are provided in Appendix F.6.

E.3 Licenses and Hardware

All experiments were implemented in Python 3.13 using open-source libraries: PyTorch 2.6 [135] (BSD license), Flower 1.12 [136] (Apache 2.0), Matplotlib 3.10 [137] (BSD), Opacus 1.5 [133] (Apache 2.0) and Pandas 2.2 [138] (BSD). We use only publicly available datasets and pretrained models and follow their respective licenses and terms of use. Specifically, we use MNIST (GNU license), CIFAR-10, IMDB (subject to IMDb's Terms of Use), CNN/DailyMail (Apache License 2.0), LFW, and ImageNet. Where an explicit open-source license is not provided by the dataset distributor, we follow the corresponding dataset-specific access and usage terms. The pretrained GPT-Neo and DistilBERT checkpoints are obtained from their publicly available HuggingFace repositories and used according to their respective model licenses. All datasets, models, and software packages are credited through their original references in the paper. The complete codebase and instructions for reproducing all experiments are available on GitHub[2] under the MIT license. Publicly available implementations were used to reimplement all baselines, except for FedAvg and Min. Leakage, which we implemented directly using Flower Library [136]. We follow the recommended hyperparameters for baselines, setting the compression ratio of SoteriaFL to 5% and the graph degree of Shatter to 4.

Experiments were run on a workstation with four NVIDIA RTX A6000 GPUs (48 GB each), dual AMD EPYC 7513 32-core CPUs, and 512 GB RAM.

F Additional Experiments and Analysis

This section presents complementary experiments and empirical validations that reinforce the theoretical claims made in the main paper. We inspect the distributional properties of model weights alongside the Gaussian example in our privacy analysis, evaluate the scalability of ERIS (with and without DSC) through distribution time comparisons, and further assess its robustness against data reconstruction attacks. Additionally, we provide detailed utility–privacy trade-off results under both IID and non-IID settings, and with unbiased and biased gradient estimators across multiple datasets and training configurations.

F.1 Empirical Validation and Operational Estimation of $C_{\max}$

Remark D.1 provides a Gaussian-channel instantiation of the per-coordinate quantity entering $C_{\max}$. As a distributional sanity check, we first inspect model-weight histograms during training. Figure 6 reports these distributions for three representative models—DistilBERT on IMDB, ResNet-9 on CIFAR-10, and LeNet-5 on MNIST. Each 3-D panel shows weight value (x-axis), training round (depth), and frequency (z-axis). Across datasets and training rounds, the distributions remain approximately Gaussian-shaped, with relatively stable variance and standard deviation below 0.2 throughout. These cross-coordinate histograms do not establish the conditional Gaussian or independent-noise assumptions of Remark D.1; the quantitative plug-in estimates below are obtained from repeated stochastic updates under that model.

Empirical audit of $C_{\max}$. We apply the estimator from Appendix D.1 to MNIST with LeNet-5, using $K = 50$ clients, 64 samples per client, $T = 250$ training rounds, and all $n = 61{,}706$ model parameters. We audit the training states immediately before rounds 1, 125, and 250. At each checkpoint, we sample $M = 40$ independent client datasets and compute $R = 20$ stochastic minibatch updates per dataset with batch size 16.

Across the resulting $3n = 185{,}118$ audited coordinate–round pairs, the estimated per-coordinate information values are

$$\operatorname{median}\left(\widehat{C}_{i,t}\right) = 0.1774 \text{ bits}, \qquad Q_{0.99}\left(\widehat{C}_{i,t}\right) = 0.3315 \text{ bits}, \qquad \widehat{C}_{\max}^{\text{obs}} = 0.5172 \text{ bits}.$$

Here, $\widehat{C}_{\max}^{\text{obs}}$ denotes the largest value observed in the audit rather than the supremum over every possible dataset and training history.

[2] https://github.com/dariofenoglio98/eris

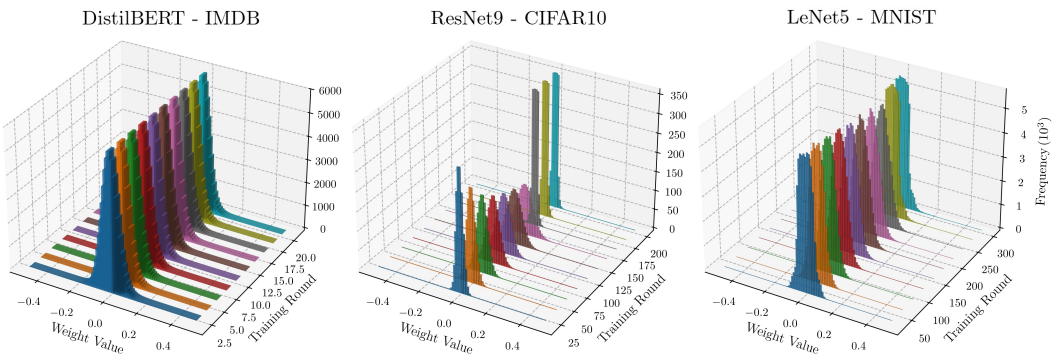

Figure 6: Weight-value histograms across coordinates over training rounds for DistilBERT, ResNet-9, and LeNet-5. Each 3-D plot shows the distribution of weight values (horizontal axis) over time (depth axis), with frequency on the vertical axis. In all cases, the distributions remain approximately Gaussian-shaped, with empirical standard deviation below 0.2 throughout training.

For the ERIS configuration evaluated in this setting, $A = 50$ and $\omega = 250$, corresponding to a retention probability $p = \frac{1}{1+\omega} = \frac{1}{251}$. Hence, the expected number of exposed coordinate–round pairs appearing in Equation 7 is

$$\frac{nTp}{A} = \frac{61{,}706 \cdot 250}{50 \cdot 251} \approx 1{,}229.20.$$

Under the independent-noise Gaussian model of Remark D.1, using the largest observed per-coordinate value gives the empirical plug-in evaluation

$$\frac{nTp}{A}\,\widehat{C}_{\max}^{\mathrm{obs}} \approx 635.7 \text{ bits}.$$

This model-based quantity concerns information about the complete 64-sample client dataset. The plug-in composition assigns the largest observed value to every exposed coordinate in every round; however, $\widehat{C}_{\max}^{\mathrm{obs}}$ itself is empirical and does not certify the worst-case value of $C_{\max}$ outside the audited states. The substantially smaller median and 99th-percentile estimates further show that the plug-in composition is dominated by the most informative audited coordinates.

Finally, for a common $C_{\max}$, relative to observing every coordinate at every round, ERIS reduces the corresponding mutual-information bound by $\frac{A}{p} = 12{,}550\times$. The audit therefore provides a model-based plug-in scale, rather than a certified absolute leakage bound. For a common $C_{\max}$, FSA and DSC scale the corresponding upper bound by p/A; the absolute leakage remains determined by the underlying model, data distribution, and training dynamics.

F.2 Scalability and Efficiency of ERIS

To evaluate the scalability and communication efficiency of ERIS, we provide both a theoretical analysis of model distribution time and empirical comparisons with existing FL frameworks.

F.2.1 Theoretical Analysis of Model Distribution Time

We begin by quantifying the minimum time required to distribute models in a single training round under various FL setups. Here, the distribution time refers to the time needed for: (i) clients to transmit their local models to the aggregation parties (either a central server or a set of aggregators), and (ii) all clients to receive the updated global model. For clarity, we assume full client participation in each round; however, the same analysis readily extends to partial participation scenarios by adjusting the number of active clients accordingly.

Single-server Federated Learning. In traditional centralized FL, we consider a single server and K clients. Let u_s and d_s denote the server's upload and download rates, and let u_k, d_k be the k-th client's upload and download rates, respectively. Assume the model has n parameters and each is represented as a 32-bit float, yielding a total model size of $b \approx 32 \cdot n$ bits. The distribution time in a single training round is governed by the following observations:

- The server must collect K local models, each of size b bits, resulting in a total inbound traffic of $K \cdot b$ bits, received at a download rate d_s.
- Each client k uploads its local model at an individual rate u_k. The server cannot complete the upload phase until the slowest client—i.e., the one with the lowest u_k—has finished its transmission.
- Once all local models are received, the server performs aggregation and then broadcasts the aggregated global model back to all K clients. This requires transmitting another $K \cdot b$ bits at the server's upload rate u_s.
- Model distribution concludes when every client has received the global model. This process is bounded by the client with the lowest download rate d_k, as it determines the last completed transfer.

Putting all these observations together, we derive the minimum distribution time in a single training round for a centralized FL setup without compression such as FedAvg, denoted by D_{FedAvg}.

$$D_{FedAvg} \geq \max\left\{\frac{K \cdot b}{d_s}, \frac{b}{\min\{u_1, \ldots, u_K\}}\right\} + \max\left\{\frac{K \cdot b}{u_s}, \frac{b}{\min\{d_1, \ldots, d_K\}}\right\} \tag{52}$$

Here, the first term captures the server's time to receive all local model uploads and the slowest client's upload time, while the second term captures the server's model broadcast time and the slowest client's download time.

To reduce distribution time, several FL methods focus on minimizing the volume of transmitted data per round, i.e., decreasing the effective model size b. For example, ***PriPrune*** [41] applies structured pruning to eliminate a fraction p of the model parameters before transmission, reducing the transmitted size to $b' \leq 32 \cdot (1-p) \cdot n$ bits. Similarly, ***SoteriaFL*** [60] compresses gradients using a shifting operator controlled by a compression factor ω, leading to $b' \leq 32 \cdot \frac{1}{\omega+1} \cdot n$ bits. In principle, ERIS could also compress both upload and download communication, e.g., by coordinating compression masks across clients so that the aggregated update remains compressed. However, for a fair comparison with baselines such as SoteriaFL and PriPrune, we adopt a worst-case communication setting in which clients use independent compression masks. As a result, upload communication is compressed, but the aggregated model shards returned by the aggregators are treated as uncompressed in the download phase.

Federated Shard Aggregation. We now extend the analysis to ERIS, where FSA distributes aggregation across $A \leq K$ aggregator nodes. Unlike centralized schemes, each client update is split into A disjoint shards, and each aggregator handles only its assigned shard. Let b' denote the transmitted update size: for FSA without DSC, $b' = b$, while with DSC, $b' \leq 32 \cdot \frac{1}{\omega+1} \cdot n$. To estimate the minimum distribution time in a single training round under ERIS, we consider the following:

- Each aggregator must collect $K-1$ update shards from the clients, excluding its own, amounting to $(K-1) \cdot \frac{b'}{A}$ bits received per aggregator at download rate d_k. The aggregation process cannot proceed before the slowest aggregator receives all required shards.
- Each client k uploads one shard to each aggregator, sending a total of $b\prime$ bits. If the client is not serving as an aggregator (worst case), it must upload the entire set of A shards at an upload rate u_k. The aggregation step is gated by the client with the lowest upload rate.
- Once aggregation is complete, each aggregator redistributes its updated model shard to all K clients. This amounts to sending $(K-1) \cdot \frac{b}{A}$ bits at upload rate u_k, constrained by the aggregator with the slowest upload speed.
- Full model reconstruction occurs only after each client receives one shard from every aggregator. In the worst case, a non-aggregator client must download the complete updated model, i.e., b bits, at rate d_k. The distribution concludes when the slowest client completes this transfer.

Putting all these observations together, we derive the minimum distribution time in a single training round for ERIS, denoted by D_{ERIS}:

$$D_{\text{ERIS}} \geq \max\left\{\frac{(K-1)b'}{A \cdot \min\{d_1,\ldots,d_A\}}, \frac{b'}{\min\{u_1,\ldots,u_K\}}\right\} + \max\left\{\frac{(K-1)b}{A \cdot \min\{u_1,\ldots,u_A\}}, \frac{b}{\min\{d_1,\ldots,d_K\}}\right\} \tag{53}$$

Decentralized baselines. While FSA distributes aggregation across multiple nodes, it differs from prior decentralized approaches because every round still recovers the full collaborative update after reassembly. We compare against two representative decentralized baselines.

Ako [51] distributes gradient computations by splitting each model into v disjoint partitions and randomly assigning them to worker nodes. However, this approach differs fundamentally from ERIS: in Ako, not all clients receive the full model in each round, which may hinder convergence to the standard FedAvg solution. Furthermore, in each round, a client uploads and receives K partitions, equivalent to the full model size, resulting in substantial bandwidth usage. We can estimate the minimum distribution time for Ako, denoted by D_{Ako}, using a similar worst-case analysis:

$$D_{Ako} \geq \max\left\{\frac{b}{\min\{d_1,\ldots,d_K\}}, \frac{b}{\min\{u_1,\ldots,u_K\}}\right\} \tag{54}$$

Shatter [78] is also a privacy preserving distributed learning framework. In Shatter, each round consists of three steps. In the first step, each node updates its local model and divides the result in l chunks. Each client (real node) runs l virtual nodes. The virtual nodes form an overlay network over which model parameter updates are multicast with a gossiping protocol. Once received r updates for each of the l virtual nodes running on a real node, the last step consists of the virtual nodes forwarding the received updates to the real node to perform the aggregation. Notice that in this setup, not all clients will receive all model updates, so the communication overhead is reduced at the cost of slower global model convergence. The model distribution occurs in the second step through a multicast among the virtual nodes. The model distribution cannot finish before each real node (via its virtual nodes) has finished uploading its model chunks, and has finished downloading model updates from r other clients. Then, we need to account for the time to upload all the model updates with the total upload capacity being the sum of all individual node upload rates. Therefore, we can estimate the minimum distribution time for Shatter, denoted by D_{Shatter}:

$$D_{\text{Shatter}} \geq \max\left\{\frac{b}{\min\{u_1,\ldots,u_K\}}, \frac{r \cdot b}{\min\{d_1,\ldots,d_K\}}, \frac{r \cdot b}{\sum_{i=1}^{K} u_i}\right\} \tag{55}$$

F.2.2 Numerical Results

Effect of Number of Clients and Model Size on Distribution Time. Figure 7 compares the minimum distribution time per training round for ERIS (with and without DSC) and other FL frameworks under varying numbers of aggregators and model sizes. We assume homogeneous network conditions across all nodes, with upload and download rates fixed at 100 Mbps. For the baselines, we apply a pruning rate of 0.3 for PriPrune, a compression ratio of $1/(\omega+1) = 0.05$ for SoteriaFL and ERIS with DSC, and the overlay topology (i.e., graph degree) for Shatter forms a 4-regular graph. Note that $1/(\omega+1) = 0.05$ (with $\omega = 18$) corresponds to the least aggressive compression used in our experiments, chosen for MNIST to preserve model utility. In other settings, such as IMDB, we adopt much stronger compression (e.g., $1/(\omega+1) = 0.00012$), further lowering communication overhead without harming performance. Thus, the results in Figure 7 represent a conservative estimate of ERIS's efficiency.

On the left, Figure 7 shows how distribution time scales with the number of participating clients: while all methods experience linear growth (except Ako and Shatter, which always exchange with a fixed number of neighbors), ERIS benefits from increased decentralization—achieving lower distribution times as the number of aggregators A increases. In the worst-case setting ($A=2$), ERIS still achieves a 4× speedup over FedAvg and a 2× improvement over SoteriaFL. When $A=50$, these gains rise dramatically to 105× and 55×, respectively, underscoring the scalability advantages of decentralized aggregation. Maximum efficiency is achieved when the number of aggregators matches the number

of clients, maximizing parallelism and evenly distributing the communication load. Notably, Ako and Shatter remain constant with respect to the number of clients, as their communication pattern does not involve distributing the full model to all participants—at the expense of model consistency and convergence guarantees. On the right, Figure 7 examines the impact of increasing model size with 50 clients. The results highlight the communication efficiency of decentralized approaches, especially ERIS, which outperforms traditional centralized frameworks as model size grows. This confirms the practical benefits of combining decentralized aggregation with compression.

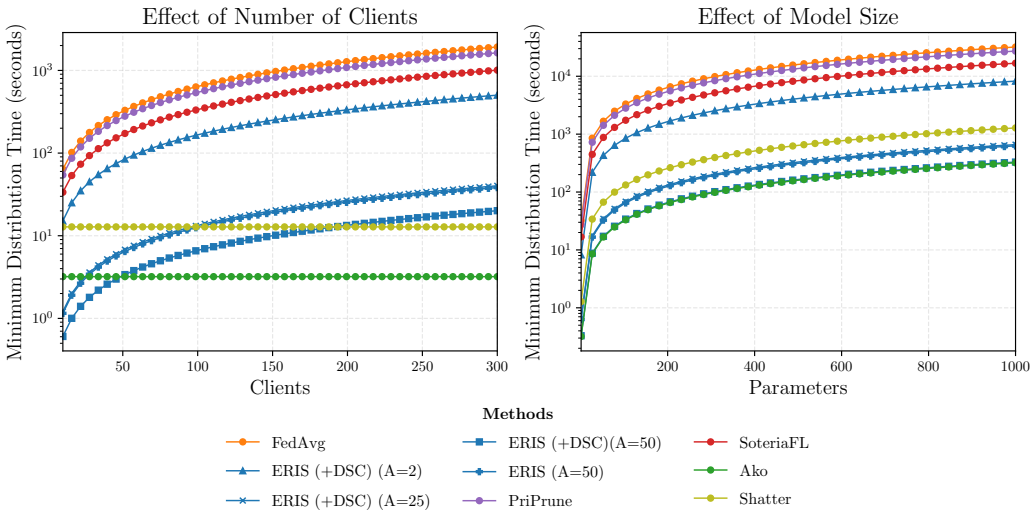

Figure 7: Minimum distribution time for a single training round for FedAvg, PriPrune, SoteriaFL, Ako, Shatter, and ERIS. The figure shows the minimum distribution time for a single round with $M = 320$ Mbit on a logarithmic scale (**left**), and the minimum distribution time for a single round with 50 training clients as the model size increases on a logarithmic scale (**right**).

Effect of Transmission Rate. We further evaluate the sensitivity of ERIS to degraded communication links by varying the effective transmission rate while measuring the minimum per-round distribution time. This analysis complements Figure 7, which assumes fixed homogeneous bandwidth, by explicitly testing slower network conditions. As shown in Figure 8, all methods exhibit the expected inverse relationship between transmission rate and distribution time. However, ERIS consistently remains more communication-efficient across the full range of rates. This advantage comes from two complementary properties: each communication link transmits only a model shard rather than a full update, and communication is parallelized across multiple aggregators. Increasing the number of aggregators further reduces latency by distributing the communication load more evenly, while DSC provides an additional reduction in the transmitted payload. Notably, even under substantially reduced transmission rates, ERIS with sufficiently many aggregators remains faster than the strongest baselines operating under more favorable bandwidth conditions.

Communication efficiency. In addition to the main paper, we provide a detailed communication comparison across a larger set of FL and decentralized FL (DFL) baselines: FedAvg, Shatter, Ako, Q-DPSGD-1, PriPrune (with multiple pruning rates), SoteriaFL, and ERIS variants. Tables 4–5 report, for each dataset/model pair, the compression ratio, per-client upload/download volume, total per-round communication per client, and the distribution time per round (20MB/s bandwidth), under the same experimental settings as in Table 1.

Among decentralized methods, Shatter and Ako rely on model partitioning, whereas Q-DPSGD-1 reduces communication via quantization. We additionally include PriPrune at three pruning rates and SoteriaFL at 5% compression. For ERIS with DSC, we report (i) a matched-compression setting with SoteriaFL (same ratio) to isolate the benefit of FSA and (ii) the more aggressive compression regime used in our main experiments. Overall, Tables 4–5 show that, for a fixed compression ratio, ERIS with DSC consistently achieves substantially lower distribution time than both centralized and

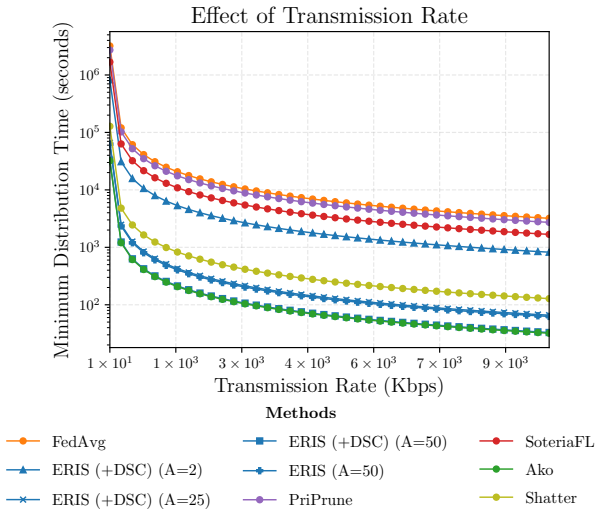

Figure 8: Effect of transmission rate on minimum per-round distribution time for FedAvg, PriPrune, SoteriaFL, Ako, Shatter, and ERIS. ERIS remains faster across transmission rates due to shard-based communication and distributed aggregation, with further gains from larger A and DSC.

Table 4: Direct communication comparison (per-client) across FL/DFL baselines.

Method	**CNN/Daily Mail — GPT-Neo (1.3B)**					**IMDB — DistilBERT (69M)**				
	Comp.	**Upload**	**Down.**	**Tot.**	**Time**	**Comp.**	**Upload**	**Down.**	**Tot.**	**Time**
FedAvg	100%	5.2 GB	5.2 GB	10.4 GB	5200.0 s	100%	268.0 MB	268.0 MB	536.0 MB	670.0 s
Shatter	100%	5.2 GB	5.2 GB	36.4 GB	780.0 s	100%	268.0 MB	268.0 MB	2.41 GB	53.6 s
Ako (p=5)	100%	9.36 GB	9.36 GB	18.72 GB	936.0 s	100%	1.29 GB	1.29 GB	2.57 GB	128.64 s
Q-DPSGD-1 (K_n=0.4)	18.8%	3.51 GB	3.51 GB	37.44 GB	351.0 s	18.8%	482.4 MB	482.4 MB	5.15 GB	48.24 s
PriPrune (0.01)	90%	4.68 GB	5.2 GB	9.88 GB	4940.0 s	90%	241.2 MB	268.0 MB	509.2 MB	636.5 s
PriPrune (0.05)	80%	4.16 GB	5.2 GB	9.36 GB	4680.0 s	80%	214.4 MB	268.0 MB	482.4 MB	603.0 s
PriPrune (0.1)	70%	3.64 GB	5.2 GB	8.84 GB	4420.0 s	70%	187.6 MB	268.0 MB	455.6 MB	569.5 s
SoteriaFL	5%	260.0 MB	5.2 GB	5.46 GB	2730.0 s	5%	13.4 MB	268.0 MB	281.4 MB	351.75 s
ERIS	100%	4.68 GB	4.68 GB	9.36 GB	468.0 s	100%	257.28 MB	257.28 MB	514.56 MB	25.73 s
ERIS (+DSC with $\omega_{\text{SoteriaFL}}$)	5%	234.0 MB	4.68 GB	4.91 GB	245.7 s	5%	12.86 MB	257.28 MB	270.14 MB	13.51 s
ERIS (+DSC)	1%	46.8 MB	4.68 GB	4.73 GB	236.34 s	0.012%	30.87 KB	257.28 MB	257.31 MB	12.87 s

decentralized baselines. In addition, ERIS with DSC enables markedly more aggressive compression while preserving accuracy, as reported in the main text.

F.3 Effect of Distributed Shifted Compression on Model Utility

This section provides additional results complementing the analysis in Paragraph 4.2. Figure 9 illustrates the impact of increasing the compression constant ω on test accuracy for CIFAR-10, under varying numbers of local training samples per client. We observe that up to $\omega = 340$—which corresponds to a compression rate of approximately 0.29%—test accuracy remains statistically unchanged. This indicates that the communication cost can be substantially reduced, sharing only 0.29% of gradients per client, without degrading performance. However, beyond this threshold, the aggressive compression starts discarding critical information, leading to compromised model convergence. As expected, further increasing ω results in progressively lower accuracy, highlighting a clear trade-off between compression strength and model utility.

F.4 Effect of Local Data Size on the Utility–Privacy Trade-off

The experiments reported in Table 1 and in Appendix F.7–F.10 focus on a low-data regime, where each client is assigned only from 4 to 128 local training samples. This setting was chosen deliberately to study privacy leakage under conditions that promote memorization and overfitting, which are

Table 5: Direct communication comparison (per-client) across FL/DFL baselines (continued).

Method	CIFAR-10 — ResNet9 (1.65M)					MNIST — LeNet5 (62K)				
	Comp.	Upload	Down.	Tot.	Time	Comp.	Upload	Down.	Tot.	Time
FedAvg	100%	6.6 MB	6.6 MB	13.2 MB	33.0 s	100%	248.0 KB	248.0 KB	496.0 KB	1.24 s
Shatter	100%	6.6 MB	6.6 MB	59.4 MB	1.32 s	100%	248.0 KB	248.0 KB	2.23 MB	0.05 s
Ako (p=5)	100%	64.68 MB	64.68 MB	129.36 MB	6.47 s	100%	2.43 MB	2.43 MB	4.86 MB	0.24 s
Q-DPSGD-1 (K_n=0.4)	18.8%	24.25 MB	24.25 MB	258.72 MB	2.43 s	18.8%	911.4 KB	911.4 KB	9.72 MB	0.09 s
PriPrune (0.01)	99%	6.53 MB	6.6 MB	13.13 MB	32.84 s	99%	245.52 KB	248.0 KB	493.52 KB	1.23 s
PriPrune (0.05)	95%	6.27 MB	6.6 MB	12.87 MB	32.17 s	95%	235.6 KB	248.0 KB	483.6 KB	1.21 s
PriPrune (0.1)	90%	5.94 MB	6.6 MB	12.54 MB	31.35 s	90%	223.2 KB	248.0 KB	471.2 KB	1.18 s
SoteriaFL	5%	330.0 KB	6.6 MB	6.93 MB	17.32 s	5%	12.4 KB	248.0 KB	260.4 KB	0.65 s
ERIS	100%	6.47 MB	6.47 MB	12.94 MB	0.65 s	100%	243.04 KB	243.04 KB	486.08 KB	0.02 s
ERIS (+DSC with $\omega_{\text{SoteriaFL}}$)	5%	323.4 KB	6.47 MB	6.79 MB	0.34 s	5%	12.15 KB	243.04 KB	255.19 KB	0.01 s
ERIS(+DSC)	0.6%	38.81 KB	6.47 MB	6.51 MB	0.33 s	3.3%	8.02 KB	243.04 KB	251.06 KB	0.01 s

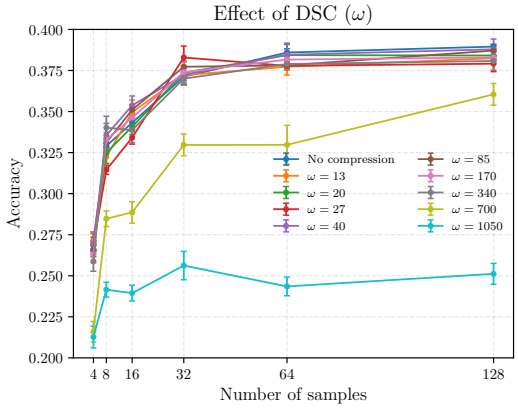

Figure 9: Effect of DSC on CIFAR-10, varying ω across different local training sample sizes.

known to amplify membership inference risk. As a result, these experiments are not intended as full-data benchmarks, but rather as controlled privacy–utility evaluations under limited local data.

To verify that the trends observed in this regime are not an artifact of unstable training or underfitting, we extend the CIFAR-10 analysis to larger local data sizes. Table 6 reports test accuracy and MIA accuracy for 64, 128, 256, and 415 samples per client. The results show a stable and monotonic trend across all methods: increasing the number of local training samples consistently improves utility while reducing privacy leakage. This confirms that the original low-data setting reflects a deliberate operating regime for privacy auditing, rather than a failure of optimization.

Importantly, ERIS with DSC continues to closely track FedAvg in utility across all larger-data settings while maintaining substantially lower MIA leakage. For example, at 415 samples per client, FedAvg reaches 63.3% accuracy, while ERIS attains 62.9%, but reduces MIA accuracy from 82.2% to 68.4%. This indicates that the privacy advantages of ERIS persist even when the local data regime becomes substantially more favorable for utility. We note that the 415-sample setting corresponds to approximately 21k training samples used for federated optimization, while the remaining samples are reserved for validation and privacy auditing. This split is necessary to ensure a consistent and controlled evaluation of membership inference across methods.

F.5 Robustness to Aggregator and Link Failures

We evaluate the robustness of ERIS under two failure modes: aggregator dropout and client–aggregator link failures. In both cases, failures remove a subset of model shards from the global update in a given round. However, because each shard corresponds only to a disjoint part of the model, failures do not invalidate the update; they primarily reduce its effective magnitude and therefore slow convergence.

Aggregator dropout. We first consider aggregator unavailability during training. At each round, a fixed proportion of aggregators is randomly deactivated and resampled independently across rounds,

Table 6: CIFAR-10 in a larger-data regime: accuracy and MIA accuracy for increasing numbers of training samples per client. Results confirm a stable utility–privacy trend beyond the low-data regime.

	64 samples		128 samples		256 samples		415 samples	
Method	Acc. (↑)	MIA Acc. (↓)	Acc. (↑)	MIA Acc. (↓)	Acc. (↑)	MIA Acc. (↓)	Acc. (↑)	MIA Acc. (↓)
FedAvg	46.4 ± 0.4	93.9 ± 0.7	50.0 ± 0.6	89.9 ± 0.3	56.6 ± 0.5	85.9 ± 0.3	63.3 ± 0.4	82.2 ± 0.1
FedAvg+LDP	27.2 ± 0.4	58.1 ± 0.2	29.9 ± 0.2	56.4 ± 0.4	33.2 ± 0.3	54.5 ± 0.5	36.3 ± 0.2	54.0 ± 0.8
PriPrune	45.0 ± 0.3	74.4 ± 0.8	49.0 ± 0.5	74.1 ± 1.0	56.1 ± 0.5	78.1 ± 0.8	63.1 ± 0.3	75.8 ± 0.3
ERIS (+DSC)	46.2 ± 0.9	71.1 ± 0.8	49.7 ± 0.7	69.8 ± 0.7	56.0 ± 0.7	68.6 ± 0.7	62.9 ± 0.4	68.4 ± 0.5
Min. Leakage	46.4 ± 0.3	72.6 ± 0.9	49.9 ± 0.7	71.3 ± 0.8	57.0 ± 0.2	69.9 ± 0.5	63.2 ± 0.3	67.9 ± 0.6

so their corresponding model shards are temporarily excluded from the global update. Consequently, coordinates omitted in one round can be updated again in subsequent rounds when their aggregator becomes available. Figure 10 reports the effect of increasing dropout rates on both (i) test accuracy and (ii) the best validation round at which the model reaches its peak performance (i.e., the minimum validation loss). The results show that ERIS maintains nearly constant test accuracy up to a dropout rate of 70%. The right plot explains this behavior: as dropout increases, the best validation round shifts steadily toward the 200-round training cap, indicating a convergence slowdown rather than algorithmic instability. Once the slowdown becomes too large, the model no longer reaches the same optimum within the fixed training budget, and accuracy drops.

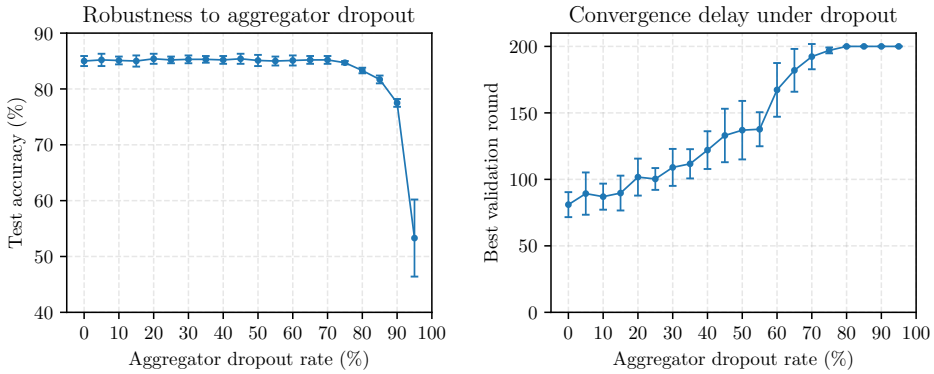

Figure 10: Robustness of ERIS to aggregator dropout. Left: test accuracy remains nearly constant up to a 70% dropout rate. Right: convergence slows as dropout increases, eventually hitting the 200-round cap, which explains the accuracy drop beyond 70%.

Client–aggregator link failures. We further evaluate random failures of individual client–aggregator communication links. When such a link fails, only the corresponding shard contribution is lost for that round. Figure 11 shows a similar pattern to aggregator dropout: accuracy remains nearly unchanged up to moderate failure rates, while the best validation round progressively approaches the 200-round cap. In particular, accuracy remains close to the no-failure setting up to roughly 50% link failures, whereas larger failure rates mainly hurt performance because training cannot fully converge within the fixed round budget. This contrasts with centralized FL, where a failed client–server link discards the entire client update; in ERIS, a failed link removes only one shard contribution.

Together, these results show that ERIS degrades gracefully under both node- and edge-level failures. The system remains stable as long as a sufficient fraction of shards is aggregated, and performance degradation is primarily caused by convergence delay under a fixed round cap rather than by instability of the sharded aggregation mechanism. Thus, the robustness to random aggregator dropout or link failures should not be interpreted as indicating that the missing shards or shard contributions are unnecessary: all shards remain required to recover the exact centralized update at each round, while random failures only postpone their contribution to the optimization process.

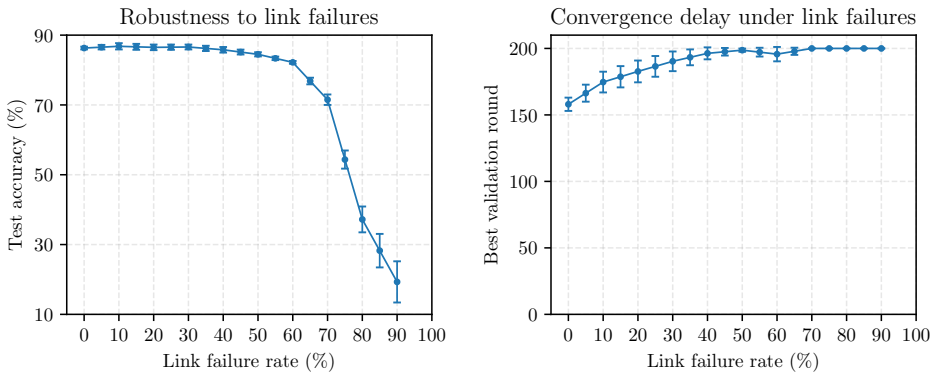

Figure 11: Robustness of ERIS to client–aggregator link failures. Left: accuracy remains close to the no-failure setting up to 50% link failures. Right: convergence slows as link failures increase, with the best round approaching the 200-round cap, which explains the accuracy drop at higher failure rates.

F.6 Data Reconstruction Attacks

To further assess the privacy guarantees of ERIS, we evaluate its resilience to Data Reconstruction Attacks (DRAs), which represent one of the most severe privacy threats in FL. To favour the attacker and stress-test our approach, we consider the uncommon but worst-case scenario where each client performs gradient descent with a mini-batch of size 1 and transmits the resulting gradient—which can be intercepted by an eavesdropper or a compromised aggregator/server. Therefore, we assume the adversary has white-box access to the client gradient.

Given this gradient, reconstruction methods such as DLG [12], iDLG [13], and ROG [8] aim to recover the original training sample by optimizing candidate inputs to match the leaked gradient. Unlike earlier gradient-matching attacks, ROG projects the unknown image into a low-dimensional latent space (e.g., via bicubic downsampling or an autoencoder) and optimizes that compact representation so that the decoded image's gradients align with the leaked gradient, before applying a learned enhancement module to obtain perceptually faithful reconstructions. In our experiments, a dedicated enhancement decoder was trained for each dataset using a hold-out set. GGL instead constrains the reconstruction to the latent space of a pretrained generative model and incorporates an adaptive gradient-matching objective that accounts for the transformation induced by privacy defenses. This makes GGL particularly suitable for auditing degraded gradients, such as those produced by compression, sparsification, clipping, or additive noise.

Figure 12 reports the reconstruction quality, measured via the LPIPS score [139], as a function of the percentage of model parameters available to the attacker. The x-axis is plotted on a non-linear scale to improve readability in the low-percentage regime. The results are averaged over 200 reconstructed samples and tested across three datasets: MNIST, CIFAR-10, and LFW. The findings show that in the full-gradient setting (e.g., FedAvg), all DRAs can almost perfectly reconstruct the original image. However, as the proportion of accessible gradients decreases, the reconstruction quality of DLG and iDLG degrades significantly, with LPIPS scores approaching the baseline of random images when only 90% of the parameters are visible. Remarkably, even in the least favourable configuration of ERIS—with only two aggregators—the system already provides sufficient obfuscation to render reconstruction attacks ineffective, as highlighted by the shaded regions in the figure. A different pattern emerges for ROG, which tends to maintain higher reconstruction quality. Closer inspection of MNIST and LFW examples, however, reveals that this apparent advantage stems primarily from the trained enhancement decoder. This module effectively biases reconstructions toward the training distribution, thereby inflating similarity scores. In fact, even when random gradients are passed through the decoder (purple dashed line), the outputs achieve LPIPS values lower than random images, underscoring that the improvement reflects postprocessing artefacts. However, once 25% or fewer gradients are accessible, the reconstructed outputs are severely distorted and no longer capture any semantically meaningful features of the original data.

To further characterize the privacy guarantees under stronger adaptive reconstruction settings, we complement the previous analysis with a focused evaluation of ROG and GGL across all implemented baselines and standalone compression mechanisms (Table 7). The results confirm that compression

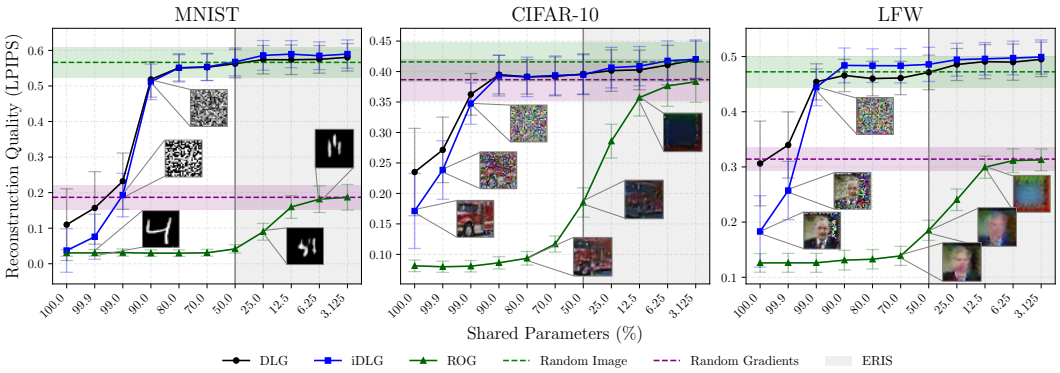

Figure 12: Reconstruction quality under DLG, iDLG, and ROG attacks as a function of the percentage of model parameters available to the attacker. The LPIPS score (higher is better) is averaged over 200 samples. The x-axis is plotted on a non-linear scale for improved clarity of the low-percentage regime. Shaded regions highlight the obfuscation achieved by ERIS, which renders reconstruction attacks ineffective even in its weakest configuration (two aggregators and no compression).

alone is insufficient: both QSGD and uniform quantization degrade reconstruction quality only at aggressive rates (e.g., $s = 4$), while Top-k sparsification becomes effective only at extreme sparsity levels (0.98–0.99), where utility is severely compromised. Similarly, differentially private training via DP-SGD shows a clear trade-off between privacy and utility: with mild clipping (clip $= 10$) and low noise ($\sigma = 10^{-4}$), reconstructions remain close to FedAvg, whereas stronger noise or tighter clipping substantially degrades image quality but at the expense of model performance. PriPrune exhibits a comparable pattern, with higher pruning probabilities providing stronger obfuscation but relying on increasingly aggressive gradient removal. Finally, ERIS provides robust protection even in its least favorable setting ($A = 2$ aggregators), and its privacy guarantees strengthen as the number of aggregators increases or, when combined with DSC, as the compression strength ω increases. Notably, the strongest ERIS configurations approach the random-gradient baseline under ROG and substantially increase LPIPS under GGL. These results indicate that the combination of model partitioning and DSC effectively limits the information available to the attacker, including in the presence of adaptive generative reconstruction, while preserving utility.

F.7 Balancing Utility and Privacy - IID Setting

This section provides detailed numerical results supporting the analysis in Paragraph 4.2. Specifically, we report test accuracy and Membership Inference Attack (MIA) accuracy for all evaluated methods across multiple datasets and varying local training sizes in IID setting. Tables 8–14 report results for CNN/DailyMail, IMDB, CIFAR-10, and MNIST. IMDB, CIFAR-10, and MNIST are evaluated across 4–128 client training samples, while CNN/DailyMail is limited to 16–128 samples, as overfitting saturates already at 16. Part of these values serve as the coordinates for Figure 3, which visualizes the utility–privacy trade-off achieved by ERIS and baselines. Importantly, the tables extend beyond the conditions illustrated in the figure by covering a wider set of hyperparameter configurations—namely, additional pruning rates (p) and privacy budgets (ϵ) for LDP-based methods.

Notably, these results (Figure 3 and Tables 8–14) show two clear trends. First, a smaller amount of local data leads all methods to lower task accuracy and higher MIA accuracy, reflecting the stronger overfitting in this regime. Second, especially under low-data conditions, ERIS (with and without DSC) consistently delivers markedly better privacy preservation while retaining competitive accuracy. For instance, on CNN/DailyMail with 16 samples, for the same ROUGE-1 score as FedAvg, ERIS reduces MIA accuracy from 100% to 77.7%; on IMDB with 4 samples, it lowers MIA accuracy from 82.9% to 65.2%, closely approaching the unattainable upper-bound of 64.4%. These findings highlight ERIS's robustness across data modalities and model capacities.

F.8 Balancing Utility and Privacy - non-IID Setting

We further evaluate the utility–privacy trade-off under non-IID client data, using K=10 clients for IMDB and K=25 for CIFAR-10 and MNIST, with a Dirichlet partition with α=0.5 for IMDB and

Table 7: Reconstruction quality under ROG and GGL attacks across privacy-preserving mechanisms and compression techniques on CIFAR-10 and ImageNet, respectively. For LPIPS, higher values indicate stronger defenses (↑).

Method	**ROG**	**GGL**
FedAvg	0.193 ± 0.059	0.666 ± 0.053
QSGD ($s = 16$)	0.209 ± 0.060	0.670 ± 0.057
QSGD ($s = 8$)	0.250 ± 0.065	0.671 ± 0.058
QSGD ($s = 4$)	0.343 ± 0.074	0.676 ± 0.055
Uniform Quantization ($s = 16$)	0.243 ± 0.066	0.676 ± 0.055
Uniform Quantization ($s = 8$)	0.302 ± 0.071	0.673 ± 0.064
Uniform Quantization ($s = 4$)	0.403 ± 0.079	0.686 ± 0.053
Top-*k* Sparsification (sparsity=0.90)	0.228 ± 0.063	0.670 ± 0.059
Top-*k* Sparsification (sparsity=0.98)	0.392 ± 0.083	0.674 ± 0.051
Top-*k* Sparsification (sparsity=0.99)	0.456 ± 0.094	0.679 ± 0.061
DP-SGD (clip=10, $\sigma = 10^{-4}$)	0.200 ± 0.062	0.668 ± 0.062
DP-SGD (clip=10, $\sigma = 10^{-3}$)	0.340 ± 0.071	0.671 ± 0.061
DP-SGD (clip=10, $\sigma = 10^{-2}$)	0.498 ± 0.091	0.676 ± 0.051
DP-SGD (clip=1, $\sigma = 10^{-4}$)	0.432 ± 0.094	0.676 ± 0.059
DP-SGD (clip=1, $\sigma = 10^{-3}$)	0.436 ± 0.095	0.682 ± 0.048
DP-SGD (clip=1, $\sigma = 10^{-2}$)	0.472 ± 0.090	0.689 ± 0.042
PriPrune ($p = 10^{-5}$)	0.305 ± 0.075	0.686 ± 0.047
PriPrune ($p = 10^{-3}$)	0.506 ± 0.087	0.688 ± 0.036
PriPrune ($p = 0.1$)	0.569 ± 0.067	0.696 ± 0.035
ERIS ($A = 2$)	0.458 ± 0.081	0.665 ± 0.060
ERIS ($A = 4$)	0.514 ± 0.078	0.675 ± 0.054
ERIS ($A = 8$)	0.546 ± 0.075	0.678 ± 0.056
ERIS (+DSC) ($\omega = 1$, $A = 2$)	0.453 ± 0.081	0.676 ± 0.053
ERIS (+DSC) ($\omega = 4$, $A = 2$)	0.524 ± 0.078	0.673 ± 0.052
ERIS(+DSC) ($\omega = 9$, $A = 2$)	0.547 ± 0.073	0.683 ± 0.048
ERIS (+DSC) ($\omega = 49$, $A = 50$)	0.569 ± 0.073	0.690 ± 0.064
Random Gradients	0.572 ± 0.065	0.812 ± 0.047

α=0.2 for CIFAR-10 and MNIST. Figure 13 illustrates the utility–privacy trade-off across methods and datasets, where the ideal region corresponds to the top-right corner (high accuracy and low privacy leakage). Non-IID distributions generally make convergence more challenging, lowering overall accuracy and increasing variability across clients [140]. Nevertheless, ERIS with DSC remains stable and consistently reduces privacy leakage. For example, on IMDB with 4 local samples, ERIS matches FedAvg in accuracy while reducing MIA accuracy from 91.7% to 80.7%, approaching the ideal upper-bound of not sharing local gradients. On CIFAR-10 and MNIST, ERIS even matches or slightly surpass non-private FedAvg in terms of accuracy, while still offering strong privacy protection. By contrast, privacy-enhancing baselines such as Shatter and FedAvg-LDP struggle to maintain utility, often remaining close to random-guess performance, particularly when models are trained from scratch. Table 15 reports detailed mean accuracy and MIA accuracy values, averaged over varying local sample sizes. Together, these results confirm that the advantages of ERIS extend robustly to heterogeneous data distributions.

F.9 Balancing Utility and Privacy - Biased Gradient Estimator

In this section, we extend our analysis of the utility–privacy trade-off to the biased setting (already adopted for CNN/DailyMail dataset), where each client performs multiple local updates per communication round, thereby introducing bias into the gradient estimator. The training hyperparameters are detailed in Section E.1.

Figure 14 summarizes the performance of ERIS with DSC and several SOTA privacy-preserving baselines in terms of test accuracy and MIA accuracy across different model sizes and local training regimes. As in the main paper (Figure 3), we evaluate datasets with distinct memorization character-

Table 8: Comparison of ERIS with and without compression (ω) against SOTA baselines in terms of ROUGE-1 and MIA accuracy on CNN/DailyMail with 16, 32, 64, and 128 local training samples.

Local Training Size	16 **samples**		32 **samples**		64 **samples**		128 **samples**	
Method	**R-1** (↑)	**MIA Acc.** (↓)	**R-1** (↑)	**MIA Acc.** (↓)	**R-1** (↑)	**MIA Acc.** (↓)	**R-1** (↑)	**MIA Acc.** (↓)
FedAvg	30.37 ± 1.25	100.00 ± 0.00	32.21 ± 1.46	98.75 ± 1.25	34.27 ± 0.65	96.46 ± 0.36	36.04 ± 0.57	96.57 ± 0.90
FedAvg $(10, \delta)$-LDP	25.66 ± 0.86	54.17 ± 5.20	26.26 ± 0.07	50.00 ± 3.31	26.30 ± 0.10	49.33 ± 2.53	25.78 ± 0.10	54.41 ± 1.47
FedAvg $(100, \delta)$-LDP	26.33 ± 0.07	54.17 ± 5.20	26.36 ± 0.10	50.42 ± 2.89	24.90 ± 0.13	48.75 ± 2.25	26.32 ± 0.10	54.31 ± 1.19
SoteriaFL ($\epsilon = 100$)	25.89 ± 0.12	54.13 ± 1.15	26.09 ± 0.07	54.27 ± 5.20	26.01 ± 0.11	50.35 ± 2.89	24.02 ± 0.15	49.15 ± 2.15
SoteriaFL ($\epsilon = 10$)	25.78 ± 0.91	54.02 ± 1.48	25.90 ± 0.36	54.17 ± 5.20	25.90 ± 0.75	50.83 ± 2.60	24.75 ± 0.79	49.54 ± 2.60
PriPrune ($p = 0.1$)	32.21 ± 0.44	95.83 ± 2.89	26.01 ± 0.70	82.92 ± 1.91	30.97 ± 0.55	90.00 ± 1.08	33.90 ± 1.45	88.73 ± 0.90
PriPrune ($p = 0.2$)	29.96 ± 0.94	88.33 ± 2.89	26.61 ± 0.72	76.67 ± 3.61	29.00 ± 0.41	79.38 ± 3.80	31.49 ± 1.89	79.02 ± 0.61
PriPrune ($p = 0.3$)	18.41 ± 15.11	74.17 ± 3.82	21.80 ± 0.67	70.00 ± 3.31	29.00 ± 1.10	70.83 ± 3.44	29.49 ± 1.67	70.39 ± 0.74
Shatter	30.05 ± 1.22	78.73 ± 7.50	30.38 ± 0.59	69.02 ± 2.60	33.04 ± 0.65	66.78 ± 5.12	34.35 ± 0.38	68.13 ± 0.90
ERIS	30.37 ± 1.25	78.50 ± 7.21	32.41 ± 1.46	69.12 ± 2.64	34.04 ± 0.76	66.18 ± 5.00	36.04 ± 0.57	68.16 ± 0.96
ERIS (+DSC) ($\omega_{\text{SoteriaFL}}$)	30.05 ± 0.86	78.33 ± 6.29	32.15 ± 1.75	68.75 ± 3.31	34.12 ± 0.38	63.75 ± 4.38	35.06 ± 1.27	67.65 ± 0.78
ERIS (+DSC) ($\omega \approx 100$)	30.04 ± 0.95	77.73 ± 6.29	31.60 ± 0.95	68.27 ± 3.61	34.14 ± 0.76	64.38 ± 5.12	35.62 ± 0.48	67.84 ± 0.74
Min. Leakage	30.37 ± 1.25	67.83 ± 10.10	32.41 ± 1.46	60.58 ± 1.91	34.27 ± 0.65	54.08 ± 4.77	36.04 ± 0.57	59.61 ± 2.54

Table 9: Comparison of ERIS with and without compression (ω) against SOTA baselines in terms of test and MIA accuracy on IMDB with 4, 8, and 16 local training samples.

Local Training Size	4 **samples**		8 **samples**		16 **samples**	
Method	**Accuracy** (↑)	**MIA Acc.** (↓)	**Accuracy** (↑)	**MIA Acc.** (↓)	**Accuracy** (↑)	**MIA Acc.** (↓)
FedAvg	71.73 ± 4.93	82.93 ± 6.39	79.56 ± 0.61	78.40 ± 3.95	80.52 ± 0.30	66.91 ± 1.81
FedAvg $(\epsilon = 100, \delta)$-LDP	53.79 ± 0.08	55.56 ± 4.91	53.82 ± 0.06	53.33 ± 1.00	53.92 ± 0.11	50.06 ± 2.81
FedAvg $(\epsilon = 10, \delta)$-LDP	53.80 ± 0.03	52.80 ± 5.82	53.81 ± 0.02	50.40 ± 3.12	53.83 ± 0.06	49.89 ± 1.74
SoteriaFL $(\epsilon = 100, \delta)$	53.46 ± 0.15	55.56 ± 6.00	54.73 ± 0.15	54.40 ± 1.96	53.74 ± 0.16	50.30 ± 2.69
SoteriaFL $(\epsilon = 10, \delta)$	53.36 ± 0.29	55.20 ± 5.63	54.01 ± 0.24	51.36 ± 3.48	53.67 ± 0.29	50.11 ± 1.58
PriPrune ($p = 0.1$)	54.40 ± 5.46	80.53 ± 4.59	71.70 ± 2.09	74.72 ± 3.63	77.64 ± 1.44	65.31 ± 2.18
PriPrune ($p = 0.2$)	52.78 ± 2.39	74.67 ± 4.99	58.55 ± 5.30	71.36 ± 2.23	65.33 ± 8.65	62.84 ± 2.41
PriPrune ($p = 0.3$)	53.52 ± 2.79	70.40 ± 3.20	55.92 ± 3.49	65.76 ± 3.63	60.32 ± 5.71	59.82 ± 2.85
Shatter	68.52 ± 4.66	67.52 ± 2.80	74.84 ± 1.88	62.56 ± 3.77	77.91 ± 0.55	54.75 ± 1.97
ERIS	70.15 ± 4.24	67.67 ± 2.89	79.39 ± 0.57	62.45 ± 3.79	80.49 ± 0.33	54.72 ± 1.90
ERIS (+DSC) ($\omega_{\text{SoteriaFL}}$)	71.74 ± 4.94	65.87 ± 3.22	79.55 ± 0.61	60.52 ± 3.00	80.51 ± 0.31	54.56 ± 1.57
ERIS (+DSC) ($\omega \approx 8000$)	71.28 ± 4.74	65.22 ± 2.95	79.28 ± 0.71	60.51 ± 2.50	80.11 ± 0.46	54.12 ± 1.78
Min. Leakage	72.39 ± 1.99	64.44 ± 2.27	79.33 ± 0.75	58.67 ± 2.10	80.68 ± 0.09	53.21 ± 2.53

Table 10: Comparison of ERIS with and without compression (ω) against SOTA baselines in terms of test and MIA accuracy on IMDB with 32, 64, and 128 local training samples.

Local Training Size	32 **samples**		64 **samples**		128 **samples**	
Method	**Accuracy** (↑)	**MIA Acc.** (↓)	**Accuracy** (↑)	**MIA Acc.** (↓)	**Accuracy** (↑)	**MIA Acc.** (↓)
FedAvg	81.62 ± 0.11	63.58 ± 1.47	81.70 ± 0.05	60.54 ± 2.11	82.45 ± 0.18	56.89 ± 0.81
FedAvg $(\epsilon = 100, \delta)$-LDP	54.11 ± 0.15	51.56 ± 1.71	54.50 ± 0.19	50.67 ± 1.60	55.09 ± 0.32	51.26 ± 0.49
FedAvg $(\epsilon = 10, \delta)$-LDP	53.97 ± 0.10	50.02 ± 1.52	54.12 ± 0.12	49.84 ± 0.86	54.30 ± 0.19	50.37 ± 0.92
SoteriaFL $(\epsilon = 100, \delta)$	54.87 ± 0.72	51.94 ± 1.94	55.81 ± 0.62	50.98 ± 1.87	56.08 ± 1.19	51.17 ± 0.52
SoteriaFL $(\epsilon = 10, \delta)$	54.35 ± 0.39	50.10 ± 1.83	54.79 ± 0.35	50.12 ± 0.93	55.28 ± 0.57	50.64 ± 0.92
PriPrune ($p = 0.1$)	79.93 ± 0.23	61.87 ± 1.25	80.34 ± 0.10	59.61 ± 2.11	80.87 ± 0.11	56.11 ± 0.93
PriPrune ($p = 0.2$)	71.48 ± 7.20	59.54 ± 1.71	72.12 ± 1.91	58.21 ± 2.11	77.52 ± 0.55	55.02 ± 1.09
PriPrune ($p = 0.3$)	61.20 ± 7.02	56.43 ± 1.55	62.01 ± 5.91	56.63 ± 1.76	68.95 ± 1.92	54.20 ± 1.18
Shatter	79.34 ± 0.64	53.79 ± 1.56	80.19 ± 0.48	53.68 ± 0.95	80.84 ± 0.19	52.02 ± 1.01
ERIS	81.63 ± 0.12	53.98 ± 1.48	81.70 ± 0.06	53.54 ± 0.93	82.38 ± 0.13	52.08 ± 1.07
ERIS (+DSC) ($\omega_{\text{SoteriaFL}}$)	81.62 ± 0.12	53.81 ± 1.47	81.71 ± 0.05	52.78 ± 1.70	82.45 ± 0.16	51.54 ± 1.27
ERIS (+DSC) ($\omega \approx 8000$)	80.99 ± 0.22	53.44 ± 1.26	81.16 ± 0.19	52.77 ± 1.62	81.59 ± 0.21	51.80 ± 1.07
Min. Leakage	81.58 ± 0.11	52.57 ± 1.27	81.67 ± 0.04	53.05 ± 1.32	82.44 ± 0.09	51.53 ± 1.12

Table 11: Comparison of ERIS with and without compression (ω) against SOTA baselines in terms of test and MIA accuracy on CIFAR-10 with 4, 8, and 16 local training samples.

Local Training Size	4 **samples**		8 **samples**		16 **samples**	
Method	**Accuracy** (↑)	**MIA Acc.** (↓)	**Accuracy** (↑)	**MIA Acc.** (↓)	**Accuracy** (↑)	**MIA Acc.** (↓)
FedAvg	27.12 ± 1.20	84.80 ± 4.59	32.98 ± 0.61	75.84 ± 2.85	34.43 ± 1.04	70.15 ± 1.41
FedAvg ($\epsilon = 10, \delta$)-LDP	10.33 ± 0.53	81.20 ± 3.90	14.93 ± 2.01	72.40 ± 2.60	18.92 ± 1.31	62.55 ± 1.19
FedAvg ($\epsilon = 1, \delta$)-LDP	10.34 ± 0.22	66.50 ± 3.78	10.00 ± 0.00	63.60 ± 2.42	10.00 ± 0.00	56.05 ± 0.49
SoteriaFL ($\epsilon = 10, \delta$)	10.00 ± 0.00	69.87 ± 1.86	10.06 ± 0.12	64.16 ± 1.25	10.85 ± 1.06	58.25 ± 2.10
SoteriaFL ($\epsilon = 1, \delta$)	9.99 ± 0.00	65.67 ± 1.11	10.00 ± 0.00	62.10 ± 1.56	10.00 ± 0.00	53.86 ± 0.67
PriPrune ($p = 0.01$)	13.74 ± 2.05	74.80 ± 2.87	28.42 ± 0.39	75.36 ± 3.60	29.57 ± 0.70	69.82 ± 1.59
PriPrune ($p = 0.05$)	10.09 ± 0.36	67.33 ± 2.63	12.77 ± 2.52	61.68 ± 3.04	11.55 ± 1.80	54.44 ± 1.50
PriPrune ($p = 0.1$)	10.00 ± 0.00	64.27 ± 2.62	10.03 ± 0.04	58.80 ± 2.83	10.00 ± 0.00	52.62 ± 1.37
Shatter	11.47 ± 1.75	77.95 ± 5.63	11.57 ± 1.96	70.49 ± 2.74	12.42 ± 1.65	64.22 ± 1.85
ERIS	27.13 ± 1.19	77.90 ± 5.55	32.90 ± 0.40	70.75 ± 2.70	34.32 ± 0.91	64.21 ± 1.95
ERIS (+DSC) ($\omega_{\text{SoteriaFL}}$)	26.84 ± 0.68	73.86 ± 5.27	32.48 ± 1.43	67.95 ± 1.45	34.08 ± 1.05	59.23 ± 1.43
ERIS (+DSC) ($\omega \approx 170$)	26.31 ± 1.16	71.63 ± 4.28	33.28 ± 1.06	68.48 ± 2.30	34.62 ± 1.42	59.58 ± 2.26
Min. Leakage	27.11 ± 1.17	70.27 ± 4.69	33.11 ± 0.62	65.44 ± 2.43	34.62 ± 0.91	56.87 ± 1.34

Table 12: Comparison of ERIS with and without compression (ω) against SOTA baselines in terms of test and MIA accuracy on CIFAR-10 with 32, 64, and 128 local training samples.

Local Training Size	32 **samples**		64 **samples**		128 **samples**	
Method	**Accuracy** (↑)	**MIA Acc.** (↓)	**Accuracy** (↑)	**MIA Acc.** (↓)	**Accuracy** (↑)	**MIA Acc.** (↓)
FedAvg	37.24 ± 0.41	64.57 ± 0.72	38.50 ± 0.44	59.29 ± 0.79	38.88 ± 0.32	56.11 ± 0.75
FedAvg ($\epsilon = 10, \delta$)-LDP	22.31 ± 1.12	57.14 ± 1.39	23.36 ± 0.85	53.99 ± 0.83	24.13 ± 0.32	52.81 ± 0.49
FedAvg ($\epsilon = 1, \delta$)-LDP	10.00 ± 0.00	57.57 ± 0.26	13.96 ± 1.14	54.42 ± 0.40	19.29 ± 0.37	53.12 ± 0.16
SoteriaFL ($\epsilon = 10, \delta$)	19.68 ± 0.78	55.68 ± 1.09	26.04 ± 0.52	52.94 ± 0.74	26.46 ± 0.25	52.07 ± 0.55
SoteriaFL ($\epsilon = 1, \delta$)	10.00 ± 0.00	54.57 ± 0.50	10.00 ± 0.00	53.28 ± 0.61	12.20 ± 1.25	52.58 ± 0.53
PriPrune ($p = 0.01$)	29.39 ± 0.50	63.73 ± 0.96	28.70 ± 0.51	57.09 ± 0.67	27.99 ± 0.32	53.22 ± 0.70
PriPrune ($p = 0.05$)	11.80 ± 2.44	52.80 ± 1.80	11.05 ± 1.60	52.01 ± 0.35	10.21 ± 0.28	51.01 ± 0.20
PriPrune ($p = 0.1$)	10.00 ± 0.00	51.94 ± 1.86	10.00 ± 0.01	51.06 ± 0.64	10.00 ± 0.00	50.48 ± 0.81
Shatter	12.32 ± 2.03	58.58 ± 0.95	12.96 ± 2.16	54.65 ± 0.54	13.64 ± 1.55	52.03 ± 0.49
ERIS	37.12 ± 0.55	58.63 ± 0.88	38.59 ± 0.50	54.60 ± 0.39	38.95 ± 0.32	52.04 ± 0.41
ERIS (+DSC) ($\omega_{\text{SoteriaFL}}$)	37.36 ± 1.59	56.54 ± 0.94	38.43 ± 1.45	53.56 ± 0.57	38.41 ± 0.51	51.66 ± 0.52
ERIS (+DSC) ($\omega \approx 170$)	37.40 ± 1.36	57.49 ± 0.85	38.16 ± 1.01	53.98 ± 0.41	38.30 ± 0.88	51.70 ± 0.53
Min. Leakage	37.25 ± 0.38	55.81 ± 0.81	38.57 ± 0.37	53.06 ± 0.47	38.88 ± 0.36	51.67 ± 0.48

Table 13: Comparison of ERIS with and without compression (ω) against SOTA baselines in terms of test and MIA accuracy on MNIST with 4, 8, and 16 local training samples.

Local Training Size	4 **samples**		8 **samples**		16 **samples**	
Method	**Accuracy** (↑)	**MIA Acc.** (↓)	**Accuracy** (↑)	**MIA Acc.** (↓)	**Accuracy** (↑)	**MIA Acc.** (↓)
FedAvg	80.69 ± 1.71	82.13 ± 1.65	86.42 ± 0.88	72.00 ± 3.01	89.23 ± 0.74	65.78 ± 2.24
FedAvg ($\epsilon = 10, \delta$)-LDP	39.65 ± 3.14	69.07 ± 1.67	50.84 ± 4.75	59.44 ± 2.00	64.40 ± 1.53	57.67 ± 1.61
FedAvg ($\epsilon = 1, \delta$)-LDP	9.73 ± 0.46	69.50 ± 1.72	10.70 ± 1.27	58.50 ± 0.71	19.80 ± 1.38	57.55 ± 1.66
SoteriaFL ($\epsilon = 10, \delta$)	8.83 ± 2.65	71.47 ± 1.81	32.15 ± 2.00	57.68 ± 1.83	67.01 ± 1.31	56.87 ± 1.69
SoteriaFL ($\epsilon = 1, \delta$)	10.31 ± 0.38	67.50 ± 2.33	10.84 ± 0.79	57.90 ± 2.37	12.72 ± 1.33	57.27 ± 1.44
PriPrune ($p = 0.01$)	47.89 ± 8.33	77.20 ± 3.33	70.60 ± 3.67	68.32 ± 4.28	84.81 ± 0.31	63.56 ± 2.09
PriPrune ($p = 0.05$)	17.01 ± 4.22	58.00 ± 3.45	18.97 ± 3.12	50.16 ± 2.60	26.47 ± 2.59	54.51 ± 1.62
PriPrune ($p = 0.1$)	11.99 ± 1.87	56.67 ± 3.18	13.33 ± 2.21	49.44 ± 2.54	19.46 ± 1.35	53.42 ± 1.54
Shatter	11.96 ± 2.33	70.42 ± 2.21	12.32 ± 2.92	56.51 ± 2.86	14.55 ± 4.24	55.61 ± 1.33
ERIS	80.47 ± 1.75	69.82 ± 2.01	86.28 ± 1.00	56.35 ± 2.89	89.27 ± 0.73	55.60 ± 1.27
ERIS (+DSC) ($\omega_{\text{SoteriaFL}}$)	78.31 ± 1.45	68.70 ± 3.08	85.59 ± 0.67	54.43 ± 2.14	90.04 ± 0.43	55.58 ± 1.77
ERIS (+DSC) ($\omega \approx 30$)	78.72 ± 1.19	68.48 ± 3.11	84.84 ± 0.58	55.14 ± 2.73	90.26 ± 0.11	56.11 ± 1.58
Min. Leakage	80.68 ± 1.95	66.67 ± 2.67	86.30 ± 1.06	54.32 ± 1.67	89.26 ± 0.74	55.38 ± 1.56

Table 14: Comparison of ERIS with and without compression (ω) against SOTA baselines in terms of test and MIA accuracy on MNIST with 32, 64, and 128 local training samples.

Local Training Size	32 **samples**		64 **samples**		128 **samples**	
Method	**Accuracy (↑)**	**MIA Acc. (↓)**	**Accuracy (↑)**	**MIA Acc. (↓)**	**Accuracy (↑)**	**MIA Acc. (↓)**
FedAvg	91.48 ± 0.37	59.94 ± 1.11	92.55 ± 0.06	56.68 ± 1.92	93.11 ± 0.16	54.14 ± 0.65
FedAvg ($\epsilon = 10, \delta$)-LDP	70.35 ± 1.31	53.05 ± 1.09	70.43 ± 1.27	53.00 ± 1.00	70.48 ± 0.39	51.20 ± 0.91
FedAvg ($\epsilon = 1, \delta$)-LDP	57.75 ± 0.87	54.71 ± 0.85	70.96 ± 0.71	52.72 ± 0.74	70.56 ± 0.34	51.40 ± 0.66
SoteriaFL ($\epsilon = 10, \delta$)	77.95 ± 2.38	53.28 ± 0.60	78.15 ± 2.66	52.16 ± 0.84	79.50 ± 1.45	51.29 ± 0.77
SoteriaFL ($\epsilon = 1, \delta$)	8.49 ± 1.87	54.14 ± 0.53	40.37 ± 1.44	52.83 ± 0.24	68.75 ± 0.38	51.89 ± 0.53
PriPrune ($p = 0.01$)	87.77 ± 0.15	56.99 ± 1.14	87.01 ± 0.17	54.54 ± 1.03	86.38 ± 0.19	52.65 ± 0.69
PriPrune ($p = 0.05$)	33.01 ± 0.73	51.41 ± 1.34	34.81 ± 1.98	51.39 ± 1.00	33.87 ± 0.63	50.69 ± 0.93
PriPrune ($p = 0.1$)	21.08 ± 1.39	50.74 ± 1.58	20.68 ± 0.90	51.14 ± 0.98	20.46 ± 0.37	50.65 ± 0.87
Shatter	16.51 ± 6.16	52.02 ± 1.09	18.50 ± 6.06	51.46 ± 0.79	21.29 ± 7.22	51.61 ± 0.76
ERIS	91.52 ± 0.32	52.08 ± 1.04	92.54 ± 0.04	51.45 ± 0.85	93.12 ± 0.16	51.53 ± 0.79
ERIS (+DSC) ($\omega_{\text{SoteriaFL}}$)	92.27 ± 0.21	52.66 ± 0.73	93.23 ± 0.10	51.57 ± 0.74	93.78 ± 0.10	51.71 ± 0.67
ERIS (+DSC) ($\omega \approx 30$)	92.56 ± 0.30	52.71 ± 0.86	93.58 ± 0.23	51.74 ± 0.66	94.02 ± 0.19	51.65 ± 0.64
Min. Leakage	91.47 ± 0.39	52.06 ± 1.04	92.56 ± 0.05	51.41 ± 0.84	93.14 ± 0.17	51.50 ± 0.85

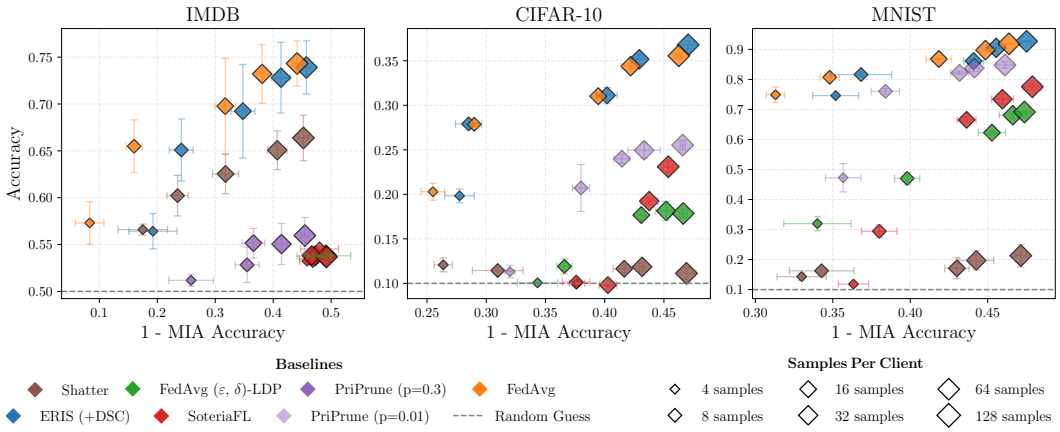

Figure 13: Comparison of accuracy and (1-MIA) accuracy across varying model sizes and client-side overfitting levels, controlled via the number of client training samples under non-IID setting.

istics—ranging from lightweight models such as LeNet-5 on MNIST to large-scale architectures like GPT-Neo 1.3B on CNN/DailyMail—and vary client-side overfitting by controlling the number of training samples per client. The observed trends mirror those under unbiased conditions: ERIS with DSC consistently achieves the best overall trade-off, retaining accuracy comparable to non-private FedAvg while substantially reducing privacy leakage toward the idealized *Min. Leakage* baseline. For instance, on IMDB with 4 local samples per client and identical training conditions (e.g., same communication rounds), ERIS with DSC achieves an accuracy of 67.8 $\pm$ 4.9, comparable to FedAvg's 66.9 $\pm$ 5.5, while significantly reducing MIA accuracy from 92.3% to 68.2%—approaching the unattainable upper bound of 66.9% obtained by not sharing local gradients. The only methods that surpass ERIS with DSC in privacy protection are DP-based approaches, which, however, degrade test accuracy to nearly random-guess levels, namely SoteriaFL (53.1 $\pm$ 0.8) and FedAvg-LDP (53.4 $\pm$ 0.5). Indeed, DP-based methods reduce leakage only at the cost of severe utility degradation, particularly for larger models, while decentralized methods with partial gradient exchange, such as Shatter, often fail to converge within the predefined number of communication rounds—especially when models are trained from scratch.

Table 16 reports mean test and MIA accuracy under the biased setting, complementing trends in Figure 14. Consistent with the figure, ERIS with DSC delivers the strongest utility–privacy balance across data- sets, maintaining accuracy close to FedAvg while reducing leakage toward the *Min. Leakage* baseline. DP-based methods achieve lower leakage but at a steep accuracy cost, PriPrune trades off utility and privacy depending on the pruning rate, and Shatter struggles to converge

Table 15: Mean test accuracy and MIA accuracy, averaged over varying local sample sizes under non-IID setting. For DP-based methods, ϵ=10; for PriPrune, pruning rates are $p \in \{0.1, 0.2, 0.3\}$ on IMDB and $p \in \{0.01, 0.05, 0.1\}$ on CIFAR-10/MNIST.

Method	IMDB – DistilBERT		CIFAR-10 – ResNet9		MNIST – LeNet5	
	Acc. (↑)	MIA Acc. (↓)	Acc. (↑)	MIA Acc. (↓)	Acc. (↑)	MIA Acc. (↓)
FedAvg	68.02 ± 7.03	72.34 ± 3.33	29.83 ± 0.85	63.52 ± 1.28	84.80 ± 1.76	60.17 ± 1.29
FedAvg (ϵ, δ)-LDP	53.79 ± 0.30	52.00 ± 3.97	15.13 ± 1.37	58.83 ± 1.73	55.69 ± 2.83	57.39 ± 2.31
SoteriaFL (ϵ, δ)	53.75 ± 0.81	52.70 ± 3.86	14.46 ± 0.55	59.07 ± 1.64	51.75 ± 2.77	57.64 ± 1.80
PriPrune (p_1)	57.01 ± 7.37	70.48 ± 3.42	21.30 ± 2.36	59.73 ± 1.52	74.80 ± 3.78	58.50 ± 1.56
PriPrune (p_2)	53.64 ± 3.40	67.52 ± 3.52	11.51 ± 1.41	57.71 ± 3.03	24.94 ± 5.34	53.95 ± 1.89
PriPrune (p_3)	54.03 ± 3.68	63.04 ± 4.23	10.98 ± 0.91	55.53 ± 2.24	15.54 ± 1.47	52.64 ± 2.45
Shatter	62.16 ± 4.19	68.26 ± 4.85	11.63 ± 1.31	62.19 ± 2.18	17.73 ± 5.56	59.67 ± 2.67
ERIS (+DSC)	67.49 ± 7.51	66.95 ± 4.72	30.16 ± 1.35	62.72 ± 2.04	85.10 ± 0.84	58.17 ± 2.24
Min. Leakage	68.88 ± 6.75	68.85 ± 3.00	29.80 ± 0.86	61.92 ± 3.09	84.95 ± 1.73	56.08 ± 1.67

reliably. These results further confirm that ERIS with DSC offers the most favorable trade-off, even in biased local training regimes.

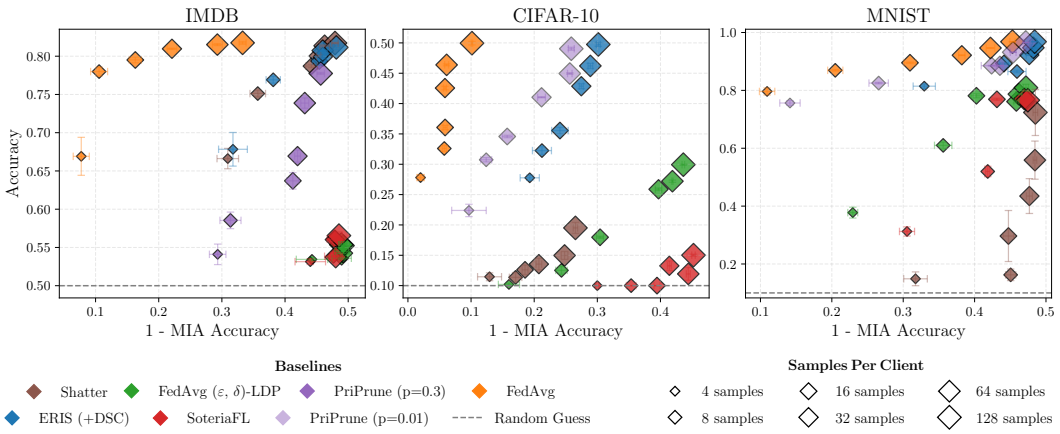

Figure 14: Comparison of test accuracy and (1-MIA) accuracy across varying model capacities (one per dataset) and client-side overfitting levels, controlled via the number of training samples per client using a biased gradient estimator.

Table 16: Mean test accuracy and MIA accuracy, averaged over varying local sample sizes using a biased gradient estimator. For DP-based methods, $\epsilon \in \{10, 100\}$ on IMDB and $\epsilon \in \{1, 10\}$ on others; for PriPrune, pruning rates are $p \in \{0.1, 0.2, 0.3\}$ on IMDB and $p \in \{0.01, 0.05, 0.1\}$ on others.

Method	IMDB – DistilBERT		CIFAR-10 – ResNet9		MNIST – LeNet5	
	Acc. (↑)	MIA Acc. (↓)	Acc. (↑)	MIA Acc. (↓)	Acc. (↑)	MIA Acc. (↓)
FedAvg	78.11 ± 1.42	80.13 ± 2.11	39.23 ± 0.76	94.03 ± 0.69	89.95 ± 0.77	68.65 ± 1.63
FedAvg (ϵ_1, δ)-LDP	54.39 ± 0.57	51.75 ± 2.69	10.78 ± 0.36	59.17 ± 1.21	29.02 ± 1.20	58.81 ± 0.95
FedAvg (ϵ_2, δ)-LDP	55.00 ± 1.10	52.69 ± 2.74	20.62 ± 0.57	67.34 ± 1.46	68.77 ± 2.09	60.33 ± 1.42
SoteriaFL (ϵ_1, δ)	54.90 ± 0.74	52.66 ± 2.73	10.03 ± 0.01	56.30 ± 0.84	14.23 ± 1.39	56.25 ± 0.95
SoteriaFL (ϵ_2, δ)	55.40 ± 1.88	53.44 ± 2.62	11.70 ± 0.75	60.69 ± 0.89	65.11 ± 1.88	57.16 ± 1.16
PriPrune (p_1)	76.52 ± 1.08	73.00 ± 2.30	37.12 ± 0.75	81.59 ± 2.17	91.11 ± 0.23	57.83 ± 1.63
PriPrune (p_2)	71.62 ± 1.30	66.66 ± 2.24	25.32 ± 1.13	64.16 ± 1.81	61.29 ± 1.16	55.14 ± 1.39
PriPrune (p_3)	65.82 ± 1.90	61.22 ± 2.10	13.91 ± 0.83	56.35 ± 1.67	53.99 ± 1.46	51.72 ± 1.42
Shatter	77.33 ± 0.87	58.26 ± 2.03	13.91 ± 1.77	79.90 ± 1.74	32.03 ± 11.54	56.47 ± 1.68
ERIS (+DSC)	77.59 ± 1.38	57.44 ± 2.24	39.06 ± 1.01	74.81 ± 1.99	90.16 ± 0.68	55.45 ± 1.68
Min. Leakage	78.11 ± 1.42	57.02 ± 2.10	39.23 ± 0.87	76.22 ± 1.91	90.06 ± 0.71	55.20 ± 1.50

F.10 Pareto Analysis under Varying Privacy Constraints

This section complements the analysis in Paragraph 4.2 of the main text with additional details and numerical results. We study how the utility–privacy trade-off evolves under different strengths of

privacy-preserving mechanisms and varying numbers of local training samples. Shatter is excluded, as it already fails to converge reliably with 16 samples per client (Figure 4). For DP-based approaches (FedAvg+LDP and SoteriaFL), we vary the privacy budget ϵ together with the clipping norm C to simulate different protection levels. Following the same configuration, we also evaluate ERIS with LDP, where LDP is applied on top of its native masking mechanism. For pruning-based methods (PriPrune), we vary the pruning rate p to control information flow through gradient sparsification. The exact configurations of these hyperparameters are in Table 17.

Figure 15 shows the utility–privacy trade-off across different numbers of local training samples. As expected, the Pareto frontier shifts toward higher accuracy and lower privacy leakage as clients are assigned more local data. Across all regimes, ERIS dominates the frontier, contributing the large majority of favorable points, while baselines are mostly dominated.

Table 17 reports the underlying quantitative results, including additional configurations not visualized in the figure. These include finer granularity in both p and ϵ values, enabling a more exhaustive comparison. The results confirm the trends observed in the main text: ERIS consistently occupies favorable positions in the utility–privacy space, contributing most points along the Pareto frontier. Moreover, when augmented with LDP, ERIS demonstrates further privacy gains with only minor utility losses—outperforming other baselines that suffer substantial degradation as privacy constraints tighten. Overall, this detailed breakdown reinforces that ERIS achieves strong privacy guarantees and high utility, even under stringent privacy budgets and aggressive compression strategies.

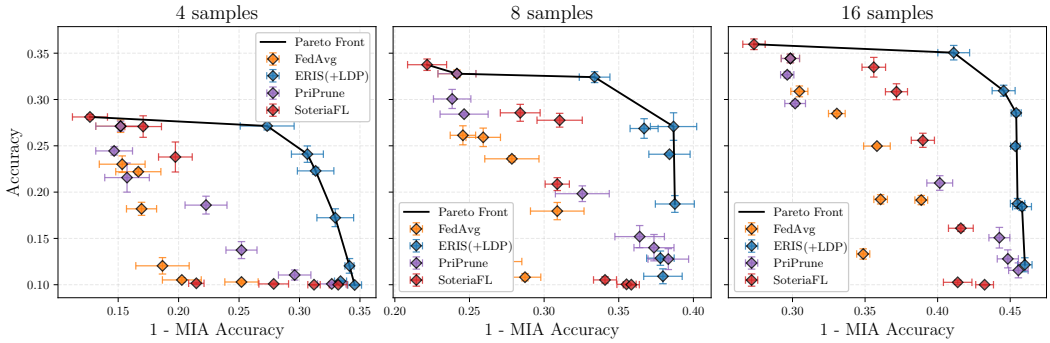

Figure 15: Utility–privacy trade-off on CIFAR-10 under varying strengths of the privacy-preserving mechanisms. Each subplot shows test accuracy vs. (1-MIA) accuracy for methods with different client training samples. The Pareto front represents a set of optimal trade-off points.

F.11 Temporal Evolution of Privacy Leakage and Utility

To complement the worst-case MIA summaries reported in the main paper, we examine how privacy leakage evolves throughout training. We consider the 64-sample IID MNIST configuration used in Table 1, with $K = 50$ clients and $A = 50$ aggregators, and train for $T = 250$ communication rounds. We compare ERIS with DSC against FedAvg, PriPrune ($p = 0.01$), and SoteriaFL. At every round, we record client validation accuracy and MIA accuracy. Curves and shaded regions report the mean and standard deviation across three seeds.

Figure 16 shows that the privacy trajectories are method-dependent and need not evolve monotonically. FedAvg exhibits a pronounced increase in MIA accuracy as training progresses, followed by a partial decrease after reaching its peak leakage, while PriPrune shows a more gradual increase. In contrast, ERIS with DSC remains close to random-guess MIA accuracy throughout training, with only mild fluctuations, while closely tracking the validation performance of FedAvg. SoteriaFL achieves similarly low MIA accuracy but at substantially lower validation accuracy. Overall, these trajectories show that the privacy advantage of ERIS is maintained throughout training.

The per-round MIA trajectories and Theorem 3.3 provide complementary views of privacy leakage. The theorem bounds the information accumulated in the adversary's transcript over T rounds and therefore scales with T, whereas Figure 16 measures the success of a specific attack at each individual round. Accordingly, the empirical MIA trajectories may stabilize or fluctuate even as the cumulative worst-case bound increases with training duration.

Table 17: Mean test accuracy and privacy leakage (with standard deviation) for various privacy-preserving mechanisms across different local sample sizes. DP-based methods use epsilon ϵ and clipping norm C; PriPrune uses rate p.

Method	4 samples		8 samples		16 samples	
	Accuracy (↑)	MIA Acc. (↓)	Accuracy (↑)	MIA Acc. (↓)	Accuracy (↑)	MIA Acc. (↓)
FedAvg + LDP						
No LDP	27.12% ± 1.20%	84.80% ± 4.59%	32.98% ± 0.61%	75.84% ± 2.85%	34.43% ± 1.04%	70.15% ± 1.41%
LDP (ϵ=0.001, C=10)	23.01% ± 2.00%	84.67% ± 4.24%	26.13% ± 2.30%	75.44% ± 1.90%	30.88% ± 1.30%	69.53% ± 1.31%
LDP (ϵ=0.01, C=5)	22.20% ± 0.71%	83.33% ± 4.17%	25.91% ± 2.23%	74.08% ± 2.58%	28.48% ± 0.72%	66.95% ± 1.29%
LDP (ϵ=0.1, C=2)	18.20% ± 1.58%	83.07% ± 2.78%	23.60% ± 0.76%	72.16% ± 4.06%	24.97% ± 0.48%	64.18% ± 2.05%
LDP (ϵ=0.3, C=1)	12.04% ± 1.98%	81.33% ± 4.94%	17.95% ± 2.08%	69.12% ± 4.01%	19.14% ± 1.12%	61.13% ± 0.99%
LDP (ϵ=0.6, C=1)	10.52% ± 0.55%	79.73% ± 3.59%	12.50% ± 1.42%	72.88% ± 3.10%	19.22% ± 1.12%	63.93% ± 1.05%
LDP (ϵ=1.0, C=1)	10.29% ± 0.40%	74.80% ± 3.08%	10.81% ± 0.88%	71.28% ± 2.37%	13.32% ± 1.24%	65.13% ± 1.01%
ERIS + LDP						
No LDP	27.14% ± 0.95%	72.67% ± 4.99%	32.21% ± 1.32%	66.62% ± 2.30%	35.05% ± 1.75%	58.89% ± 2.45%
LDP (ϵ=0.001, C=10)	24.10% ± 1.98%	69.35% ± 2.95%	26.87% ± 2.37%	63.30% ± 2.10%	30.95% ± 1.32%	55.45% ± 1.77%
LDP (ϵ=0.01, C=5)	22.29% ± 0.86%	68.67% ± 3.37%	27.08% ± 3.33%	59.32% ± 3.47%	28.55% ± 0.71%	54.59% ± 0.77%
LDP (ϵ=0.1, C=2)	17.24% ± 2.15%	67.04% ± 3.40%	24.09% ± 0.33%	61.60% ± 3.09%	24.96% ± 0.63%	54.64% ± 0.73%
LDP (ϵ=0.3, C=1)	12.03% ± 1.77%	65.87% ± 0.84%	18.72% ± 2.03%	61.24% ± 2.92%	18.76% ± 1.23%	54.49% ± 1.10%
LDP (ϵ=0.6, C=1)	10.38% ± 0.16%	66.61% ± 1.12%	12.87% ± 1.73%	62.21% ± 2.06%	18.43% ± 0.93%	54.18% ± 1.45%
LDP (ϵ=1.0, C=1)	9.97% ± 0.04%	65.44% ± 1.27%	10.91% ± 1.80%	62.94% ± 2.88%	12.15% ± 1.69%	53.99% ± 1.11%
SoteriaFL						
No LDP	28.11% ± 0.61%	87.33% ± 3.24%	33.76% ± 1.38%	77.84% ± 2.92%	35.96% ± 1.28%	72.65% ± 1.72%
LDP (ϵ=0.001, C=10)	27.08% ± 2.60%	82.93% ± 3.39%	28.56% ± 2.05%	71.60% ± 2.98%	33.48% ± 2.37%	64.40% ± 1.82%
LDP (ϵ=0.01, C=5)	23.79% ± 3.62%	80.27% ± 3.12%	27.76% ± 1.67%	68.96% ± 3.39%	30.84% ± 1.93%	62.84% ± 1.70%
LDP (ϵ=0.1, C=2)	10.16% ± 0.26%	78.53% ± 1.42%	20.86% ± 1.56%	69.12% ± 1.81%	25.59% ± 1.71%	61.02% ± 1.78%
LDP (ϵ=0.3, C=1)	10.09% ± 0.18%	72.13% ± 2.75%	10.52% ± 0.76%	65.92% ± 1.72%	16.10% ± 0.78%	58.40% ± 1.91%
LDP (ϵ=0.6, C=1)	10.00% ± 0.00%	68.80% ± 1.15%	10.04% ± 0.08%	64.48% ± 1.44%	10.26% ± 0.52%	58.62% ± 2.20%
LDP (ϵ=1.0, C=1)	10.00% ± 0.00%	66.80% ± 1.65%	10.00% ± 0.00%	64.16% ± 1.20%	10.00% ± 0.00%	56.76% ± 1.37%
PriPrune						
No Pruning	27.12% ± 1.20%	84.80% ± 4.59%	32.98% ± 0.61%	75.84% ± 2.85%	34.43% ± 1.04%	70.15% ± 1.41%
Pruning (p=0.0005)	24.45% ± 1.00%	85.33% ± 3.40%	30.06% ± 2.30%	76.16% ± 2.80%	32.67% ± 0.79%	70.36% ± 1.02%
Pruning (p=0.001)	21.57% ± 3.51%	84.27% ± 4.12%	28.42% ± 0.39%	75.36% ± 3.60%	29.57% ± 0.70%	69.82% ± 1.59%
Pruning (p=0.005)	18.60% ± 2.15%	77.73% ± 3.88%	19.83% ± 1.89%	67.44% ± 4.04%	20.98% ± 1.77%	59.85% ± 1.99%
Pruning (p=0.01)	13.74% ± 2.05%	74.80% ± 2.87%	15.20% ± 2.67%	63.60% ± 3.71%	15.08% ± 2.49%	55.75% ± 1.61%
Pruning (p=0.03)	11.05% ± 1.22%	70.40% ± 3.00%	14.00% ± 3.18%	62.64% ± 3.00%	12.79% ± 2.17%	55.16% ± 1.63%
Pruning (p=0.05)	10.09% ± 0.36%	67.33% ± 2.63%	12.77% ± 2.52%	61.68% ± 3.04%	11.55% ± 1.80%	54.44% ± 1.50%

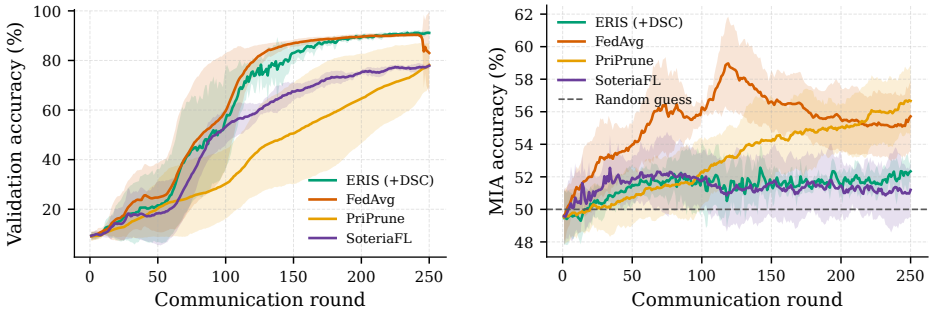

Figure 16: Temporal evolution of utility and privacy leakage on MNIST. *Left:* validation accuracy. *Right:* MIA accuracy, with random guessing indicated at 50%. Curves and shaded regions report the mean and standard deviation over three seeds.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See section 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See section 5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: See Appendix C and Appendix D.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: See subsection 4.1 and Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The data used is publicly available, and we provide the official code.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: See subsection 4.1 and Appendix E for the experimental settings, and section 4 and Appendix F for details on specific experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: See figures and tables.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See subsection E.3 for the hardware configuration and subsection F.2 for the computational/communication scalability analysis and per-round distribution costs. Exact total wall-clock compute across all experimental runs was not systematically tracked.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have thoroughly reviewed and adhered to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See section 4 and section 5.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not address safeguards for the responsible release of high-risk data or models, as our research does not involve such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We are the authors of the code used in this work. Information about other assets and the licenses of used models can be found in subsection E.3.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: See subsection 4.1, Appendix E, and the provided code.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve research with user study.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.