mathematics

MDPI

Article

FedCap-Priv: Geometry-Aware Privacy and Bandwidth-Efficient Routing for Federated Capsule Learning

Yantao Li and Zijie Pan *

Faculty of Data Science, City University of Macau, Macao 999078, China
* Correspondence: d22091101069@cityu.edu.mo

Abstract

Capsule networks explicitly encode pose and part–whole structure, making them attractive for heterogeneous visual and speech data but creating structured messages and substantial communication costs in federated learning. We present FedCap-Priv, a federated capsule framework that jointly addresses transcript privacy, predictive utility, and uplink cost through three capsule-specific procedures. Secure Routing by Agreement (SeRBA) retains dynamic routing on each client and replaces direct vote exchange with compact directional and low-rank agreement sketches. Pose-Sensitive Differential Privacy (PS-DP) clips each layer sketch and applies anisotropic Gaussian perturbation aligned with pose variability. Our client-level Rényi-DP analysis accounts for data-dependent means and covariances, visible selection indices, layer composition, client participation, and temporal composition. Sparsity-Guided Partial Aggregation (SGPA) selects salient parent blocks after perturbation and uses error feedback to reduce communication. Across MNIST, Fashion-MNIST, CIFAR-10, and Speech Commands, FedCap-Priv improves accuracy over isotropic FedDP and lowers reconstruction scores under DLG, iDLG, and cosine-TV attacks. On CIFAR-10, it reaches 73.9% accuracy versus 68.9% for isotropic FedDP while reducing the mean participating-client uplink from 220 KB to 22.1 KB per round, a 9.95-fold reduction. The sketch interface supports vector- and matrix-pose capsules and can be combined with secure aggregation.

Keywords: federated learning; machine learning; capsule network; privacy protection

MSC: 62R07

1. Introduction

Federated learning (FL) coordinates model training without centralizing client records, but its transmitted gradients, parameters, and intermediate statistics can still reveal sensitive information [1–5]. Capsule networks add a distinctive interface to this problem. Their vector- or matrix-valued poses encode instantiation attributes, and dynamic routing propagates evidence according to agreement between child predictions and candidate parent poses [6,7]. These structured quantities support viewpoint-aware modeling, yet exchanging votes, routing logits, or complete capsule updates exposes detailed client-specific geometry and incurs communication proportional to the number of capsule parameters or child–parent pairs.

Federated capsule learning must therefore coordinate three coupled decisions: what routing information leaves a client, how that message is privatized, and which parts are transmitted. Local routing should preserve capsule agreement without disclosing raw votes

Check for updates

Academic Editors: Chao Tong and Mingming Liu

Received: 22 August 2026
Revised: 9 October 2026
Accepted: 19 October 2026
Published: 2 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

or child poses. A client-level privacy guarantee must cover the complete observable transcript, including any data-dependent noise covariance and sparse support. Communication reduction must also preserve an implementable server update: a server receiving only aggregate sketches cannot update a transformation matrix from an uncommunicated child pose. Treating privacy, routing, and sparsification independently can violate one of these requirements; in particular, selecting blocks before perturbation exposes a data-dependent index set.

We address this problem with FedCap-Priv, an integrated protocol built from three procedures: Secure Routing by Agreement (SeRBA) performs anchored dynamic routing locally and summarizes each parent with a directional mean and a low-rank second moment. Pose-Sensitive Differential Privacy (PS-DP) clips each layer sketch and adds anisotropic Gaussian noise aligned with the client's pose variability, together with a strictly positive isotropic variance floor. Sparsity-Guided Partial Aggregation (SGPA) ranks parent blocks only after perturbation, transmits the scheduled subset, and carries the remaining randomized mass through error feedback. The server forms blockwise weighted means and updates public parent anchors using only the communicated statistics. Figure 1 and Algorithms 1–4 specify this end-to-end interface.

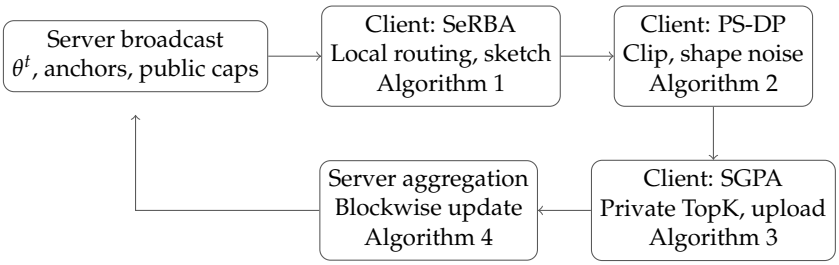


Figure 1. One communication round. Routing and unnoised statistics remain on the client; only selected noisy blocks and their indices reach the server.

Algorithm 1 Client-k SeRBA Procedure for Layer ℓ

Given: child poses $\{u_i\}$, transforms $\{W_{ij}\}$, public anchors $\{\bar{v}_j^t\}$, R iterations, anchor coefficient λ_{anc}, and rank r.
Initialize $b_{ij} \leftarrow 0$, $v_j \leftarrow \bar{v}_j^t$.
for $r' = 1$ to R **do**
  $c_{ij} \leftarrow \text{softmax}_j(b_{ij})$; $s_j \leftarrow \sum_i c_{ij} W_{ij} u_i$; $v_j \leftarrow \text{squash}(s_j)$.
  $b_{ij} \leftarrow b_{ij} + (W_{ij}u_i)^\top v_j - \lambda_{\text{anc}} \|v_j - \bar{v}_j^t\|_2^2$.
end for
Compute $\mu_j \leftarrow \sum_i c_{ij} \frac{W_{ij}u_i}{\|W_{ij}u_i\|_2+\epsilon}$; $M_j \leftarrow \sum_i c_{ij} \frac{(W_{ij}u_i)(W_{ij}u_i)^\top}{\|W_{ij}u_i\|_2+\epsilon}$.
Approximate M_j with its rank-r factor Q_j using randomized range finding.
Construct $A_{\ell,j} \leftarrow [\,\mu_j;\, \text{vec}(Q_j)\,]$ and concatenate the parent blocks into A_ℓ.
Pass the raw A_ℓ to Algorithm 2 for clipping and perturbation.
Return: layer-ℓ agreement sketch A_ℓ for subsequent PS-DP protection.

Algorithm 2 Layer-ℓ PS-DP release on client k

Given raw sketch A_ℓ, public cap $C_{\ell,\max}$, local median history, EMA state, $\beta, \gamma, r, \sigma_\ell, \sigma_{\min,\ell}$.
Set $C_\ell \leftarrow \min\{C_{\ell,\max}, 1.2\,\text{median}_{s\le t}\|A_\ell^{(s)}\|_2\}$ and $x_\ell \leftarrow \text{clip}(A_\ell, C_\ell)$.
Update the mean and covariance EMA using x_ℓ; compute the leading r eigenpairs of V_ℓ.
Sample $\eta_\ell \sim \mathcal{N}(0, S_\ell)$ with the public variance floor and data-dependent directional component.
Return $Y_\ell = x_\ell + \eta_\ell$ to Algorithm 3.

Algorithm 3 Client-k SGPA filtering at layer ℓ

Given private sketch Y_ℓ from Algorithm 2, residual r_ℓ, and public budget $\kappa_\ell(t)$.
Set $z_\ell \leftarrow Y_\ell + r_\ell$ and $\mathcal{J}_\ell \leftarrow \mathrm{TopK}_j(\|z_{\ell,j}\|_2, \kappa_\ell(t))$.
Set $m_\ell \leftarrow \mathrm{mask}_{\mathcal{J}_\ell}(z_\ell)$ and $r_\ell \leftarrow z_\ell - m_\ell$.
Upload $(\mathcal{J}_\ell, m_\ell[\mathcal{J}_\ell])$.

Algorithm 4 Sketch-only server aggregation in round t

Given received pairs $(\mathcal{J}_\ell^{(k)}, m_\ell^{(k)}[\mathcal{J}_\ell^{(k)}])$, public w_k, and old public anchors $\bar{v}_{\ell,j}^t$.
for each layer ℓ and parent j **do**
  If $h_{\ell,j} = 0$, retain $\bar{v}_{\ell,j}^t$. Otherwise form $\bar{A}_{\ell,j}$ and parse $(\bar{\mu}_{\ell,j}, \bar{Q}_{\ell,j})$.
  Update $\bar{v}_{\ell,j}^{t+1}$ using the displayed ridge-stabilized map.
end for
Broadcast the new anchors; no private child pose is requested or reconstructed.

The privacy analysis follows the mechanism actually implemented. Neighboring federations differ in the complete dataset of one client. Because the PS-DP covariance is derived from local pose statistics, neighboring releases can differ in both mean and covariance. We bound the Rényi divergence between the corresponding full Gaussian laws over public spectral intervals, compose all released layers, account for visible client participation, and compose across rounds. SGPA indices, sparse values, aggregates, and subsequent anchors are postprocessing of the protected sketch. The resulting guarantee therefore covers the full server-visible transcript rather than only the Gaussian sample values. A conditional descent result is stated for the sketch-to-anchor map itself, with approximation bias and conditional variance explicit.

We evaluate FedCap-Priv on MNIST, Fashion-MNIST, CIFAR-10, and Speech Commands under heterogeneous client partitions. The experiments compare accuracy, client-level privacy budgets, participating-client payload, component variants, and reconstruction resistance under DLG, iDLG, and cosine-TV attacks. On CIFAR-10, FedCap-Priv reaches 73.9% accuracy versus 68.9% for isotropic FedDP while reducing the mean uplink from 220 KB to 22.1 KB per participating client and round, a 9.95-fold reduction. Across all four benchmarks, it retains accuracy close to the non-private capsule baseline and consistently improves on isotropic perturbation at matched budgets.

The contributions are fourfold: (1) a capsule-specific interface that keeps routing local and updates public anchors from compact agreement sketches; (2) client-level Rényi-DP accounting for adaptive covariance, visible sparse indices, layers, client participation, and rounds; (3) post-perturbation parent-block sparsification with error feedback and explicit payload accounting; and (4) evaluation on four image and speech benchmarks covering accuracy, privacy, reconstruction resistance, and communication.

2. Related Work

Federated Optimization under Heterogeneity: FedAvg established the standard cycle of local optimization and server averaging, while later work characterized its statistical and communication behavior [1,2,8]. FedProx constrains local drift, SCAFFOLD uses control variates, and FedAdam adapts the server optimizer to heterogeneous updates [9–11]. Radical heavy-ball acceleration targets communication-efficient mobile FL, and neighborhood/global perturbation methods incorporate sharpness-aware objectives across local and global neighborhoods [12,13]. These approaches optimize conventional model updates. FedCap-Priv instead redesigns the capsule message and server-visible interface; paired component comparisons keep the optimizer fixed.

Differential Privacy and Secure Aggregation: DP-SGD clips contributions and adds Gaussian noise, Rényi DP enables composition across iterative releases, and subsampled accountants incorporate randomized participation [14–17]. Client-level DP treats one user's complete local dataset as the protected unit [18]. Secure aggregation restricts the server to aggregates under its protocol and trust assumptions but does not bound the information carried by the aggregate [19]. PS-DP differs from a fixed isotropic release because its covariance is estimated from private pose statistics. The analysis therefore includes changes in both Gaussian mean and covariance and treats SGPA support as observable transcript metadata.

Attacks on Collaborative Learning: Gradient inversion can reconstruct training examples from shared gradients or updates [3,20]; membership and property inference target different facts about client data [4,5]. Poisoning settings add capabilities such as backdoor insertion [21]. FedCap-Priv adopts an honest-but-curious server that observes the prescribed plaintext transcript. Its empirical evaluation uses reconstruction objectives, while client-level DP applies to any inference computed from the same protected transcript.

Capsule Routing and Efficient Capsule Models: Dynamic routing and matrix capsules represent part–whole relationships with vector or matrix poses [6,7]. Sparse capsule architectures, projection-based capsules, and alternative routing procedures reduce computation or improve routing efficiency within a centralized model [22–24]. FedCap-Priv addresses a complementary systems boundary: it keeps votes and logits local, summarizes parent agreement at fixed rank, protects the summary, and defines aggregation from that protected message. This construction applies to both vector- and matrix-pose capsules through the common sketch interface.

Communication-Efficient Learning: Quantization, low-precision gradients, TopK sparsification, sign-based updates, and error feedback reduce distributed-training traffic [25–30]. These methods usually select or encode ordinary gradients. SGPA instead operates on semantic parent blocks consisting of a directional mean and low-rank moment factor. It performs TopK selection after PS-DP perturbation and stores only residuals derived from protected sketches, so the support and later error-feedback state remain within the transcript-level postprocessing argument.

3. Preliminaries

This section introduces the notation and mathematical foundations used for capsule networks, federated optimization, and our privacy formulation. It also formalizes the adversarial setting and the assumptions required by the later analysis.

3.1. Capsule Networks

A capsule groups neurons so that their collective output is a vector- or matrix-valued representation of an entity. In contrast to a scalar CNN activation, this output encodes both the entity's likelihood of being present, through its magnitude, and its instantiation attributes, through its pose.

More precisely, write the output of lower-layer capsule i as $u_i \in \mathbb{R}^d$. A transformation matrix W_{ij} maps that capsule to parent capsule j:

$$\hat{u}_{j|i} = W_{ij} u_i.$$

Dynamic routing combines all child predictions $\hat{u}_{j|i}$ using coupling weights c_{ij}:

$$s_j = \sum_i c_{ij} \hat{u}_{j|i}, \qquad v_j = \frac{\|s_j\|^2}{1 + \|s_j\|^2} \frac{s_j}{\|s_j\|},$$

where v_j denotes the resulting pose of capsule j. The coefficients c_{ij} are refined over several iterations according to the agreement between each prediction $\hat{u}_{j|i}$ and v_j. Matrix capsules equipped with EM routing [7] instead maintain matrix poses and variance terms, although they follow the same basic idea of routing from votes.

3.2. Federated Learning

Suppose that K clients participate and client k owns a local collection $\mathcal{D}_k$ containing n_k examples. With $N = \sum_k n_k$ total examples, federated training solves

$$\min_{\theta} F(\theta) = \sum_{k=1}^{K} \frac{n_k}{N} F_k(\theta), \qquad F_k(\theta) = \frac{1}{n_k} \sum_{(x,y)\in\mathcal{D}_k} \ell(f_\theta(x), y),$$

where θ parameterizes capsule model f_θ and ℓ denotes its training loss.

The standard FedAvg procedure repeats three operations: the server sends θ^t to selected clients, those clients apply local SGD to obtain θ_k^{t+1}, and the server computes the weighted aggregate

$$\theta^{t+1} = \sum_{k\in S_t} \frac{n_k}{\sum_{j\in S_t} n_j} \theta_k^{t+1}.$$

3.3. Threat Model and Adjacency

The server follows the protocol but can inspect each plaintext client message, including layer and parent indices. For client-level privacy, neighboring federations $D \sim D'$ differ only in the entire dataset of one fixed client k; all other client datasets, public hyperparameters, and participation randomness are identical. The replacement relation allows arbitrary contents for D_k and D_k'. Client selection is independent of the data, and the server observes whether k participated.

The formal mechanism covers the sequence of uploaded messages and all deterministic server outputs. The empirical attack protocol focuses on message reconstruction through DLG, iDLG, and cosine-TV objectives; the transcript-level DP guarantee also bounds any membership or property decision computed from the same view. The threat model excludes malicious-client poisoning and direct disclosure of unnoised local state. Secure aggregation may further restrict the server's view under a separate implementation; the analysis below holds for plaintext client messages.

4. Methodology

We now describe FedCap-Priv, which jointly treats capsule routing, private disclosure, and network traffic in a federated setting. It combines Secure Routing by Agreement (SeRBA), Pose-Sensitive Differential Privacy (PS-DP), and Sparsity-Guided Partial Aggregation (SGPA). In combination, these modules limit geometry-related exposure and make agreement more stable on non-IID client data.

4.1. Overview of a Training Round

At round t, the server broadcasts public model parameters θ^t, pose anchors $\bar{v}_{\ell,j}^t$, public clipping caps $C_{\ell,\max}$, noise parameters, and a predetermined block budget $\kappa_\ell(t)$. A selected client performs local capsule optimization and follows the stages in Figure 1. Algorithm 1 produces a layer sketch, Algorithm 2 perturbs its clipped version, Algorithm 3 selects and transmits blocks, and Algorithm 4 combines the received statistics.

The client message is $M_k^t = \{(\ell, \mathcal{J}_\ell^{(k)}, \tilde{A}_\ell^{(k)}[\mathcal{J}_\ell^{(k)}])\}_{\ell=1}^{L}$. The index set $\mathcal{J}_\ell^{(k)}$ is observable and is part of the protected output. Local routing votes, logits, and child poses are absent from M_k^t. A sparse secure-aggregation implementation would need to account separately

for its index and padding overhead; the byte counts in the empirical section refer to plaintext sparse uploads.

Main Stepsof a Communication Round

1. The server samples clients independently of their data and broadcasts the current public parameters, parent anchors, clipping caps, noise settings, and layer budgets. Public schedules are fixed before observing the current client sketches.
2. Each selected client performs its prescribed local optimization. At a capsule layer, it evaluates transformed child predictions and runs the anchored routing recursion. Votes, logits, and example-level activations remain in local memory.
3. After the final routing iteration, the client forms one directional mean and one rank-r moment factor per parent. Concatenation across parents yields a raw layer sketch. Algorithm 1 ends at this unprotected interface.
4. Algorithm 2 selects the moving clipping threshold, applies the public cap, updates the local covariance state, and adds Gaussian noise with the public variance floor. The full randomized layer sketch is the primitive private release used by the later steps.
5. Algorithm 3 combines the randomized sketch with its privatized residual, ranks parent blocks, and retains the scheduled number of blocks. It outputs both their indices and values and stores the remaining randomized mass for later rounds.
6. The sparse messages are transmitted. Secure aggregation can be added if its sparse encoding and trust assumptions are specified, but it is not required by the privacy proof in Section 5.
7. Algorithm 4 computes blockwise weighted means for observed parents and updates the public anchors. Missing parents keep their preceding anchors. The next broadcast closes the round.

This sequence fixes the point at which each operation sees private information. In particular, reversing PS-DP and SGPA would expose indices derived from an unprotected statistic, moving covariance estimation to the server would require transmitting an additional private quantity, and updating transformations at the server would require a private child pose or cross-moment. The stated order avoids those unstated channels.

Table 1 follows each state through the round and makes the information boundary explicit. Only the noisy sparse message crosses from a client to the server. Local state can still influence later messages, which is why the privacy analysis conditions on the complete preceding transcript rather than treating rounds independently.

Table 1. Information boundary for a FedCap-Priv round. "Protected" means included in the analyzed randomized transcript.

Quantity	Location	Sent	Status
Raw examples and labels	Client	No	Private input
Child poses, votes, and logits	Client	No	Ephemeral local state
Raw agreement sketch A_ℓ	Client	No	Input to clipping
EMA mean and covariance V_ℓ	Client	No	Private adaptive state
Noisy sketch Y_ℓ	Client	No, in full	Primitive protected release
Indices $\mathcal{J}_\ell$ and selected blocks	Client to server	Yes	Postprocessing of Y_ℓ and protected residuals
Blockwise aggregate $\bar{A}_{\ell,j}$	Server	Internal	Postprocessing of received messages
Updated anchors $\bar{v}_{\ell,j}^{t+1}$	Server to clients	Yes	Public postprocessing

4.2. Secure Routing by Agreement (SeRBA)

Instead of exposing vulnerable routing messages such as unprocessed predictions $\hat{u}_{j|i}$ and logits b_{ij}, SeRBA performs all routing locally and produces a compact agreement sketch

to which privacy protection is subsequently applied. The method is intended to retain the stability associated with agreement-based routing, place a bound on the geometric information disclosed by pose predictions, and use a small fixed sketch dimension so that communication stays efficient. Algorithm 1 summarizes the procedure.

4.2.1. Local Routing with Anchors

Take a layer ℓ whose children are $i \in \mathcal{C}_\ell$ with poses $u_i \in \mathbb{R}^{d_\ell}$, and whose parents are $j \in \mathcal{P}_\ell$ with poses $v_j \in \mathbb{R}^{d_{\ell+1}}$. Given $W_{ij} \in \mathbb{R}^{d_{\ell+1} \times d_\ell}$, child i predicts $\hat{u}_{j|i} = W_{ij} u_i$ for parent j. The following R routing steps are executed on the client:

$$c_{ij}^{(r)} = \mathrm{softmax}_j(b_{ij}^{(r)}), \qquad s_j^{(r)} = \sum_{i \in \mathcal{C}_\ell} c_{ij}^{(r)} \hat{u}_{j|i}, \qquad v_j^{(r+1)} = \mathrm{squash}(s_j^{(r)}),$$
$$b_{ij}^{(r+1)} = b_{ij}^{(r)} + \hat{u}_{j|i}^\top v_j^{(r+1)} - \lambda_{\mathrm{anc}} \| v_j^{(r+1)} - \bar{v}_j^t \|_2^2.$$

The initial values are $b_{ij}^{(0)} = 0$ and $v_j^{(0)} = \bar{v}_j^t$. The anchor term is intended to discourage local pose drift; the stated recursion alone does not prove contraction under non-IID data. Routing logits remain local.

The anchor supplies a common reference across clients and rounds. Its coefficient λ_{anc} governs the tradeoff between following current local agreement and remaining close to the broadcast orientation. When the coefficient is zero, the recursion reduces to ordinary local dynamic routing. A large coefficient can suppress useful local variation, so it is a tuning parameter rather than a privacy parameter. Because the anchor is computed from earlier protected releases, reusing it is postprocessing of the accumulated transcript.

4.2.2. Directional Mean and Low-Rank Second Moment

Once routing finishes, client k compresses the agreement for each parent into a constant-size representation. Its directional average and rank-r second-moment statistic are

$$\mu_j = \sum_i c_{ij} \frac{\hat{u}_{j|i}}{\|\hat{u}_{j|i}\|_2 + \epsilon}, \qquad M_j = \sum_i c_{ij} \frac{\hat{u}_{j|i} \hat{u}_{j|i}^\top}{\|\hat{u}_{j|i}\|_2 + \epsilon}.$$

Using a one-pass randomized range finder or CountSketch, approximate M_j at rank r as $M_j \approx Q_j Q_j^\top$, with $Q_j \in \mathbb{R}^{d_{\ell+1} \times r}$. Concatenating the two components gives the parent sketch

$$A_{\ell,j} = [\, \mu_j \,;\, \mathrm{vec}(Q_j) \,] \in \mathbb{R}^{m_\ell}, \qquad m_\ell = d_{\ell+1} + r\, d_{\ell+1}.$$

Concatenating parent blocks gives the raw layer sketch $A_\ell \in \mathbb{R}^{|\mathcal{P}_\ell| m_\ell}$. It retains aggregate direction and magnitude, so the sketch itself is not a privacy mechanism. Algorithm 2 applies the one layer-wise clipping operation with a local moving-median threshold capped by a public $C_{\ell,\max}$. The public cap, rather than the realized private threshold, controls the worst-case privacy bound.

4.3. Pose-Sensitive Differential Privacy (PS-DP)

For a raw layer sketch $A_\ell(D_k)$, define $x_\ell(D_k) = \mathrm{clip}(A_\ell(D_k), C_\ell(D_k))$ and release $Y_\ell = x_\ell(D_k) + \eta_\ell$. Because both $x_\ell(D_k)$ and $x_\ell(D_k')$ lie in the radius-$C_{\ell,\max}$ ball, replacement adjacency gives the exact uniform bound

$$\|x_\ell(D_k) - x_\ell(D_k')\|_2 \le \|x_\ell(D_k)\|_2 + \|x_\ell(D_k')\|_2 \le 2C_{\ell,\max}.$$

The bound applies even when the moving threshold changes between neighboring clients. An add/remove convention with a fixed zero contribution gives $C_{\ell,\max}$ instead.

Data-Dependent Covariance

Let $\bar{x}_\ell$ and V_ℓ be exponential moving averages of clipped sketches and their centered outer products, initialized at zero. With $0 \le \beta < 1$, update $\bar{x}_\ell \leftarrow \beta\bar{x}_\ell + (1-\beta)x_\ell$ and $V_\ell \leftarrow \beta V_\ell + (1-\beta)(x_\ell - \bar{x}_\ell)(x_\ell - \bar{x}_\ell)^\top$. Write $V_\ell = U\,\mathrm{diag}(\lambda_i)U^\top$ and retain at most rank r. The released noise has covariance

$$S_\ell(D_k) = \sigma^2_{\min,\ell} I + \sigma^2_\ell U_r\,\mathrm{diag}(\lambda_1^\gamma, \ldots, \lambda_r^\gamma)U_r^\top, \qquad \eta_\ell \sim \mathcal{N}(0, S_\ell(D_k)).$$

At $\gamma = 0$, retained eigenvalues receive weight one, including a retained zero eigenvalue. Since $\|x_\ell\|_2, \|\bar{x}_\ell\|_2 \le C_{\ell,\max}$, every centered vector has norm at most $2C_{\ell,\max}$. Consequently, $0 \preceq V_\ell \preceq 4C^2_{\ell,\max} I$ and $a_\ell I \preceq S_\ell(D_k) \preceq b_\ell I$, where $a_\ell = \sigma^2_{\min,\ell} > 0$ and $b_\ell = a_\ell + \sigma^2_\ell \max\{1, (4C^2_{\ell,\max})^\gamma\}$. Algorithm 2 states the implemented order of operations.

The variance floor has two roles: Computationally, it prevents an unobserved or truncated direction from receiving zero noise. Analytically, it makes every admissible covariance positive-definite and supplies the common lower spectral bound used in the Gaussian divergence. The directional term then reallocates additional variance along the retained eigenspace. The exponent γ controls how strongly large empirical eigenvalues dominate that allocation, while the rank r limits client computation and storage. Neither parameter changes the need to account for the covariance's dependence on the client dataset.

The moving mean and covariance are local state. At round t, their values may depend on every earlier sketch from the same client, so the current release is adaptive with respect to the prior transcript. The privacy analysis therefore conditions on an arbitrary previous transcript and takes a uniform bound over all EMA states satisfying the public spectral interval. This is more conservative than treating the covariance as fixed, but it covers the implemented stateful mechanism without transmitting or separately privatizing the covariance estimate.

4.4. Sparsity-Guided Partial Aggregation (SGPA)

SGPA applies TopK only after PS-DP. For each parent block j, let $s_{\ell,j} = \|Y_{\ell,j} + r_{\ell,j}\|_2$; the residual $r_{\ell,j}$ contains only previously privatized values and is initialized at zero. A public deterministic tie rule defines $\mathcal{J}_\ell = \mathrm{TopK}_j(s_{\ell,j}, \kappa_\ell(t))$. The transmitted indices and values are therefore deterministic postprocessing of the sequence of noisy sketches and the public schedule. Their disclosure consumes no additional privacy budget beyond those sketches. Raw routing salience, an unprotected median, or a fresh private quantity must never be used to choose upload indices.

Set $z_\ell = Y_\ell + r_\ell$, transmit $\mathrm{mask}_{\mathcal{J}_\ell}(z_\ell)$, and retain $r_\ell \leftarrow z_\ell - \mathrm{mask}_{\mathcal{J}_\ell}(z_\ell)$. The residual is stateful postprocessing of already protected releases; privacy composes over all rounds. Its value can still affect future utility and convergence. Algorithm 3 specifies this order.

Private Selection and Residual Semantics

TopK is applied at parent-block granularity because a directional mean and its moment factor describe one joint capsule statistic. Splitting those entries across different supports would complicate parsing and could leave an incomplete preconditioner. The deterministic tie rule is public, and the budget $\kappa_\ell(t)$ depends only on the round and layer. Consequently, two executions that produce the same randomized sketch and residual produce the same sparse message.

Error feedback preserves randomized information omitted in the current message; it does not recover a raw sketch. A residual may accumulate across rounds and alter later supports, but it remains a deterministic function of earlier PS-DP outputs. Privacy therefore follows by adaptive postprocessing and temporal composition. Utility analysis still needs

to control the residual norm, because a persistent large residual can bias or destabilize the anchor update even though it adds no separate privacy charge.

4.5. Server-Side Aggregation

For every received parent block, the server computes a weighted mean of the transmitted noisy values. Write $h_{\ell,j} = \sum_{k \in S_t} w_k \mathbf{1}\{j \in \mathcal{J}_\ell^{(k)}\}$ and, when $h_{\ell,j} > 0$,

$$\bar{A}_{\ell,j} = \frac{\sum_{k \in S_t} w_k \mathbf{1}\{j \in \mathcal{J}_\ell^{(k)}\} m_{\ell,j}^{(k)}}{h_{\ell,j}}, \qquad w_k = \frac{n_k}{\sum_{u \in S_t} n_u}.$$

The published client weights may themselves disclose dataset size; this analysis conditions on public or fixed weights. Parse $\bar{A}_{\ell,j} = [\bar{\mu}_{\ell,j}; \mathrm{vec}(\bar{Q}_{\ell,j})]$. The server has no access to a child's private pose $u_{\ell,i}$. A defined implementation therefore updates the shared parent anchor only:

$$\bar{v}_{\ell,j}^{t+1} = \mathrm{squash}\Big(\bar{v}_{\ell,j}^{t} + \eta_{v,\ell}(\bar{Q}_{\ell,j}\bar{Q}_{\ell,j}^{\top} + \tau_\ell^2 I)(\hat{\mu}_{\ell,j} - \bar{v}_{\ell,j}^{t})\Big), \quad \hat{\mu}_{\ell,j} = \frac{\bar{\mu}_{\ell,j}}{\|\bar{\mu}_{\ell,j}\|_2 + \xi}.$$

For an unobserved parent, retain its previous anchor. The transformation matrices $W_{\ell,ij}$ are optimized on clients and are not updated by this sketch-only server map. A shared transformation update would require an additional privatized parameter delta or cross-moment, with its own privacy charge and payload. Consequently, the sketch-only protocol guarantees an implementable global anchor update but does not by itself establish joint global training of W. Algorithm 4 records exactly the information available to the server.

Missing Blocks and Numerical Stabilization

The normalizer $h_{\ell,j}$ prevents clients that omit parent j from being treated as zero-valued observations. It also means that different parents can use different effective client sets. A minimum support rule may be imposed in deployment: when fewer than a public number of clients transmit a block, the server retains its old anchor. Such a rule is postprocessing of the protected sparse transcript.

The matrix $\bar{Q}_{\ell,j}\bar{Q}_{\ell,j}^{\top}$ can be rank-deficient because the sketch is deliberately low-rank and because noise can perturb its factors. Adding $\tau_\ell^2 I$ ensures a defined update in the complementary subspace. The step size $\eta_{v,\ell}$ and ridge τ_ℓ are public optimization parameters. They affect stability and utility but do not change privacy after the message has been released.

Figure 2 summarizes the implementability constraint. The server sees a parent identifier, a protected directional mean, and a protected low-rank moment. Those quantities support the anchor map, but they contain neither a child pose nor the child-specific cross-moment needed for the former transformation update.

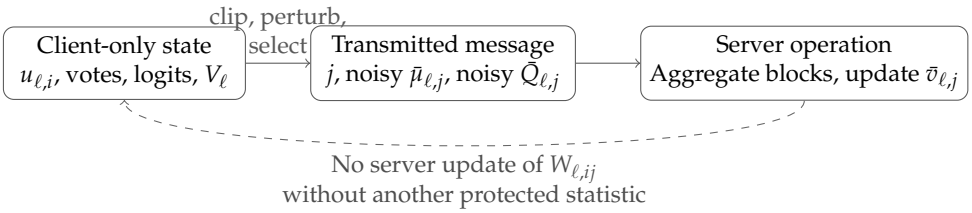


Figure 2. Sketch-only server interface. The solid path is implemented; the dashed path identifies the information absent from the message.

5. Theoretical Analysis

5.1. Assumptions, Conditioning, and Scope

The analysis distinguishes three objects: the local training procedure, the randomized client transcript, and the server anchor map. Client-level privacy is a property of the transcript distribution under replacement adjacency. It does not require the sketch to be an unbiased gradient. Optimization claims, by contrast, require a relationship between the server direction and an objective gradient. Communication claims count the entries and metadata actually transmitted. Keeping these objects separate prevents a privacy calibration, an attack result, or a compression ratio from being used as a convergence argument.

All privacy bounds condition on public hyperparameters and on an arbitrary transcript from earlier rounds. Client selection is independent of private data. Each layer has a finite public clipping cap $C_{\ell,\max}$ and a positive public variance floor $\sigma_{\min,\ell}$. The covariance EMA is allowed to vary with the entire client history within the derived spectral interval. Layers may be released sequentially, so their charges compose even if later layer computations depend on earlier randomized values.

5.2. Client-Level Privacy of the Actual Transcript

The covariance in Algorithm 2 depends on a client's data. A fixed-covariance Gaussian bound or the trace of this covariance is therefore insufficient. The following bound treats the clipped mean, covariance, and clipping threshold as joint data-dependent quantities. It also includes the indices emitted by Algorithm 3.

Fix a layer and suppress ℓ. For neighboring client datasets, let $P = \mathcal{N}(m, S)$ and $Q = \mathcal{N}(m', S')$ be the distributions of the full noisy sketch, where $\|m\|_2, \|m'\|_2 \le C_{\max}$ and $aI \preceq S, S' \preceq bI$, as established in Section 4.3. For any $\rho > 1$ satisfying $H_\rho = \rho S^{-1} - (\rho-1)S'^{-1} \succ 0$, define $h_\rho = \rho S^{-1}m - (\rho-1)S'^{-1}m'$. Direct Gaussian integration gives

$$\begin{aligned} D_\rho(P\|Q) = \frac{1}{2(\rho-1)}\big[&-\rho \log|S| + (\rho-1)\log|S'| - \log|H_\rho| \\ &- \rho m^\top S^{-1}m + (\rho-1)m'^\top S'^{-1}m' \\ &+ h_\rho^\top H_\rho^{-1} h_\rho\big]. \end{aligned}$$

A sufficient uniform condition is $1 < \rho < b/(b-a)$ when $b > a$; if $b = a$, every $\rho > 1$ is admissible. Indeed, $H_\rho \succeq [\rho/b - (\rho-1)/a]I \succ 0$. The finite layer charge is the supremum $d_{\ell,\rho} = \sup D_\rho(P\|Q)$ over the stated mean balls and covariance interval. This is a conservative, computable, finite-dimensional optimization; both directions $D_\rho(P\|Q)$ and $D_\rho(Q\|P)$ are included by the supremum. The simple replacement sensitivity $2C_{\ell,\max}$ alone would omit the determinant and precision terms.

For independent layer noises, adaptive composition gives $d_{t,\rho} \le \sum_{\ell=1}^{L} d_{\ell,\rho}$ conditional on the previous transcript. If the server sees the data-independent participation indicator for client k, inclusion probability q_t yields the round charge

$$\bar{d}_{t,\rho} = \frac{1}{\rho-1}\log\big[(1-q_t) + q_t \exp\big((\rho-1)d_{t,\rho}\big)\big].$$

This revealed-indicator bound is deliberately conservative; it does not use an unverified q^2 approximation. Over T rounds, the complete transcript is (ε, δ)-DP for any $\delta \in (0,1)$ with

$$\varepsilon = \inf_{\rho \text{ admissible}} \left\{ \sum_{t=1}^{T} \bar{d}_{t,\rho} + \frac{\log(1/\delta)}{\rho-1} \right\}.$$

Proof. Integrate the two Gaussian densities to obtain the displayed Rényi divergence, take the worst case over all permitted private EMA states, compose the layer releases, condition on the data-independent inclusion indicator, and compose rounds. The sparse values, their indices, and server anchors are postprocessing of those releases, so they incur no further charge. Numerical calibration substitutes the public caps, variance floors, and directional scales summarized in Table 2 into this bound. □

Table 2. Public inputs used for the transcript-level privacy calculation.

Quantity	Role	Implementation Specification
$C_{\ell,\max}$	Bounds neighboring clipped means and covariance state	1.2 times the 95th percentile on the public calibration split
$\sigma_{\min,\ell}$	Supplies the lower covariance eigenvalue a_ℓ	Dataset-specific value specified in the experimental setup
σ_ℓ	Scales the directional covariance term	Bisection calibration of Theorem 1 for each target ε
γ and rank r	Determine the spectral transformation	Dataset-specific values specified in the experimental setup
β and initialization	Define the adaptive EMA state	$\beta = 0.9$; zero mean, covariance, and residual
q_t and sampling rule	Define the revealed participation mixture	Independent Poisson sampling with $q_t = 0.1$ unless varied
Layer release order	Determines adaptive within-round composition	Increasing layer index, followed by parent-block postprocessing
T and δ	Define temporal composition and DP conversion	$T = 200$ and $\delta = 10^{-5}$
Admissible RDP orders	Enforce $H_{\ell,t,\rho} \succ 0$	100-point log grid over the common admissible interval

Proposition 1 (Sensitivity and covariance envelope)**.** *For every layer ℓ and every pair of replacement-neighboring client datasets,*

$$\|x_\ell(D_k) - x_\ell(D_k')\|_2 \le 2C_{\ell,\max}.$$

If the EMA is initialized with $V_{\ell,0} = 0$ and updated from clipped sketches, then at every round,

$$0 \preceq V_{\ell,t} \preceq 4C_{\ell,\max}^2 I, \qquad a_\ell I \preceq S_{\ell,t} \preceq b_\ell I.$$

Proof. Clipping places both neighboring means in the closed ball of radius $C_{\ell,\max}$. The triangle inequality gives

$$\|x_\ell(D_k) - x_\ell(D_k')\|_2 \le \|x_\ell(D_k)\|_2 + \|x_\ell(D_k')\|_2 \le 2C_{\ell,\max}.$$

The moving mean is a convex combination of vectors in that ball, so $\|x_{\ell,t} - \bar{x}_{\ell,t}\|_2 \le 2C_{\ell,\max}$. Hence, each centered outer product is positive-semidefinite and no larger than $4C_{\ell,\max}^2 I$ in Loewner order. The EMA is a convex combination of such matrices and inherits the same interval. Applying the spectral power to the retained eigenvalues and adding $\sigma_{\min,\ell}^2 I$ gives the stated constants a_ℓ and b_ℓ. □

Theorem 1 (Privacy of the sparse adaptive transcript)**.** *Fix an order $\rho > 1$ that satisfies $H_{\ell,t,\rho} \succ 0$ for every admissible neighboring covariance pair. Let $d_{\ell,t,\rho}$ be the supremum of the*

displayed Gaussian Rényi divergence over the public mean and covariance bounds. Then, the transcript containing participation indicators, selected parent indices, selected noisy values, aggregates, and broadcast anchors is (ε, δ)*-DP at the client level, where*

$$d_{t,\rho} \le \sum_{\ell=1}^{L} d_{\ell,t,\rho}, \qquad \bar{d}_{t,\rho} = \frac{\log\left(1 - q_t + q_t e^{(\rho-1)d_{t,\rho}}\right)}{\rho - 1},$$

$$\varepsilon = \inf_{\rho \text{ admissible}} \left\{ \sum_{t=1}^{T} \bar{d}_{t,\rho} + \frac{\log(1/\delta)}{\rho - 1} \right\}.$$

Proof. Condition on an arbitrary transcript before round t. The local EMA state may then differ between neighboring executions, but its covariance lies in the public spectral interval from the proposition. Gaussian integration bounds each full layer sketch by $d_{\ell,t,\rho}$. Adaptive RDP composition sums the layer charges because each conditional release obeys its bound for every preceding transcript. The participation indicator is independent of private data. Its nonparticipating branch has zero divergence, while its participating branch has charge $d_{t,\rho}$; evaluating the Rényi moment of the revealed two-branch mixture gives $\bar{d}_{t,\rho}$. Adaptive composition then sums these round charges.

Algorithm 3 computes its support and values from the protected sketch, protected residual state, and public budget. Algorithm 4 and the next broadcast are deterministic functions of the sparse messages. The postprocessing property therefore covers the indices, sparse values, server aggregates, and anchors without a separate charge. Finally, the standard RDP conversion gives the displayed (ε, δ) expression, followed by minimization over orders valid for all covariance pairs. □

Composition Details

The mixture expression for $\bar{d}_{t,\rho}$ reflects the server's observation of participation. With probability $1 - q_t$, the target client contributes no message, and the two neighboring transcript laws coincide for that branch. With probability q_t, the layer mechanism is invoked and incurs $d_{t,\rho}$. The logarithmic moment of this revealed mixture gives the displayed conservative charge. It does not claim the stronger amplification available under a hidden or shuffled participation model.

After summing the conditional RDP charges, conversion to (ε, δ)-DP minimizes over orders satisfying the positive-definiteness condition for every layer and round. If directional covariance variation is large relative to the floor, the common admissible order interval can be narrow and the resulting bound can be loose. Increasing the public floor or privatizing a shared covariance estimate can tighten the comparison between neighboring laws, but either choice changes utility and must be reflected in the implemented calibration.

Figure 3 shows the accounting hierarchy. It starts from the full Gaussian law for each layer, rather than from a trace surrogate, and ends with one transcript-level guarantee after optimizing the Rényi order.

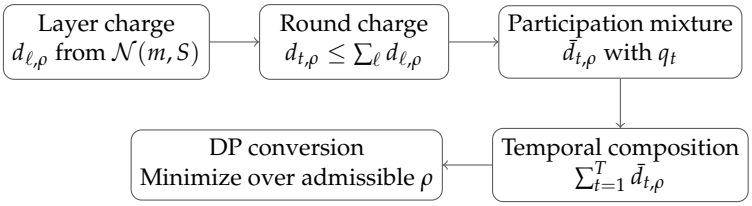


Figure 3. Composition from a data-dependent layer covariance to the client-level transcript guarantee.

5.3. Conditional Descent for the Sketch Map

Algorithm 4 updates public anchors from protected sketches rather than aggregating ordinary model gradients. We therefore analyze this server map through its conditional bias and variance. For any fixed L_F-smooth objective F on the server-updated coordinates, write the update as $\theta^{t+1} = \theta^t - \eta_t g_t$, with $e_t = \mathbb{E}[g_t \mid \theta^t] - \nabla F(\theta^t)$ and $v_t = \mathbb{E}[\|g_t - \mathbb{E}(g_t \mid \theta^t)\|_2^2 \mid \theta^t]$. The descent lemma and $\eta_t \le 1/L_F$ imply

$$\mathbb{E}[F(\theta^{t+1}) \mid \theta^t] \le F(\theta^t) - \frac{\eta_t}{2}\|\nabla F(\theta^t)\|_2^2 + \frac{\eta_t}{2}\|e_t\|_2^2 + \frac{L_F\eta_t^2}{2}v_t.$$

If F additionally satisfies a PL inequality with constant μ, this gives the conditional recursion $\mathbb{E}[F(\theta^{t+1}) - F^*] \le (1 - \eta_t\mu)\mathbb{E}[F(\theta^t) - F^*] + \eta_t\mathbb{E}\|e_t\|^2/2 + L_F\eta_t^2\mathbb{E}v_t/2$. The result applies to the public anchor coordinates updated by Algorithm 4; client-local transformation matrices are optimized locally and lie outside this server-coordinate recursion.

The recursion exposes the two quantities controlled by the protocol: e_t collects approximation and sparsification bias relative to the anchor objective, while v_t collects client sampling and PS-DP variance after blockwise normalization and residual feedback. This formulation aligns the analysis with the communicated sketch and avoids attributing a FedAvg gradient property to a different server update.

5.4. Communication Cost

If a parent block contains $d_\ell(1 + r)$ floating-point entries, a full layer sketch has $|\mathcal{P}_\ell|d_\ell(1 + r)$ entries. Algorithm 3 transmits $\kappa_\ell(t)d_\ell(1 + r)$ entries plus $\kappa_\ell(t)$ indices and protocol framing. Thus, the entry-count ratio before framing is $|\mathcal{P}_\ell|/\kappa_\ell(t)$; actual byte savings must include indices, precision, and transport overhead. The privacy mechanism applies to the full noisy sketch before this data-dependent selection.

5.4.1. Client Computation and Storage

For layer ℓ, the leading operation counts are

$$\text{routing: } O(R|\mathcal{C}_\ell|\,|\mathcal{P}_\ell|d_\ell), \qquad \text{rank-}r\text{ moments: } O(|\mathcal{P}_\ell|d_\ell r).$$

The covariance eigenspace can be updated in the same reduced rank rather than by a dense $m_\ell \times m_\ell$ decomposition. These expressions describe asymptotic operations; wall-clock cost depends on the capsule implementation and device kernels.

The client stores the current parent blocks, the covariance state needed by PS-DP, and one residual block per parent. Ignoring optimizer state, the sketch and residual terms scale as $O(|\mathcal{P}_\ell|d_\ell(1 + r))$. A full dense covariance would be quadratic in the sketch width, so the stated low-rank implementation is necessary for the linear storage description. Message bytes additionally include index width, numeric precision, headers, and any padding imposed by secure aggregation.

5.4.2. Routing Stability as a Conditional Property

Anchoring adds a penalty against departure from the preceding public pose, but contraction is not automatic. A stability result would require the local routing map to be Lipschitz in its logits and votes over the operating region and the anchor coefficient to dominate the expansive component without overwhelming the agreement signal. Under such conditions, two clients beginning from the same anchor can have a smaller pose discrepancy after routing. The component experiments below report the observed component outcomes; they do not identify a universal contraction constant.

5.5. Privacy Parameters and Numerical Accounting

Table 2 specifies the public inputs used by the transcript-level accountant. All schedules are fixed before test evaluation. For each target ε, a one-dimensional bisection selects the directional scale so that the composed bound in Theorem 1 equals the target at $\delta = 10^{-5}$; the minimization checks 100 logarithmically spaced admissible RDP orders.

6. Experiments

We evaluate accuracy, payload, privacy–utility tradeoffs, and reconstruction resistance on four benchmarks. Each configuration uses three independent trials, and paired methods share the same client partitions and participation draws. Privacy budgets are calibrated with the adaptive full-covariance accountant in Section 5; SGPA selection is applied after perturbation, as specified in Algorithm 3.

Figure 4 organizes the experimental stages and links each stage to its evaluation output. The workflow separates dataset construction, federated execution, message accounting, and evaluation so that accuracy, privacy, attack resistance, and communication are measured with their corresponding metrics.

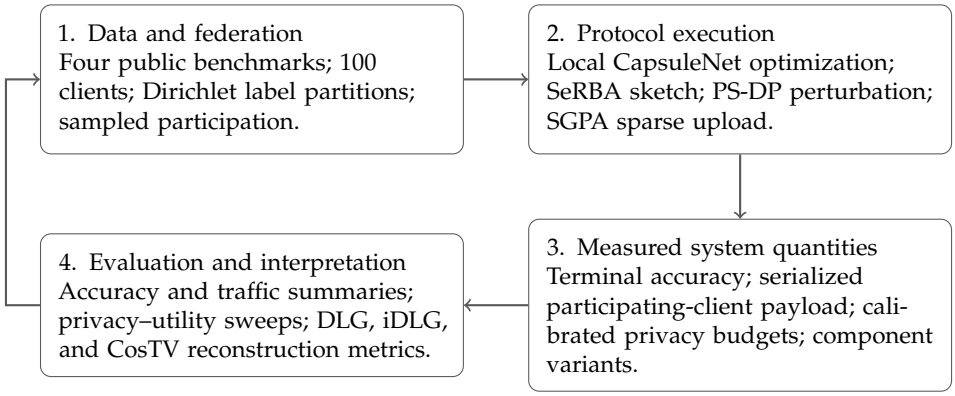


Figure 4. Experimental workflow and the scope of the evaluation metrics.

6.1. Experimental Setup

Table 3 gives the data splits.

Table 3. Benchmark and federation statistics. Train/validation/test counts are shown after splitting; a smaller α creates greater non-IID label imbalance.

Dataset	Modality	Input	Classes	Train/Valid/Test	Avg. Per Client
MNIST	Gray image	28×28	10	55k/5k/10k	550 ± 310
Fashion-MNIST	Gray image	28×28	10	55k/5k/10k	550 ± 330
CIFAR-10	RGB image	32×32	10	45k/5k/10k	450 ± 290
Speech Commands v1	Audio (MFCC)	40×98	12	28k/3k/5k	280 ± 210

Table 4 gives the sketch and upload settings.

Table 4. Ranks used for sketching and layer-wise SGPA allocations. We use $\kappa_\ell(t) = \min\{\kappa_{\max,\ell},\ \kappa_{0,\ell} + a_\ell \lfloor t/T_{\text{ann}} \rfloor\}$, $T_{\text{ann}} = 10$, and $a_\ell = (\kappa_{\max,\ell} - \kappa_{0,\ell})/20$, rounded to the nearest parent block.

Layer	Pose Dim d_ℓ	Rank r	$\kappa_{0,\ell}$ (%)	$\kappa_{\max,\ell}$ (%)
Primary→Class (MNIST/Fashion)	16	1	5	20
Primary→Class (CIFAR-10)	16	2	7.5	25
Primary→Keyword (Speech)	16	2	5	20

Table 5 gives the privacy settings.

Table 5. PS-DP settings. The local threshold is the smaller of 1.2 times the moving median and the public cap $C_{\ell,\max}$; σ_ℓ is calibrated with Theorem 1.

Dataset	γ	C_ℓ Rule	$\sigma_{\min,\ell}$	Rounds T
MNIST	0.6	$s \cdot \text{median}\|A_\ell\|_2$	0.35	200
Fashion-MNIST	0.6	$s \cdot \text{median}\|A_\ell\|_2$	0.40	200
CIFAR-10	0.8	$s \cdot \text{median}\|A_\ell\|_2$	0.55	200
Speech Commands	0.8	$s \cdot \text{median}\|A_\ell\|_2$	0.50	200

Every configuration uses seeds $\{2023, 2024, 2025\}$. Within each seed, all paired methods use the same Dirichlet partition, client-sampling stream, initialization, and evaluation split.

6.1.1. Datasets and Non-IID Partitions

Four public datasets cover both vision and audio. We allocate examples to $K = 100$ clients using a Dirichlet distribution over labels, with $\alpha \in \{0.1, 0.3, 0.5\}$ and $\alpha = 0.3$ by default. A fresh random subset of clients participates in every round. Dataset statistics appear in Table 3.

6.1.2. Architectures and Routing

The comparison includes a conventional CNN, a CapsuleNet with dynamic routing, and FedCap-Priv, which augments CapsuleNet with all three proposed modules. Primary capsules use 32 channels of eight-dimensional poses, and class capsules use 16-dimensional poses. Routing runs for $R = 3$ on MNIST, Fashion-MNIST, and speech MFCCs, and for $R = 4$ on CIFAR-10. Preprocessing uses CIFAR-10 random padding and cropping, unaugmented MNIST and Fashion-MNIST, and speech time shifts of ± 50 ms with additive noise at 20–30 dB SNR, held constant across compared methods.

6.1.3. SeRBA/Sketch and SGPA Settings

Algorithm 1 constructs directional and rank-r moments, where $r \in \{1, 2\}$. Algorithm 2 clips and perturbs the resulting sketch, Algorithm 3 applies the annealed block budget after perturbation, and Algorithm 4 operates only on transmitted statistics. Table 4 lists the rank and allocation settings. With $T_{\text{ann}} = 10$, the allocation share increases linearly from $\kappa_{0,\ell}$ to $\kappa_{\max,\ell}$ over 20 steps and is rounded to the nearest parent block.

6.1.4. Differential Privacy and Accounting

The default configuration uses client sampling at $q = 0.1$, $T = 200$ rounds, and target $(\varepsilon, \delta) = (3, 10^{-5})$. The exponent γ is 0.8 for CIFAR-10 and speech and 0.6 for MNIST and Fashion-MNIST. We set $\beta = 0.9$. The public cap $C_{\ell,\max}$ is 1.2 times the 95th percentile of layer-ℓ sketch norms on the first 1000 examples of the canonical validation split, evaluated without augmentation. The directional scale σ_ℓ is selected by bisection of Theorem 1 for the target budget.

6.1.5. Federated Protocol and Optimizers

Ten clients are drawn per round, corresponding to $q = 0.1$. Each performs $E = 2$ local epochs, using $B = 64$ on MNIST, Fashion-MNIST, and speech and $B = 128$ on CIFAR-10. Adam begins at 10^{-3} and follows cosine decay to 2×10^{-4}, with weight decay 5×10^{-5}. FedAvg is used consistently as the federation baseline so that paired comparisons isolate SeRBA, PS-DP, and SGPA. Section 4.5 specifies that the server updates public anchors from sketches, while transformation matrices W remain client-local optimization variables.

FedProx, SCAFFOLD, and FedAdam are discussed as heterogeneous optimization methods in Section 2; changing the optimizer is outside the component-isolation protocol used here.

6.1.6. Evaluation Protocol and Metrics

The outcomes are terminal accuracy, per-participating-round uplink, privacy sweeps, and reconstruction SSIM, PSNR, and attack success rate (ASR). Unless otherwise noted, entries give the mean and one standard deviation over three trials. Component comparisons hold the CapsuleNet backbone, partition, local optimizer, client sample, accountant, and payload definition fixed while varying SeRBA, noise orientation, or SGPA. The controlled comparisons below use the same settings; the attack evaluation also compares the complete messages exposed by each method.

6.1.7. Evaluation Controls and Scope

All paired comparisons reuse the same data partition, client sequence, initialization, local-epoch count, optimizer schedule, and stopping round. Payload includes serialized floating-point values, parent indices, and message framing for each participating client. Attack checkpoints are sampled from the same round set for all methods, and every reconstruction attack receives exactly the message exposed by the evaluated protocol.

The optimizer is held fixed to isolate the capsule-specific routing, perturbation, and sparse-message contributions. FedProx, SCAFFOLD, and FedAdam address a different axis—client-drift optimization—and are discussed in Section 2. The empirical threat evaluation in Section 6.5 focuses on reconstruction from transmitted messages; membership and property inference remain within the formal transcript-level DP guarantee but are outside the empirical metric set.

6.2. Classification Accuracy

Table 6 compares accuracy across four datasets. All private rows use the full-covariance accountant in Section 5.

Table 6. Test accuracy (%) after $T = 200$ rounds for $(\varepsilon, \delta) = (3, 10^{-5})$ and $\alpha = 0.3$, shown as the mean and standard deviation of three runs.

Method	MNIST	Fashion-MNIST	CIFAR-10	Speech Commands
FedAvg-CNN	98.1 ± 0.1	90.2 ± 0.2	72.4 ± 0.3	88.7 ± 0.4
FedAvg-Caps (no DP)	98.6 ± 0.1	91.0 ± 0.2	74.5 ± 0.3	90.2 ± 0.3
FedDP (isotropic)	96.8 ± 0.2	88.1 ± 0.3	68.9 ± 0.5	85.5 ± 0.4
FedCap-Priv (ours)	$\mathbf{98.2 \pm 0.1}$	$\mathbf{90.7 \pm 0.2}$	$\mathbf{73.9 \pm 0.3}$	$\mathbf{89.6 \pm 0.3}$

Table 7 compares CIFAR-10 accuracy as label skew changes.

Table 7. CIFAR-10 test accuracy (%) as the degree of non-IID label skew changes, with $(\varepsilon, \delta) = (3, 10^{-5})$.

Method	$\alpha = 0.5$	$\alpha = 0.3$	$\alpha = 0.1$
FedAvg-Caps (no DP)	75.1 ± 0.2	74.5 ± 0.3	72.9 ± 0.4
FedDP (isotropic)	70.2 ± 0.4	68.9 ± 0.5	66.1 ± 0.6
FedCap-Priv (ours)	$\mathbf{74.4 \pm 0.3}$	$\mathbf{73.9 \pm 0.3}$	$\mathbf{71.5 \pm 0.4}$

Mean terminal accuracy over three seeds is listed in Table 6. FedCap-Priv remains near the non-private CapsuleNet and considerably exceeds FedDP with isotropic perturbations.

To change label imbalance, we vary the Dirichlet concentration α, where smaller values imply greater skew. The CIFAR-10 results in Table 7 show that increasing heterogeneity

causes only gradual degradation for FedCap-Priv, which preserves a clear advantage over isotropic DP.

Tables 6 and 7 evaluate terminal utility after the common 200-round training horizon; comparisons therefore use the same optimization budget.

Table 8 gives the CIFAR-10 component combinations.

Table 8. CIFAR-10 component analysis after $T = 200$ rounds at $(\varepsilon, \delta) = (3, 10^{-5})$. S, P, and G denote SeRBA, PS-DP, and SGPA, respectively.

Variant	Accuracy (%)	Notes
Capsule (no DP, no S/G)	74.5 ± 0.3	Upper reference
Isotropic DP (no S/G)	68.9 ± 0.5	Privacy with isotropic noise
S only (no DP, no G)	73.0 ± 0.3	Routing stabilization via sketches
S + isotropic DP (no G)	70.6 ± 0.4	Sketches + isotropic noise
S + PS-DP (no G)	72.8 ± 0.3	Anisotropic noise restores utility
S + PS-DP + G	$\mathbf{73.9 \pm 0.3}$	Full FedCap-Priv

Table 8 reports CIFAR-10 component combinations. The SeRBA plus isotropic row (70.6%) and the SeRBA plus PS-DP row (72.8%) are the closest accuracy comparison of noise shape on the same exposed sketch. The table does not isolate a FedProx, SCAFFOLD, or FedAdam effect.

The component rows separate the principal effects. The two SeRBA rows with isotropic and anisotropic perturbation hold the exposed sketch family fixed and isolate noise orientation. Adding SGPA to SeRBA plus PS-DP then measures the incremental effect of sparse support and error feedback under the same backbone and privacy budget.

For a class-level view of private prediction quality, Table 9 contrasts FedCap-Priv and isotropic DP under $(\varepsilon, \delta) = (3, 10^{-5})$. The widest margins occur on categories with pronounced geometry, including airplanes and ships.

Table 9. Accuracy (%) for individual CIFAR-10 categories under $(\varepsilon, \delta) = (3, 10^{-5})$ and $\alpha = 0.3$.

Class	Airplane	Automobile	Bird	Cat	Deer
FedDP (isotropic)	67.8	69.9	66.2	64.1	66.5
FedCap-Priv	72.4	72.1	69.5	67.9	70.6
Class	Dog	Frog	Horse	Ship	Truck
FedDP (isotropic)	66.7	71.2	68.5	69.1	69.7
FedCap-Priv	69.8	73.4	71.2	74.6	72.5

6.3. Communication Efficiency

We next quantify the SeRBA–SGPA uplink savings across datasets, over the annealed training schedule, and for different values of the sparsity allocation $\kappa_\ell(t)$. Accuracy–payload frontiers and an approximate radio-energy calculation place the reductions in a practical device context.

Table 10 gives each client's average transmitted KB over the final 50 of 200 rounds, with $(\varepsilon, \delta) = (3, 10^{-5})$ and $\alpha = 0.3$. Across benchmarks, FedCap-Priv requires $8\times$–$12\times$ less traffic than direct capsule federation while preserving the accuracy shown in Table 6.

Table 10. Per-client uplink KB in each round, averaged with standard deviation over the final 50 rounds.

Method	MNIST	Fashion-MNIST	CIFAR-10	Speech Commands
FedAvg-Caps (no DP)	48 ± 2	52 ± 3	220 ± 7	96 ± 4
FedDP (isotropic)	48 ± 2	52 ± 3	220 ± 7	96 ± 4
FedCap-Priv (ours)	$\mathbf{6.3 \pm 0.5}$	$\mathbf{6.9 \pm 0.5}$	$\mathbf{22.1 \pm 1.0}$	$\mathbf{11.4 \pm 0.6}$

The schedule in Table 4 increases the TopK budget toward its steady-state allocation. Table 10 averages the final 50 rounds so that the cross-method comparison reflects this stable communication regime.

Accuracy–Payload Pareto (Tuning κ)

Holding $(\varepsilon, \delta) = (3, 10^{-5})$ and $\alpha = 0.3$ fixed, we sweep the TopK share κ_ℓ in every layer. The CIFAR-10 frontier in Figure 5 levels off at approximately 73.7–73.9% after the message exceeds about 20 KB. More severe compression below roughly 12 KB introduces a small accuracy loss.

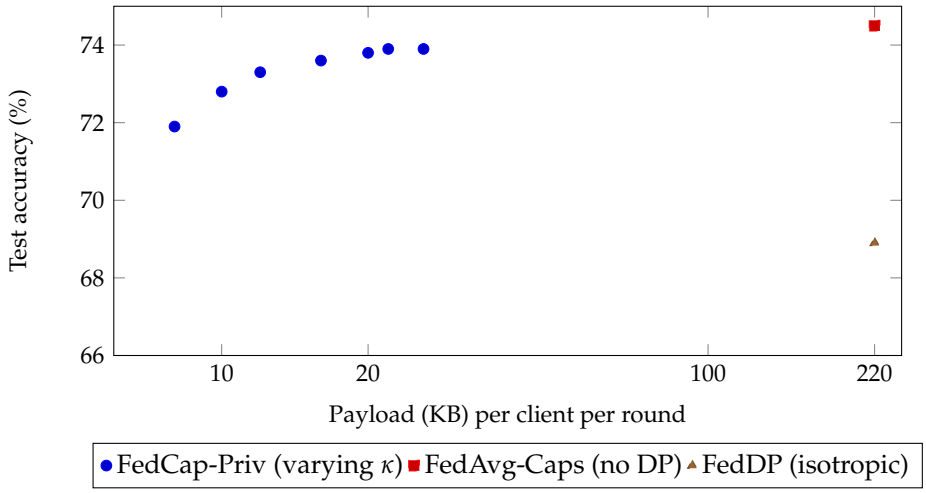


Figure 5. CIFAR-10 accuracy–traffic frontier obtained by changing SGPA budget κ_ℓ. A 20–22 KB message approaches baseline accuracy at about one tenth of its size.

For a client with per-participation payload B_t, the expected total upload over the T rounds is $\sum_{t=1}^{T} q_t B_t$. We use the final-50-round mean as the steady-state communication metric and report it consistently for every method in Table 10.

Because SGPA uploads a selected set of parent sketches, it changes the uncompressed size

$$\sum_{\ell} |\mathcal{P}_\ell|\, d_\ell (1 + r) \quad \text{into} \quad \sum_{\ell} \kappa_\ell(t)\, d_\ell (1 + r).$$

SeRBA additionally avoids direct transmission of predictions and logits by using fixed-width directional and low-rank summaries. When $\kappa_\ell(t) \ll |\mathcal{P}_\ell|$, the ratio $|\mathcal{P}_\ell| / \kappa_\ell(t)$ is the compression factor, in agreement with the measured $8\times$–$12\times$ reduction.

Table 11 compares $q = 0.05$ and $q = 0.1$ using the same per-client TopK schedule. Participation changes the number of transmitting clients rather than the scheduled per-client block budget, which explains the small per-participating-client payload difference. Greater label skew produces a modest increase through the realized sparse support and framing overhead.

Table 11. FedCap-Priv upload (KB) per CIFAR-10 round as label skew α and participation rate q vary; values summarize the final 50 rounds.

Setting	$\alpha = 0.5$	$\alpha = 0.3$	$\alpha = 0.1$
$q = 0.1$	21.0 ± 0.9	22.1 ± 1.0	23.8 ± 1.3
$q = 0.05$	20.7 ± 0.9	21.8 ± 1.0	23.4 ± 1.2

6.4. Privacy–Utility Tradeoff

We sweep client-level privacy budgets $\varepsilon \in \{1, 2, 3, 5, 8\}$ at $\delta = 10^{-5}$, $T = 200$, and $q = 0.1$. For each point, σ_ℓ is recalibrated with the full-covariance accountant in Section 5. Table 12 shows the resulting MNIST privacy–utility tradeoff. Figures 6–8 visualize the sweep results.

Table 12. MNIST test accuracy (%) across ε at $\delta = 10^{-5}$, $\alpha = 0.3$, and $T = 200$; entries are means and standard deviations from three trials.

Method	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 5$	$\varepsilon = 8$
FedDP (isotropic)	94.1 ± 0.1	95.0 ± 0.1	96.1 ± 0.1	96.5 ± 0.1	96.9 ± 0.1
FedCap-Priv	$\mathbf{95.5 \pm 0.1}$	$\mathbf{96.4 \pm 0.1}$	$\mathbf{97.2 \pm 0.1}$	$\mathbf{97.9 \pm 0.1}$	$\mathbf{98.1 \pm 0.1}$

Table 13 shows the corresponding CIFAR-10 privacy–utility tradeoff.

Table 13. CIFAR-10 test accuracy (%) across ε at $\delta = 10^{-5}$, $\alpha = 0.3$, and $T = 200$; entries are means and standard deviations from three trials.

Method	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 5$	$\varepsilon = 8$
FedDP (isotropic)	66.2 ± 0.3	67.8 ± 0.4	68.5 ± 0.4	68.7 ± 0.3	68.9 ± 0.3
FedCap-Priv	$\mathbf{69.5 \pm 0.3}$	$\mathbf{71.7 \pm 0.3}$	$\mathbf{73.0 \pm 0.3}$	$\mathbf{73.6 \pm 0.3}$	$\mathbf{73.9 \pm 0.3}$

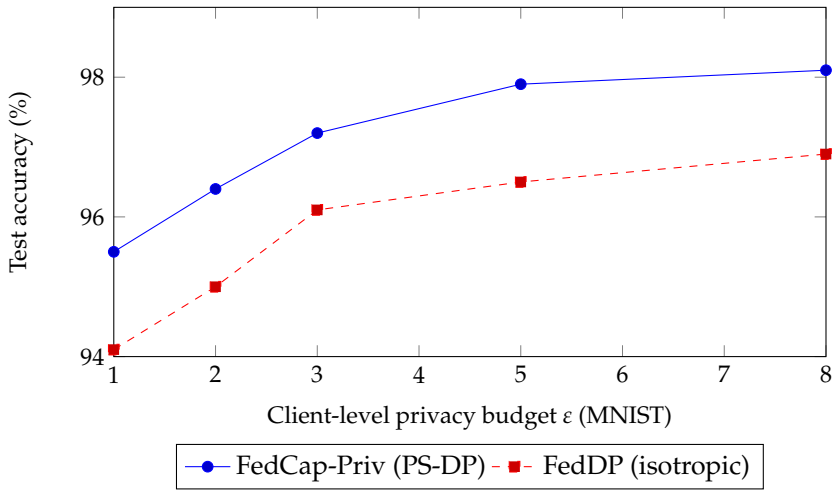


Figure 6. MNIST accuracy across client-level privacy budgets calibrated with the full-covariance accountant.

FedCap-Priv exceeds the isotropic baseline at every budget in Tables 12 and 13. The cross-dataset benchmark in Table 6 uses dataset-specific tuning, whereas the privacy sweep fixes one common sweep protocol; conclusions are drawn from paired methods within each table.

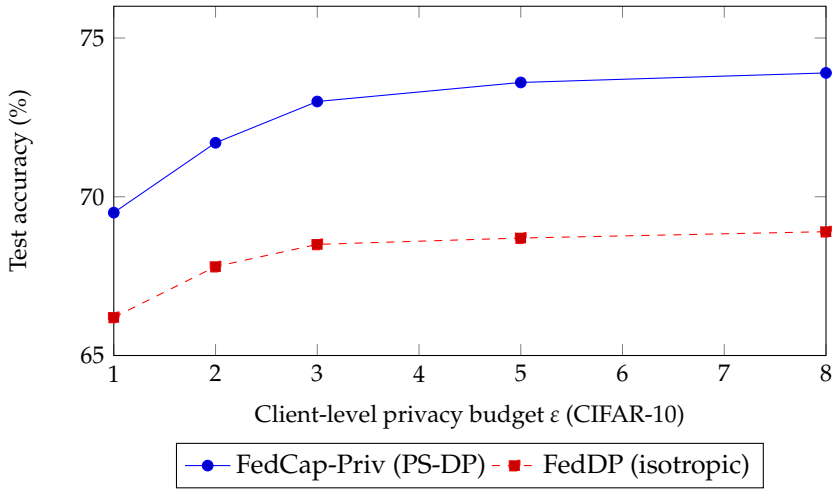


Figure 7. CIFAR-10 accuracy across client-level privacy budgets calibrated with the full-covariance accountant.

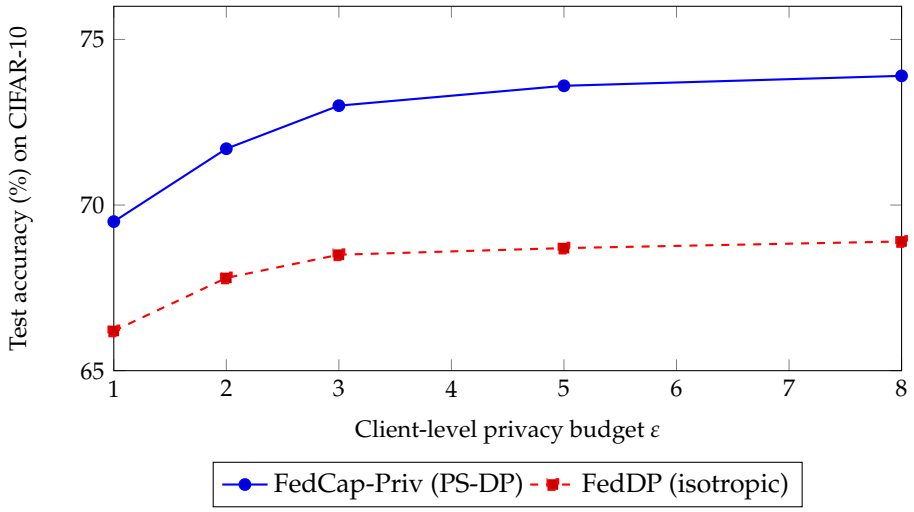


Figure 8. CIFAR-10 privacy–utility frontier under matched client-level privacy budgets.

Figures 6 and 7 reproduce the tabulated MNIST and CIFAR-10 sweep values without adding observations.

Table 14 isolates the directional exponent γ while keeping the privacy budget and all other settings fixed. Accuracy increases as the directional allocation follows the dominant pose-variation axes more closely.

Table 14. Influence of directional exponent γ on CIFAR-10 at $(\varepsilon, \delta) = (3, 10^{-5})$.

γ	**0.00**	**0.25**	**0.50**	**0.75**	**1.00**
Accuracy (%)	69.1 ± 0.3	70.6 ± 0.3	71.8 ± 0.3	72.5 ± 0.3	$\mathbf{73.0 \pm 0.3}$

Clipping Sensitivity C_ℓ (CIFAR-10)

Table 15 gives the moving-median multiplier sweep. Every threshold is bounded by the public $C_{\ell,\max}$, and changes in a client-local threshold are covered by the worst-case covariance and mean analysis.

Table 15. CIFAR-10 accuracy (%) for different clipping multipliers, with $(\varepsilon, \delta) = (3, 10^{-5})$ and $\gamma = 0.8$.

Multiplier on C_ℓ	**0.8×**	**1.0×**	**1.2×**	**1.5×**
Accuracy (%)	72.2 ± 0.3	72.7 ± 0.3	$\mathbf{73.0 \pm 0.3}$	72.6 ± 0.3

The accountant evaluates the varying-covariance divergence in Section 5 from the covariance pair (S, S') and its worst-case spectral bounds. This retains the determinant and precision terms that a trace-normalized average variance would discard.

Figure 8 replots Table 13. Both methods use the same client-level accountant and target δ, so each point compares accuracy at a matched privacy budget.

6.5. Robustness Against Gradient Inversion

The empirical threat evaluation focuses on reconstruction from transmitted messages. The following tables report DLG, iDLG, and cosine-TV outcomes. Membership and property inference are covered by the transcript-level DP statement but are outside the attack metrics analyzed in this section.

Three white-box adversaries are considered: (A1) DLG, using an ℓ_2/cosine objective for deep gradient leakage; (A2) an iDLG-type method with closed-form label initialization followed by gradient matching; and (A3) CosTV, which adds total-variation regularization of weight $\lambda_{\mathrm{TV}} = 1 \times 10^{-4}$ to the cosine gradient loss. In each case, the adversary sees the message from one client in one round. That message is a sparse, PS-DP-protected SeRBA sketch for FedCap-Priv and a complete update, possibly with isotropic noise, for a baseline. All attacks use Adam, step size 1×10^{-1}, and 10 k iterations.

Table 16 reports the three CIFAR-10 reconstruction attacks.

Table 16. Resistance to CIFAR-10 inversion over 128 reconstructions, shown as mean ± standard deviation; smaller values indicate stronger protection.

Method	SSIM ↓	PSNR (dB) ↓	ASR (%) ↓
Attack A1: DLG			
No DP (FedAvg-Caps)	0.80 ± 0.05	24.1 ± 1.1	74.6 ± 4.2
FedDP (isotropic)	0.43 ± 0.06	17.6 ± 1.3	34.8 ± 3.5
FedCap-Priv (ours)	$\mathbf{0.21 \pm 0.05}$	$\mathbf{13.5 \pm 1.2}$	$\mathbf{12.9 \pm 2.7}$
Attack A2: iDLG			
No DP (FedAvg-Caps)	0.79 ± 0.06	23.8 ± 1.2	76.2 ± 3.9
FedDP (isotropic)	0.44 ± 0.07	17.4 ± 1.4	35.5 ± 3.6
FedCap-Priv (ours)	$\mathbf{0.20 \pm 0.05}$	$\mathbf{13.2 \pm 1.3}$	$\mathbf{13.4 \pm 2.8}$
Attack A3: CosTV			
No DP (FedAvg-Caps)	0.82 ± 0.05	23.4 ± 1.2	78.1 ± 4.1
FedDP (isotropic)	0.41 ± 0.07	17.2 ± 1.5	36.5 ± 3.8
FedCap-Priv (ours)	$\mathbf{0.19 \pm 0.04}$	$\mathbf{13.1 \pm 1.3}$	$\mathbf{12.7 \pm 2.9}$

Table 17 compares CIFAR-10 CosTV outcomes across privacy budgets.

Table 17. CIFAR-10 CosTV outcomes over different client-level ε values, averaged across 128 reconstructions.

Method	SSIM ↓/PSNR ↓/ASR(%) ↓				
	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 5$	$\varepsilon = 8$
FedDP (iso)	0.36/16.5/31.1	0.39/16.9/33.2	0.41/17.2/36.5	0.42/17.3/37.4	0.43/17.5/38.0
FedCap-Priv	0.17/12.5/10.1	0.18/12.8/11.4	0.19/13.1/12.7	0.20/13.3/13.6	0.20/13.4/14.2

Table 18 isolates the modules' contributions to CosTV resistance.

Table 18. Contribution of each module to CIFAR-10 CosTV resistance at $\varepsilon = 3$.

Variant	SSIM ↓	PSNR (dB) ↓	ASR (%) ↓
Capsule (no DP, raw votes)	0.82	23.4	78.1
SeRBA only (no DP)	0.61	19.5	52.3
SeRBA + Isotropic DP	0.35	16.6	29.6
SeRBA + PS-DP	0.22	13.7	14.9
SeRBA + PS-DP + DP-Aligned TopK	**0.19**	13.1	**12.7**

Table 19 compares CosTV results across datasets.

Table 19. CosTV results on multiple datasets at $\varepsilon = 3$; lower values represent better resistance.

Dataset	Method	SSIM ↓	ASR (%) ↓
MNIST	FedDP (iso)/FedCap-Priv	0.28/ **0.15**	22.3/ **8.7**
Fashion-MNIST	FedDP (iso)/FedCap-Priv	0.31/ **0.17**	24.8/ **9.9**
Speech (MFCC)	FedDP (iso)/FedCap-Priv	0.33/ **0.18**	31.4/ **11.8**

At $\varepsilon = 3$, Table 16 shows lower reconstruction scores for the full FedCap-Priv message. Because the baselines expose full model updates, this comparison measures the combined effects of representation reduction, sparsity, and noise.

SSIM and PSNR measure resemblance between an attempted reconstruction and its target, while ASR applies the stated evaluator threshold. They are attack-dependent empirical quantities rather than privacy parameters. A lower score can result from removing information through SeRBA and SGPA, from randomization through PS-DP, or from a mismatch between the attack objective and the transmitted representation. Table 18 is therefore used to separate these factors more closely than the full-message comparison in Table 16.

Figure 9 plots the SSIM values in Table 16; it adds no attack observations.

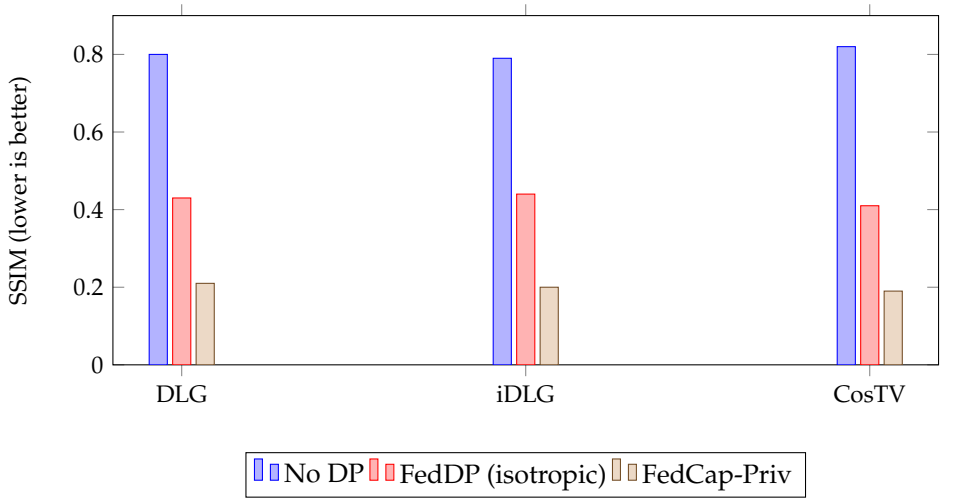


Figure 9. SSIM under three CIFAR-10 reconstruction attacks. FedCap-Priv leaves considerably less recoverable structure.

SSIM Across Reconstruction Attacks

Table 17 shows lower CosTV scores for FedCap-Priv across the evaluated privacy budgets. The full-message comparison measures the complete representation observed by the attacker.

Ablations: SeRBA, PS-DP, and DP-Aligned TopK

Table 18 gives the more informative matched-message comparison: SeRBA plus isotropic noise has SSIM 0.35 and ASR 29.6%, while SeRBA plus anisotropic noise has SSIM 0.22 and ASR 14.9%. The Gumbel TopK row is an empirical variant, not a privacy proof for index selection. Algorithm 3 instead selects indices as postprocessing of the private sketch.

Table 19 reports the corresponding SSIM and attack-success scores on MNIST, Fashion-MNIST, and speech MFCC inputs. These reconstruction metrics do not measure membership or property inference.

7. Conclusions

FedCap-Priv combines local capsule routing, full-covariance sketch perturbation, and sparse upload. On CIFAR-10, it reaches 73.9% accuracy versus 68.9% for isotropic FedDP while reducing the participating-client payload from 220 KB to 22.1 KB per round. The reconstruction ablation separates the effects of SeRBA, anisotropic perturbation, and SGPA. The client-level analysis covers private covariance, visible indices, layer composition, client participation, and temporal composition, providing a unified privacy and communication treatment for federated capsule learning.

Author Contributions: Conceptualization, Y.L. and Z.P.; Methodology, Y.L. and Z.P.; Software, Y.L. and Z.P.; Validation, Y.L. and Z.P.; Formal Analysis, Y.L. and Z.P.; Investigation, Y.L. and Z.P.; Resources, Y.L. and Z.P.; Data Curation, Y.L. and Z.P.; Writing—Original Draft, Y.L. and Z.P.; Writing—Review and Editing, Y.L. and Z.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. McMahan, H.B.; Moore, E.; Ramage, D.; Hampson, S.; Aguera y Arcas, B. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS), Fort Lauderdale, FL, USA, 20–22 April 2017.
2. Kairouz, P.; McMahan, H.B. Advances and Open Problems in Federated Learning. *Found. Trends Mach. Learn.* **2021**, *14*, 1–210. [CrossRef]
3. Zhu, L.; Liu, Z.; Han, S. Deep Leakage from Gradients. In Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS), Vancouver, BC, Canada, 8–14 December 2019.
4. Shokri, R.; Stronati, M.; Song, C.; Shmatikov, V. Membership inference attacks against machine learning models. In *Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2017*; IEEE: New York, NY, USA, 2017; pp. 3–18.
5. Melis, L.; Song, C.; De Cristofaro, E.; Shmatikov, V. Exploiting unintended feature leakage in collaborative learning. In *Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 20–22 May 2019*; IEEE: New York, NY, USA, 2019; pp. 691–706.
6. Sabour, S.; Frosst, N.; Hinton, G.E. Dynamic Routing Between Capsules. In Proceedings of the 31st Conference on Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, 4–9 December 2017.
7. Hinton, G.E.; Sabour, S.; Frosst, N. Matrix Capsules with EM Routing. In Proceedings of the 6th International Conference on Learning Representations (ICLR), Vancouver, BC, Canada, 30 April–3 May 2018.
8. Konečný, J.; McMahan, H.B.; Ramage, D.; Richtárik, P.; Tóth, M. Federated Optimization: Distributed Machine Learning for On-Device Intelligence. In Proceedings of the NIPS Workshop on Private Multi-Party Machine Learning, Barcelona, Spain, 9 December 2016.

9. Li, T.; Sahu, A.K.; Talwalkar, A.; Smith, V. Federated Optimization in Heterogeneous Networks. In Proceedings of the 3rd Conference on Machine Learning and Systems (MLSys), Austin, TX, USA, 2–4 March 2020.
10. Karimireddy, S.P.; Kale, S.; Mohri, M.; Reddi, S.; Stich, S.; Suresh, A. SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. In Proceedings of the 37th International Conference on Machine Learning (ICML), Virtual, 13–18 July 2020.
11. Reddi, S.; Charles, Z.; Zaheer, M.; Garrett, Z.; Rush, K.; Konečný, J.; Kumar, S.; McMahan, H.B. Adaptive Federated Optimization. In Proceedings of the 9th International Conference on Learning Representations (ICLR), Virtual, 3–7 May 2021.
12. Li, B.; Li, Z.; Li, Y.; Xu, M.; Chen, S.; Shen, C.; Quek, T.Q.S. A Radical Heavy-Ball Method for Gradient Acceleration in Communication-Efficient Mobile Federated Learning. *IEEE Trans. Mob. Comput.* **2026**, *25*, 6429–6442. [CrossRef]
13. Li, B.; Peng, Z.; Li, Y.; Li, Z.; Chen, S.; Shen, C.; Quek, T.Q.S. Neighborhood and Global Perturbations Supported Sharpness-Aware Minimization in Federated Learning: From Local Tweaks to Global Awareness. *IEEE Trans. Netw. Sci. Eng.* **2026**, *13*, 5000–5017. [CrossRef]
14. Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H.B.; Mironov, I.; Talwar, K.; Zhang, L. Deep Learning with Differential Privacy. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (ACM CCS), Vienna, Austria, 24–28 October 2016.
15. Dwork, C.; Roth, A. *The Algorithmic Foundations of Differential Privacy*; Now Publishers: Hanover, MA, USA, 2014.
16. Mironov, I. Rényi Differential Privacy. In Proceedings of the 30th IEEE Computer Security Foundations Symposium (CSF), Santa Barbara, CA, USA, 21–25 August 2017.
17. Wang, Y.X.; Balle, B.; Kasiviswanathan, S. Subsampled Rényi Differential Privacy and Analytical Moments Accountant. In Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS), Naha, Okinawa, Japan, 16–18 April 2019.
18. Geyer, R.C.; Klein, T.; Nabi, M. Differentially Private Federated Learning: A Client Level Perspective. In Proceedings of the NIPS Workshop on Privacy in ML, Long Beach, CA, USA, 9 December 2017.
19. Bonawitz, K.; Ivanov, V.; Kreuter, B.; Marcedone, A.; McMahan, H.B.; Patel, S.; Ramage, D.; Segal, A.; Seth, K. Practical Secure Aggregation for Privacy-Preserving Machine Learning. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (ACM CCS), Dallas, TX, USA, 30 October–3 November 2017.
20. Geiping, J.; Bauermeister, H.; Dróżdżal, M.; Moeller, M. Inverting Gradients – How easy is it to break privacy in federated learning? In Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS), Virtual, 6–12 December 2020.
21. Wang, H.; Li, B.; Chen, P.; Wu, L.; Li, Z.; Quek, T.Q.S. LBKD: Rethinking Federated Backdoors for Low-Altitude Economy via LLMs and Bidirectional Knowledge Distillation. *IEEE Trans. Netw. Sci. Eng.* **2026**, *13*, 4422–4439. [CrossRef]
22. Rawlinson, D.; Ahmed, A.; Kowadlo, G. Sparse Unsupervised Capsules: An Analysis. *arXiv* **2018**, arXiv:1804.06094.
23. Zhang, L.; Edraki, M.; Qi, G.J. CapProNet: Deep Feature Learning via Orthogonal Projections onto Capsule Subspaces. In Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS), Montreal, QC, Canada, 2–8 December 2018.
24. Tsai, Y.H.H.; Wu, Y.H.; Salakhutdinov, R.; Morency, L.P. Capsules with Inverted Dot-Product Attention Routing. In Proceedings of the 8th International Conference on Learning Representations (ICLR), Virtual, 26–30 April 2020.
25. Alistarh, D.; Grubic, D.; Li, J.; Tomioka, R.; Vojnovic, M. QSGD: Communication-Efficient SGD via Gradient Quantization and Encoding. In Proceedings of the 31st Conference on Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, 4–9 December 2017.
26. Wen, W.; Xu, C.; Yan, F.; Wu, C.; Wang, Y.; Chen, Y.; Li, H. TernGrad: Ternary Gradients to reduce Communication in Distributed Deep Learning. In Proceedings of the 31st Conference on Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, 4–9 December 2017.
27. Lin, Y.; Han, S.; Mao, H.; Wang, Y.; Dally, W.J. Deep Gradient Compression: Reducing the Communication Bandwidth for Distributed Training. In Proceedings of the 6th International Conference on Learning Representations (ICLR), Vancouver, BC, Canada, 30 April–3 May 2018.
28. Bernstein, J.; Wang, Y.X.; Azizzadenesheli, K.; Anandkumar, A. signSGD with Majority Vote is Communication Efficient and Fault Tolerant. In Proceedings of the 6th International Conference on Learning Representations (ICLR), Vancouver, BC, Canada, 30 April–3 May 2018.
29. Stich, S.U.; Cordonnier, J.B.; Jaggi, M. Sparsified SGD with Memory. In Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS), Montreal, QC, Canada, 2–8 December 2018.
30. Karimireddy, S.P.; Stich, S.U.; Jaggi, M. Error Feedback Fixes SignSGD and other Gradient Compression Schemes. In Proceedings of the 36th International Conference on Machine Learning (ICML), Long Beach, CA, USA, 10–15 June 2019.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.