FROM UPDATE GEOMETRY TO MODEL RELATEDNESS: INDEX-ONLY SALIENT SUPPORT SIMILARITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Model similarity supports personalization, clustering, and retrieval, but standard methods typically require high-dimensional real-valued weights, updates, or representations. We study a layer-normalized Top-K index-support signal for client and checkpoint relatedness in a shared parameter coordinate system, which we call Salient Parameter Overlap Similarity (*SPOS*). Each client computes update-magnitude salience locally, normalizes salience within parameter blocks, and communicates only the identities of its selected coordinates. With $K/M = 0.1$ and 32-bit coordinate IDs, the similarity-plane payload is one tenth that of an M-entry FP32 vector. Across four vision, language, and speech model–dataset pairs, *SPOS* provides useful affinities for clustered personalization and neighbor retrieval. With label marginals held exactly fixed, the signal remains informative under feature shift, concept shift, and unequal client data and computation. For checkpoints adapted from a shared reference, *SPOS* obtains functional-agreement and cross-domain-transfer R@3 of 0.716 and 0.702, compared with 0.733 and 0.720 for signed dense-update cosine. These results characterize salient-support overlap as an index-only relatedness primitive when real-valued per-client affinity inputs are undesirable or unavailable.

1 INTRODUCTION

Comparing high-dimensional machine learning (ML) objects—model weights, gradients/updates, and embeddings—is a core primitive. It underpins federated learning (FL) operations such as client clustering and personalized aggregation (Ghosh et al., 2020; Fallah et al., 2020; Dinh et al., 2020), streaming/continual-learning tools such as memory retrieval and drift detection (Gama et al., 2014; Lu et al., 2018), and model analytics for provenance and similarity search (Andoni & Indyk, 2008). Yet, default similarity metrics on *full real-valued vectors* (cosine/Euclidean) become increasingly untenable at scale and under value-exposure constraints for three reasons:

(1) Cost scales with dimension. Computing and transmitting similarities over M-dimensional real vectors typically requires $\Theta(M)$ arithmetic and communication. With modern models, this quickly becomes prohibitive: even "small" CNN backbones have $M{=}10^6$–10^7 trainable parameters, mid-size architectures such as ResNet-18/50 ($\sim$11M/$\sim$25M) and VGG16 ($\sim$138M) imply per-client payloads of tens to hundreds of MB per round when exchanging 16–32 bit values; transformers further exacerbate this (e.g., BERT-base $\sim$110M, BERT-large $\sim$340M).

(2) Dense value sharing creates leakage risk. Exposing logits or intermediate activations enables inversion attacks; repeated exposures across rounds facilitate membership and property inference (Shokri et al., 2017; Melis et al., 2019); and gradients or updates can be inverted to recover training examples or labels (Zhu et al., 2019; Geiping et al., 2020). Secure aggregation can hide update values within the aggregation protocol, but does not protect a client-specific summary released through a separate relatedness channel, while computing similarity on shielded updates remains inefficient.

(3) Raw geometry is poorly calibrated. Cosine or Euclidean distances between weights or updates can distort client relatedness under layer-wise scale differences (e.g., normalization layers, weight decay, mixed precision) and optimizer-induced drift across rounds without careful normalization or tuning (Loshchilov & Hutter, 2019; Micikevicius et al., 2018). Over-parameterized networks exhibit a head–tail importance structure: a small set of coordinates carries most first-order training signal, while the tail contributes less to model comparison (Michel et al., 2019; Voita et al., 2019; Gale et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2019; Lee et al., 2019). Models trained on similar distributions tend to share importance rankings, consistent with representational-similarity and FL client-relatedness findings (Kornblith et al., 2019).

These observations motivate an index-only view of client-update relatedness from a shared reference model: represent each model by the *indices* of its most salient parameters, rather than by real-valued weights or updates. Each client computes update-magnitude salience from its local update, selects the Top-K coordinates, and reports only their integer indices. Related clients are then identified by the overlap of these salient-coordinate supports: clients whose data drive learning on similar parameter subsets receive higher similarity. We call this signal *Salient Parameter Overlap Similarity (*SPOS*)*. *SPOS* is *index-only*: salience is computed locally, and only coordinate identifiers are communicated–not weights, gradients, logits, activations, or salience values. This design has three benefits. **First**, *SPOS* reduces the similarity-plane representation: when relatedness is needed without full-vector transmission, communication drops from $\Theta(M)$ real values to $\Theta(K)$ indices, with $K/M = 0.1$; when full updates are already transmitted under protected aggregation, *SPOS* adds only a $\Theta(K)$ sketch. **Second**, *SPOS* is invariant to global positive rescaling and can be less sensitive to moderate scale heterogeneity than value-based similarities, since it compares salient supports rather than real-valued magnitudes. **Third**, *SPOS* removes real-valued magnitudes and signs from the similarity message. The resulting support is still a deterministic, client-specific disclosure.

Our contribution. We study whether layer-normalized index-only salient-support overlap can serve as a benign client-relatedness signal when individual update values are shielded. Importantly, ***SPOS* does not aim to dominate value-based similarities such as cosine similarity. Its goal is to retain useful relatedness while removing real-valued per-client update magnitudes and signs from the similarity message.** *SPOS* addresses a key FL bottleneck: personalization needs client-level relatedness, yet plaintext value-based affinities conflict with secure aggregation, while encrypted or perturbed alternatives can be costly or unreliable.

Our findings. *SPOS* is evaluated on clustered personalization, neighbor selection, distribution alignment across vision, language, and speech. Separate shared-reference checkpoint experiments test whether the same support signal extends beyond client updates. Our main findings are:

- *Relatedness beyond label skew, within shared coordinates.* With identical label marginals, *SPOS* achieves R@5 of 0.757, 0.973, and 0.597 under feature shift, concept shift, and quantity/compute imbalance. For checkpoints adapted from a shared reference, it reaches R@3 of 0.716 for predictive agreement and 0.702 for cross-domain transfer, versus 0.733 and 0.720 for signed dense cosine.
- *Layer-normalized utility at index-only cost.* Block-wise RMS normalization before global Top-K selection improves clustered-personalization accuracy over unnormalized support in all 16 model–split settings, with mean gains ranging from $+0.21$ to $+9.63$ percentage points. On Wav2Vec2/Dir(0.3), *SPOS* reaches R@6 $= 0.770$ versus 0.733 for dense signed-update cosine.
- *An explicit disclosure and deployment boundary. SPOS* removes magnitudes and signs from the affinity message but retains a client-specific support. At the 10% support ratio, the empirical coordinate-popularity null is substantially above the uniform 0.10 baseline—0.386, 0.261, and 0.198 on FMNIST, CIFAR-10, and 20News—showing that support exposure is not equivalent to privacy. Round-keyed coordinate remapping preserves every within-round *SPOS* overlap exactly while hiding direct coordinate labels under the stated non-collusion assumption.

2 Background

2.1 Background and Motivation

Federated Learning (FL). We consider N clients with private datasets D_i and a shared model $w \in \mathbb{R}^M$, trained by minimizing $F(w) = \sum_{i=1}^{N} p_i F_i(w), p_i = \frac{|D_i|}{\sum_{j=1}^{N}|D_j|}, F_i(w) = \frac{1}{|D_i|}\sum_{(x,y)\in D_i} \ell(w; x, y)$. At round t, the server broadcasts $w^{(t)}$, client i performs E local SGD steps to obtain $w_i^{(t+1)}$ and update $\Delta w_i^{(t)} := w_i^{(t+1)} - w^{(t)}$. FedAvg aggregates $w^{(t+1)} = w^{(t)} + \sum_i \bar{p}_i \Delta w_i^{(t)}, \bar{p}_i = \frac{|D_i|}{\sum_j |D_j|}$.

Why salient support encodes relatedness. Local update magnitude is a cheap proxy for parameter importance: large $|\Delta w_{i,m}^{(t)}|$ marks where a client's training signal concentrates. Such magnitude criteria are common in pruning and sparsification (Li et al., 2017; Gale et al., 2019) and avoid the extra

statistics of Fisher-style measures (Theis et al., 2018). Support overlap has also been used to identify backdoored clients in secure federated PEFT for LLMs (Dogani et al., 2026). Under non-IID data, client drift changes both update values and *which coordinates* become salient (Karimireddy et al., 2020; Zhao et al., 2018; Wang et al., 2023), while related clients may share important subnetworks or compatible update directions (Morcos et al., 2019; Ilharco et al., 2022).

Motivation under perturbation and encrypted computation. We empirically evaluate how standard value-shielding mechanisms affect relatedness estimation. Under clipped Gaussian perturbation at $\sigma = 1$, similarities computed from the perturbed update approach random-neighbor retrieval (R@6 ≈ 0.207), whereas a support constructed locally from the unperturbed update retains R@6 of 0.640–0.760 across FMNIST, CIFAR-10, and 20News. The two are not privacy-matched: the latter is a non-DP release and is reported as a utility reference, not as a mechanism with an (ϵ, δ) guarantee. We also benchmark dense similarity under CKKS with TenSEAL. Although similarity is preserved accurately, cost grows from 2.3 to 603.4 seconds with model size, motivating local salient-support extraction and index-only transmission over protected dense comparison. Full results are in Appx. A.

2.2 Related Work

Value-based model relatedness in FL. Most FL personalization and clustering methods estimate relatedness from *values*: update/weight distances in clustered FL and IFCA-style mixtures (Ghosh et al., 2020); validation utilities exchanging model values or updates, as in FedFomo (Zhang et al., 2021); feature-, prototype-, or CKA-based representations (Tan et al., 2022; Kornblith et al., 2019); and influence/Shapley methods using losses, gradients, or KNN distances (Koh & Liang, 2017; Ghorbani & Zou, 2019; Jia et al., 2019). All remain value-bearing, requiring updates, weights, representations, or utility surrogates.

Distribution- and embedding-based relatedness. A parallel line infers client similarity from data-side summaries: label-distribution distances and skew (Zhao et al., 2018), Wasserstein/barycenter-style (Li et al., 2024), or subspace/embedding geometry (Vahidian et al., 2023). These methods avoid sharing updates but are often computationally heavy, require auxiliary feature extraction (e.g., SVD/prototypes), and are agnostic to which *parameters* drive learning dynamics.

Sketching and compression (still value-bearing). Communication-efficient FL uses sketches or compressed updates (Rothchild et al., 2020; Rabbani et al., 2021), and DFL methods also use sketches to approximate similarity (e.g., sketch-based screening) (Rangwala et al., 2025). However, they still transmit *value-derived sketches*.

Support selection, geometry, and disclosure. IndexGuard uses overlap geometry among salient update supports to vet backdoored clients in secure federated PEFT (Dogani et al., 2026). Selecting a support does not itself ensure privacy: FedSel accounts for support selection within an LDP mechanism (Liu et al., 2020), while sparsification, quantization, and perturbation require explicit threat-model evaluation before privacy claims (Yue et al., 2022).

Positioning. *SPOS* measures benign client and shared-reference checkpoint relatedness through layer-normalized support overlap. Its index-only affinity message omits update values, signs, logits, and activations and is compatible with secure aggregation: dense updates from predeclared cohorts remain protected, while the separate support message guides later cohorts or neighbors. We assess the utility, cost, and leakage of this explicit disclosure without claiming that coordinate IDs are private.

2.3 System Model, Threat Model, and Scope

Aggregation and similarity planes. We distinguish an *aggregation plane*, carrying updates consumed by FL aggregation, from a *similarity plane*, carrying information used to estimate client relatedness. In affinity-only T2 (neighbor selection) and T3 (distribution alignment), clients update from the same reference model, so parameter coordinates and update reference are shared; here, *SPOS* replaces an M-value affinity message with K coordinate IDs. T1 (CPFL) differs: once cluster-specific aggregation begins, clients in different clusters may train from different model values, though architecture, parameter ordering, and training ancestry remain shared. We therefore treat *SPOS* in T1 as an empirical application of coordinate-support relatedness. The shared-reference interpretation and theoretical linkage apply directly to T2, T3, and the checkpoint experiment, but do not establish invariance to cluster-specific reference drift in T1. When T1 also transmits dense updates for aggregation, *SPOS* adds a K-index similarity side channel; in affinity-only T2/T3, it replaces the dense per-client relatedness vector.

Threat model. On the aggregation plane we inherit the adversary model and guarantees of the

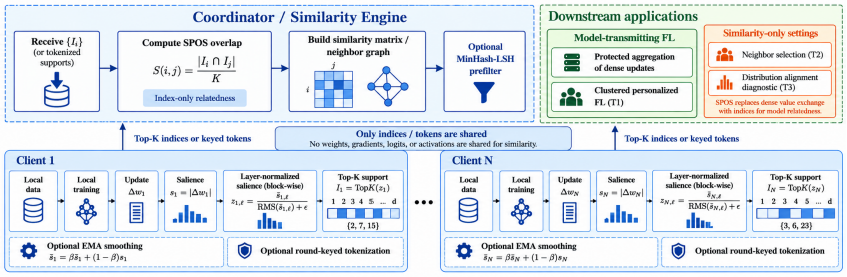


Figure 1: ***SPOS* workflow.** Clients compute layer-normalized salience, extract Top-K parameter indices from local updates, and share only indices. The coordinator computes support overlap for similarity-driven applications.

underlying protocol—secure aggregation (Bonawitz et al., 2017), HE-based aggregation (Gentry, 2009; Paillier, 1999; Jin et al., 2023), or client-side DP (Abadi et al., 2016). We study consistency checking for an authenticated client that may report coordinates inconsistent with a committed input. In aggregation plane, when the commitment is linked to the update accepted by the aggregation protocol, the sampled audit constrains report inconsistency as described in Appx. F. In similarity plane, the proof establishes consistency only with a client-chosen committed input and does not prevent a malicious client from choosing an update that induces a desired support; adversarial input manipulation in these regimes is outside our guarantee.

Scope. The protocols provide aggregation protection; *SPOS* adds none. Its index-only signal complements them by estimating relatedness without disclosing update values, magnitudes, or signs. Coordinator–client collusion, Byzantine training, and attacks on clustering output are outside scope; collusion can weaken both the underlying protection and the tokenization round key, beyond what a similarity method can repair. Raw coordinate IDs may reveal coarse salience patterns and enable cross-round linkage; Appx. I quantifies this and evaluates round-keyed tokenization as a mitigation.

3 METHOD

SPOS is an *index-only* similarity primitive that reveals no update magnitudes or signs. Each client derives update salience, applies block-wise RMS normalization, selects a global TopK support, and communicates only the selected parameter indices. For large client populations, MinHash–LSH can optionally prefilter candidates before exact *SPOS* evaluation. Cross-client similarity is computed solely from support overlap, without sharing real-valued weights, gradients, logits, or activations.

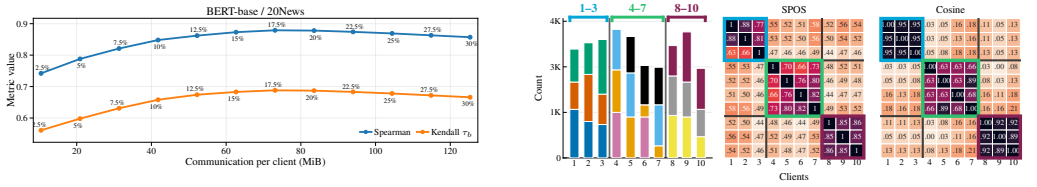
3.1 *SPOS*: SALIENT PARAMETER OVERLAP SIMILARITY

Figure 1 summarizes the *SPOS* protocol. Each client converts its local update into a sparse index support by selecting the most salient parameter coordinates after layer-wise normalization. The coordinator estimates model relatedness from support overlap using only indices or keyed tokens, thereby separating similarity estimation from dense value exchange. The protocol steps are as follows.

Local salience from update magnitude. At round t, client i starts from the broadcast model $w^{(t)}$ and performs local training to obtain $w_i^{(t+1)}$ and the corresponding update $\Delta w_i^{(t)} := w_i^{(t+1)} - w^{(t)}$. We derive a nonnegative salience vector

$$s_i^{(t)} \in \mathbb{R}^M_{\geq 0}, \qquad s_{i,m}^{(t)} := \left|\Delta w_{i,m}^{(t)}\right|.$$

This is a first-order proxy for *where learning concentrates*, requires no second-order estimation, and is available once the local update is formed. To reduce round-to-round noise, we optionally smooth salience via $\bar{s}_i^{(t)} \leftarrow \beta \bar{s}_i^{(t-1)} + (1-\beta) s_i^{(t)}$, $\beta \in [0,1)$, and use $\bar{s}_i^{(t)}$ in place of $s_i^{(t)}$ when smoothing is enabled. This EMA is updated *once per FL round* after the client forms $\Delta w_i^{(t)}$; we do *not* maintain a per-mini-batch EMA of gradients during training. In T1, the EMA state is maintained per client across communication rounds. Because cluster-specific reference models may change after reclustering, we do not claim that this EMA is invariant to reference drift; the T1 results are therefore empirical rather than a consequence of the shared-reference theory.



(a) Communication–quality tradeoff on 20News.

(b) Client similarity induced by the data distribution, *SPOS*, and cosine similarity.

Figure 2: Selecting the *SPOS* support budget. (a) Spearman and Kendall correlation as a function of communication cost while varying the support ratio K/M, using 32-bit coordinate IDs. (b) Client similarity structure induced by the data distribution, *SPOS*, and cosine similarity.

Layer-normalized salience. Raw salience can differ systematically across parameter blocks due to layer-dependent gradient and importance scales (Xiong et al., 2020; Persand et al., 2021), so a global Top-K may reflect block scale rather than client-specific learning. We therefore normalize each block before global selection. For block $\ell \in \mathcal{L}$,

$$z_{i,\ell}^{(t)} = \frac{\bar{s}_{i,\ell}^{(t)}}{\mathrm{RMS}(\bar{s}_{i,\ell}^{(t)}) + \epsilon}, \qquad \mathrm{RMS}(x) = \sqrt{\frac{1}{|x|}\sum_{m} x_m^2}.$$

Concatenating the normalized blocks yields $z_i^{(t)} \in \mathbb{R}_{\geq 0}^{M}$. It preserves within-block salience ordering and reduces block-scale bias, allowing global Top-K to compare coordinates on a common scale.

Support extraction. Given a budget $K/M = 0.1$, client i is represented by the *support* of its K most salient layer-normalized coordinates $I_i^{(t)} := \mathrm{TopK}(z_i^{(t)}) \subseteq [M], |I_i^{(t)}| = K$, with deterministic tie-breaking by index. All clients share the same architecture and parameter ordering, so indices in $[M]$ refer to a common coordinate space. Equivalently, we can view $I_i^{(t)}$ as a sparse binary mask $m_i^{(t)} \in \{0,1\}^M$, where $m_{i,m}^{(t)} = \mathbf{1}\{m \in I_i^{(t)}\}$. Crucially, *only the indices* $I_i^{(t)}$ are communicated; no real-valued weights, gradients, logits, or activations are revealed.

Salient parameter overlap similarity. *SPOS* defines client relatedness purely from set overlap. Given $|I_i^{(t)}| = |I_j^{(t)}| = K$, we define

$$S(i,j) := \frac{|I_i^{(t)} \cap I_j^{(t)}|}{K} \in [0,1], \qquad d(i,j) := 1 - S(i,j).$$

Higher overlap in layer-normalized salient coordinates indicates that clients concentrate their updates on similar parts of the shared model, providing an index-only proxy for model relatedness.

3.2 Choosing K

The support size K sets the communication–relatedness tradeoff. We use $K/M = 10\%$ as a single fixed operating point across all main experiments rather than tuning K separately by architecture or split. Appx. H.2 reports a retrospective sensitivity analysis over $K/M \in [2.5\%, 30\%]$ using the same evaluation oracles as the relatedness experiments. These oracle-based sweeps are diagnostic and are not proposed as a deployable online procedure for choosing K. The 10% operating point retains most of the observed ranking quality while avoiding architecture-specific tuning. The other models show similar diminishing returns; Figure 2b provides a complementary structural view. The *SPOS* adjacency matrix reproduces the client-group structure visible in the data-distribution and cosine-similarity matrices: clients with similar local distributions exhibit stronger support overlap, while dissimilar clients remain weakly connected.

3.3 Scalability, Security, and Deployment Considerations

Scalable retrieval. Exact all-pairs *SPOS* costs $O(N^2K)$. When only candidate neighbors are needed, MinHash–LSH acts as a candidate-pruning prefilter and exact *SPOS* is computed on the

retrieved candidates, so retrieval cost changes but the similarity definition does not; a MinHash-only approximation suffices for coarse filtering. The MinHash signature is therefore not used as a replacement for exact *SPOS* in the reported retrieval pipeline: it identifies candidate clients, after which exact support overlap requires access to their support sets. Signature-only approximate Jaccard/*SPOS* retrieval is possible but is not evaluated as the final affinity here. Appx. B gives the construction, and Appx. G.9 reports Recall@4 above 0.90 against exact *SPOS* up to $N = 400$.

Tokenization and report verification. To limit semantic leakage and cross-round linkability from persistent coordinate IDs, we apply a round-specific secret permutation, $\widetilde{I}_i^{(t)} = \{\pi_{\mathsf{rk}_t}(k) : k \in I_i^{(t)}\}$, with rk_t shared among participating clients but hidden from the coordinator. Because the mapping is bijective and common within each round, it preserves *SPOS* overlaps, rankings, and clusters exactly, while preventing direct coordinate interpretation and equality matching across independently keyed rounds unless the key is disclosed. Appx. C details the construction and compromise assumptions. Appx. F further specifies a zero-knowledge consistency check that binds tokens to committed inputs and verifies support construction without revealing raw coordinates or the round key. For T1, the proof is also linked to the aggregation-accepted update; for T2/T3, it certifies consistency with a client-committed input. Sampled audits detect Top-K deviations with the stated soundness bound.

Communication implication. With $K/M = 10\%$ and 32-bit coordinate IDs, *SPOS* sends $32K = 0.1(32M)$ bits per client per round, one tenth the similarity-plane payload of an uncompressed M-entry FP32 relatedness vector: approximately 0.17, 4.47, 43.80, and 36.15 MB for CNN/FMNIST, ResNet18/CIFAR-10, BERT-base/20News, and Wav2Vec2/SpeechCommands. These are payload, not end-to-end protected-aggregation or latency, comparisons; for larger models, the side channel remains tens of MB per client. In CPFL, *SPOS* adds this index-only channel without altering the aggregation plane; in affinity-only settings, it replaces the dense relatedness vector, yielding the 32-bit-ID versus uncompressed-FP32 reduction. Local construction is $O(M)$ for salience, EMA, and block normalization, followed by Top-K extraction. Appx. D.9 evaluates exact index encodings, while Appx. G.6 reports bandwidth-dependent communication and runtime.

4 Theory and Interpretation

Null overlap is nonzero. Let $z_i \in \mathbb{R}^M_{\geq 0}$ be client i's layer-normalized EMA salience, $I_i = \mathrm{TopK}(z_i)$, and $S(i,j) = |I_i \cap I_j|/K$. Even independent supports overlap because K coordinates must be selected. If their common coordinate-inclusion probabilities are p_u, then $E[S(i,j)] = K^{-1}\sum_{u=1}^{M} p_u^2 \geq K/M$, with equality under uniform fixed-size sampling. A chance-corrected score calibrates this null to zero but is an affine transform of S, so it cannot improve rankings; a centered-mask variant provides a popularity-sensitive diagnostic using the same communicated support.

Graded stability gives a ranking guarantee. For a score vector a, let $I_a = \mathrm{TopK}(a)$ and $B_a(r) := \big|\{u \in I_a : a_u - a_{(K+1)} \leq r\}\big|$, where $a_{(K+1)}$ is the largest score outside I_a. If $\|a - b\|_\infty \leq \delta$, then

$$\frac{|I_a \triangle I_b|}{2K} \leq \frac{B_a(2\delta)}{K}. \tag{1}$$

Thus the guarantee degrades with the fraction of genuinely ambiguous boundary coordinates rather than failing whenever the minimum margin is small. Applied to client-score estimates, Eq. (1) bounds pairwise *SPOS* error. If a target's population similarity gap between its rth and $(r+1)$th neighbors exceeds twice this error, its Top-r neighbor set is recovered (Appx. D.4). This guarantees recovery of neighbors defined by population *support overlap* under its score-gap and estimation conditions; it does not guarantee agreement with label-JS, predictive agreement, or transfer oracles. A conditional linkage corollary appears in Appx. D.4.

Distribution mechanism and limits. Let $\mu(P)$ denote population normalized salience and let S^* be overlap between its Top-K supports. Under the local continuity condition $\|\mu(P_i) - \mu(P_j)\|_\infty \leq LD(P_i, P_j)$, we obtain $S^*(i,j) \geq 1 - \frac{B_{\mu(P_i)}\left(2LD(P_i,P_j)\right)}{K}$, so closer distributions admit more overlapping population supports. The class-mixture construction in Appx. D.5 is one idealized sufficient example, not a finite-sample model of multi-epoch SGD. There is no distributional converse because different distributions can induce the same support. There is, however, a score-level partial converse: when $B_a(2\delta)$ is small, overlap below the right side of Eq. (1) certifies $\|a - b\|_\infty > \delta$.

Empirical theory diagnostics. Empirical head concentration, temporal persistence, popularity, and the looseness of the sufficient bounds are calibrated in Appx. E.

Table 1: Relatedness on Wav2Vec2/SpeechCommands under Dir(0.3). "Real-valued input" indicates whether affinity computation uses real-valued per-client information; distribution-based rows are evaluation-only. Pairwise JS defines the ranking oracle; the oracle ranking need not minimize JS@k.

Method	Real-valued input?	R@6↑	R@12↑	JS@6↓	JS@12↓	Spearman↑	Kendall↑
SPOS	**No**	0.770	0.818	0.186	0.198	0.882	0.673
Cosine update	Yes	0.733	0.800	0.186	0.200	0.839	0.650
Euclidean update	Yes	0.650	0.764	0.202	0.210	0.732	0.537
Cosine full salience	Yes	0.772	0.822	0.189	0.202	0.848	0.665
Euclidean full salience	Yes	0.628	0.744	0.209	0.220	0.686	0.503
Cosine Top-K values	Yes	0.706	0.806	0.187	0.199	0.853	0.668
Euclidean Top-K values	Yes	0.639	0.781	0.203	0.209	0.747	0.553
FedFomo utility (Zhang et al., 2021)	Yes	0.711	0.719	0.190	0.210	0.702	0.557
pFedSim-head cosine (Tan et al., 2023)	Yes	0.678	0.769	0.199	0.214	0.756	0.563
CKA-style representation (Kornblith et al., 2019)	Yes	0.461	0.617	0.235	0.239	0.475	0.328
Distribution cosine	Yes	0.794	0.836	0.185	0.200	0.909	0.738
Distribution Hellinger	Yes	0.906	0.928	0.179	0.199	0.986	0.901
Oracle distribution (JS)	Yes	1.000	1.000	0.179	0.200	1.000	1.000

Table 2: Relatedness across architectures and modalities under Dir(0.3). Each entry reports R@6 ↑ / JS@6 ↓ / Kendall ↑. Distribution cosine uses private label histograms and is evaluation-only.

Model / Dataset	Cosine update	FedFomo	Distribution cosine	***SPOS***
CNN / FMNIST	.650 / .124 / .598	.522 / .134 / .362	.800 / .105 / .791	.656 / .124 / .582
ResNet18 / CIFAR-10	.700 / .105 / .586	.606 / .129 / .437	.794 / .115 / .792	.756 / .109 / .588
BERT-base / 20News	.756 / .156 / .656	.706 / .168 / .509	.822 / .152 / .747	.777 / .152 / .658
Wav2Vec2 / SpeechCommands	.733 / .186 / .650	.711 / .190 / .557	.794 / .185 / .738	.770 / .186 / .673

5 EXPERIMENTS

Setup. We evaluate CNN/FMNIST, ResNet18/CIFAR-10, BERT-base/20News, and Wav2Vec2/SpeechCommands with $N = 30$ clients under Dir(0.1), Dir(0.3), Patho(20%), and Patho(30%) non-IID partitions. Unless noted, similarity experiments run for 10 FL rounds with EMA $\beta = 0.6$; CPFL runs for 60 rounds. *SPOS* applies per-block RMS normalization followed by a global Top-K, with $K/M = 10\%$ for all models. Full hyperparameters are in Appx. G. All point estimates, including relatedness, average five seeds. Primary comparisons use within-seed paired differences after averaging clients or client pairs per seed; client-round observations are not independent replicates. For CPFL, Appx. G.4 additionally reports paired seed-level differences and descriptive 95% confidence intervals.

Metrics. R@k measures recall of the JS-oracle top-k neighbors, JS@k the divergence between a client and the mixture of its selected neighbors, and Spearman/Kendall rank agreement with the pairwise JS oracle. Higher R@k/correlation and lower JS@k are better. Oracle-JS is exact for pairwise ranking but need not minimize the mixture-level JS@k objective. For $N = 30$, uniformly random neighbor selection has expected R@6 $= 6/(N-1) = 0.207$ and R@12 $= 12/(N-1) = 0.414$. Because R@k is normalized by k, R@12 need not exceed R@6.

5.1 RELATEDNESS WITHOUT UPDATE VALUES

We first isolate relatedness quality from downstream FL. On Wav2Vec2/SpeechCommands (Table 1), index-only *SPOS* achieves R@6/R@12 of $0.770/0.818$ and Kendall 0.673, versus $0.733/0.800$ and 0.650 for signed full-update cosine. Value-bearing cosine on full salience is slightly higher on R@6/R@12 ($0.772/0.822$). Thus, *SPOS* does not uniformly dominate dense representations; rather, its index-only support preserves similar neighborhood information despite removing magnitudes and signs from the communicated affinity input. Distribution-based methods remain stronger because they directly access the private label histograms defining the oracle. Across all four model/dataset pairs (Table 2), *SPOS* remains competitive with full-update cosine and consistently outperforms FedFomo without transmitting update values.

Beyond-label and checkpoint relatedness. Table 3 shows that *SPOS* retains strong top-neighbor retrieval beyond label skew and extends to shared-reference checkpoints, though dense cosine remains

Table 3: Relatedness beyond label skew and for shared-reference checkpoints. Means over five seeds. The first three scenarios have exactly equal label marginals and use R@5; the checkpoint scenarios use R@3. The value-bearing comparator is signed dense-update cosine.

Scenario	k	*SPOS* R@k	Dense R@k	*SPOS* Spearman	Dense Spearman
Feature shift	5	.757	.741	.556	.523
Concept shift	5	.973	.988	.781	.791
Quantity/compute imbalance	5	.597	.564	.617	.628
Checkpoint: predictive agreement	3	.716	.733	.539	.740
Checkpoint: cross-domain transfer	3	.702	.720	.508	.670

Table 4: Communication-controlled comparison on Wav2Vec2/SpeechCommands under Dir(0.3). Payload is normalized by the *SPOS* budget.

Method	Payload/*SPOS*	R@6↑	R@12↑	JS@6↓	JS@12↓	Spearman↑	Kendall↑
SPOS	1.000	0.770	0.818	0.186	0.198	0.882	0.673
Signed support (same K)	2.000	0.683	0.775	0.192	0.203	0.786	0.592
Signed support (same budget)	1.000	0.683	0.772	0.192	0.203	0.786	0.591
Signed support (split budget)	1.000	0.683	0.772	0.192	0.203	0.786	0.591
Sparse cosine, 4-bit	1.000	0.711	0.797	0.186	0.200	0.845	0.658
Sparse cosine, 8-bit	1.000	0.711	0.808	0.187	0.198	0.856	0.672

stronger for full checkpoint ranking. Three appendix analyses dissect this signal. **(i) Beyond label skew:** Appx. H.4 shows best or tied-best retrieval under matched-label feature shift (R@5 $= .757$) and quantity/compute imbalance ($.597$), and near-parity with signed dense cosine under concept shift ($.781$ versus $.791$ Spearman). **(ii) Support budget:** Appx. H.2 shows that $K/M = 10\%$ retains at least 96.5% of peak Spearman across support-size sweeps. **(iii) Representation and normalization:** Appx. H.1 isolates sign removal, block normalization, and Top-K binarization. Full checkpoint results appear in Appx. H.3.

5.2 Communication-Controlled Compressed Baselines

Table 4 compares representation quality at matched communication on Wav2Vec2/SpeechCommands. *SPOS* uses $K/M = 10\%$ and transmits $32K$ bits as 32-bit coordinate IDs. Baselines include signed support and quantized sparse cosine, with same-budget variants constrained to the *SPOS* payload. At the matched payload, *SPOS* gives the highest R@6, R@12, and Spearman values among the representations shown. The comparison is much closer on the remaining metrics: 8-bit sparse cosine reaches Kendall 0.672 versus 0.673 for *SPOS*, and both obtain JS@12 0.198. We therefore interpret the experiment as showing that support-only relatedness is competitive at the same transmitted bit budget, not as evidence of uniform superiority over quantized sparse values. Thus, layer-normalized support overlap preserves strong relatedness without transmitting parameter values.

5.3 Evaluating *SPOS* in FL Applications

We evaluate *SPOS* on three relatedness-driven tasks, keeping model, splits, training budget, and seeds fixed and varying only the affinity among cosine, Euclidean, Oracle-JS, and index-only *SPOS*.

T1: Clustered personalized FL. Clients are grouped using a fixed-G periodic reclustering wrapper inspired by clustered personalization. We use $G = 3$ clusters and recompute assignments every five communication rounds. Table 5 reports test accuracy. Across the 16 model–split settings, *SPOS* exceeds both dense cosine and Euclidean affinities in 9 settings. Where it trails the stronger dense baseline, the largest gap is 0.93 percentage points. Full results, adding local-only training, unpersonalized FedAvg, and non-normalized Global *SPOS* across all settings, are in Appx. G.4.

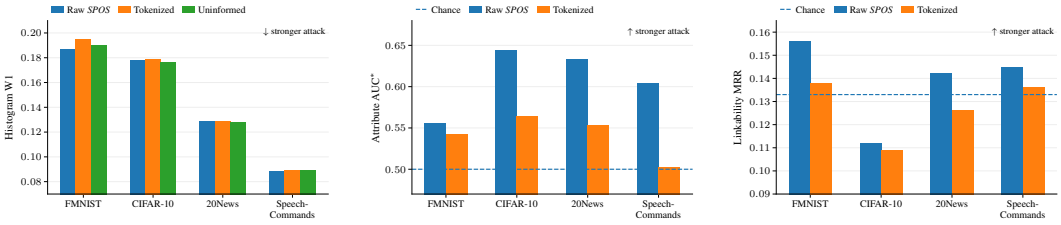
T2: Neighbor selection. We isolate the peer-selection step of decentralized FL. The aggreagator computes its candidate neighbors using pairwise *SPOS* affinities (Kharrat et al., 2025; Fallah et al., 2020). For evaluation only, the oracle neighbor sets are computed centrally from the corresponding ground-truth relatedness measure. Table 6 reports R@6/R@12/Spearman against JS-oracle neighbors. Results vary by architecture and heterogeneity; on Wav2Vec2 Dir(0.3), *SPOS* improves over cosine on all three ($0.770/0.818/0.882$ vs. $0.733/0.800/0.839$). The advantage is not uniform: on ResNet18 Dir(0.1), *SPOS* gives R@6/R@12/Spearman of $.56/.68/.63$, compared with $.70/.81/.85$ for cosine.

Table 5: T1: CPFL test accuracy (%) across non-IID splits.

	Dir(0.3)				**Dir**(0.1)				**Patho**(20%)				**Patho**(30%)			
Model / Dataset	Cos.	Eucl.	Oracle-JS	***SPOS***	Cos.	Eucl.	Oracle-JS	***SPOS***	Cos.	Eucl.	Oracle-JS	***SPOS***	Cos.	Eucl.	Oracle-JS	***SPOS***
CNN / FMNIST	85.42	86.72	86.98	86.82	84.27	84.87	85.48	85.13	70.97	60.97	73.15	72.18	79.55	77.33	79.85	78.62
ResNet18 / CIFAR-10	65.59	64.09	65.97	65.71	62.40	58.28	64.01	61.54	33.85	33.89	35.85	33.36	34.95	37.25	37.29	36.63
BERT-base / 20News	70.96	71.12	71.47	71.26	73.68	71.84	74.72	73.32	64.21	63.92	67.33	64.58	70.88	71.32	76.93	72.09
Wav2Vec2 / SpeechCommands	97.04	96.63	97.15	96.75	96.50	96.30	96.98	96.63	90.92	89.80	92.70	90.78	94.13	94.51	96.90	94.68

Table 6: T2: Neighbor selection across non-IID splits. Each entry reports R@6 / R@12 / Spearman, rounded to two decimal places to save space. All Oracle-JS values are 1 and are therefore omitted.

	Dir(0.3)			**Dir**(0.1)			**Patho**(20%)			**Patho**(30%)		
Model / Dataset	Cos.	Eucl.	***SPOS***	Cos.	Eucl.	***SPOS***	Cos.	Eucl.	***SPOS***	Cos.	Eucl.	***SPOS***
CNN / FMNIST	.65/.77/.78	.64/.71/.66	.66/.68/.79	.65/.73/.75	.61/.65/.54	.64/.72/.74	.58/.59/.42	.57/.51/.34	.59/.59/.44	.54/.59/.35	.53/.59/.44	.53/.57/.42
ResNet18 / CIFAR-10	.70/.76/.78	.50/.65/.41	.76/.81/.78	.70/.81/.85	.67/.69/.12	.56/.68/.63	.79/.65/.64	.65/.61/.52	.77/.63/.64	.77/.82/.74	.63/.71/.69	.76/.80/.73
BERT-base / 20News	.76/.81/.85	.76/.81/.83	.78/.82/.85	.78/.87/.85	.74/.83/.81	.78/.88/.86	.57/.55/.36	.56/.52/.35	.56/.55/.37	.49/.57/.35	.50/.52/.32	.50/.58/.36
Wav2Vec2 / SpeechCommands	.73/.80/.84	.65/.76/.73	.77/.82/.88	.80/.86/.91	.68/.77/.71	.88/.87/.91	.60/.60/.45	.57/.55/.36	.60/.60/.44	.54/.61/.44	.52/.56/.36	.54/.60/.43



(a) Label-histogram inference. (b) Attribute inference. (c) Message-only re-identification.

Figure 3: **Leakage under *SPOS* support exposure.** (a) Label-histogram reconstruction; lower W1 indicates a stronger attack. (b) Binary attribute inference; AUC* $= 0.5$ is chance. (c) Identity-stripped message re-identification; random ranking among 30 clients gives MRR $= 0.133$.

T3 is an additional ranking diagnostic that measures the distributional compatibility of selected neighbors using private label histograms; results are reported in Appx. G.5.

5.4 Leakage under Support Exposure

We evaluate three disclosure channels: label-histogram reconstruction (W1; lower is stronger), binary attribute inference (AUC*; 0.5 is chance), and identity-stripped cross-round re-identification (MRR; higher is stronger). Figure 3 shows that raw *SPOS* remains near the uninformed baseline for histogram inference (W1 differs by at most 0.003), but coordinate semantics leak more attribute information (AUC* $= 0.556$–0.644, reduced to 0.502–0.564 after tokenization). Raw IDs are also strongly cross-round linkable (MRR $= 0.800$–0.940); independently keyed tokenization reduces direct-overlap re-identification to MRR $= 0.131$–0.139, near the random-ranking reference 0.133, while preserving all within-round *SPOS* similarities. This does not guarantee anonymity or prevent frequency- or graph-based linkage. In an FMNIST inversion stress test, dense gradients yield MSE 7.27×10^{-4}, PSNR 31.38, and SSIM 0.798, versus 0.0457, 13.40, and 0.342 for raw *SPOS*. Tokenization hides direct coordinate semantics only while the round key remains secret. Thus, *SPOS* reduces value exposure but is not a privacy mechanism; full attacks and controls appear in Appx. I and Appx. I.1.

6 Conclusion

SPOS resolves a practical bottleneck in similarity estimation under secure aggregation: an explicit index-only support channel yields client affinities while dense updates remain protected. Across label skew and equal-label-marginal controls, it identifies useful neighborhoods under feature and concept shifts and unequal client data and computation. With shared parameter coordinates, sending K coordinate IDs instead of M floating-point affinity values reduces communication time and supports future cohort formation, neighbor selection, and checkpoint retrieval.

Reproducibility Statement

We provide the complete source code required to reproduce the experiments in the supplementary material. The released code includes the *SPOS* implementation, all evaluated baselines, data-partitioning procedures, model configurations, training scripts, hyperparameters, evaluation metrics, and scripts used to generate the reported tables and figures. Random seeds and experiment configurations are specified explicitly so that the main similarity, CPFL, communication, scalability, and ablation results can be reproduced from the provided materials. No proprietary datasets or private infrastructure are required beyond the publicly available datasets and pretrained model checkpoints described in the paper.

Ethics Statement

This work studies client relatedness in federated learning using an index-only representation of salient model-update coordinates. *SPOS* does not transmit update magnitudes or signs, but the selected coordinate support is still a client-specific signal and may reveal information about local training behavior. When deployed alongside secure aggregation, the support message should be treated as a separate disclosure channel and protected according to the application threat model. Our experiments use standard benchmark datasets and do not involve newly collected personal or sensitive data. The complete source code and experimental configuration are provided in the supplementary material to support reproducibility and independent evaluation of these tradeoffs.

AI Use Statement

We used generative AI tools for manuscript editing and proofreading. The authors checked the cited literature, calculations, experimental outputs, and final text and take responsibility for the submission.

References

Martín 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)*, pp. 308–318. ACM, 2016. doi: 10.1145/2976749.2978318.

Alexandr Andoni and Piotr Indyk. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. *Communications of the ACM*, 51(1):117–122, 2008.

James Bell, Adrià Gascón, Tancrède Lepoint, Baiyu Li, Sarah Meiklejohn, Mariana Raykova, and Cathie Yun. {ACORN}: input validation for secure aggregation. In *32nd USENIX Security Symposium (USENIX Security 23)*, pp. 4805–4822, 2023.

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)*, pp. 1175–1191. ACM, 2017. doi: 10.1145/3133956.3133982.

Canh T. Dinh, Nguyen H. Tran, and Tuan D. Nguyen. Personalized federated learning with moreau envelopes. In *NeurIPS*, 2020.

Javad Dogani, Devriş İşler, Nikolaos Laoutaris, et al. Index-only backdoor vetting for secure federated peft of large language models. In *International Conference on Machine Learning*, 2026.

Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning: A meta-learning approach. In *NeurIPS*, 2020.

Trevor Gale, Erich Elsen, and Sara Hooker. The state of sparsity in deep neural networks. *arXiv preprint arXiv:1902.09574*, 2019.

João Gama, Indrè Žliobaitė, Albert Bifet, Mykola Pechenizkiy, Indre Žliobaite, et al. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4):44:1–44:37, 2014.

Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients – how easy is it to break privacy in federated learning? In *NeurIPS*, 2020.

Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC)*, pp. 169–178. ACM, 2009. doi: 10.1145/1536414.1536440.

Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. In *ICML*, 2019.

Avishek Ghosh, Jichan Chung, Dong Yin, and Kannan Ramchandran. An efficient framework for clustered federated learning. In *Advances in neural information processing systems*, 2020.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*, 2022.

Ruoxi Jia, Daphne Dao, Boxin Wang, Frank Hubis, et al. Efficient task-specific data valuation for nearest neighbor algorithms. *PVLDB*, 12(11):1610–1623, 2019.

Weizhao Jin, Yuhang Yao, Shanshan Han, Carlee Joe-Wong, Srivatsan Ravi, Salman Avestimehr, and Chaoyang He. Fedml-he: An efficient homomorphic-encryption-based privacy-preserving federated learning system. *CoRR*, abs/2303.10837, 2023. doi: 10.48550/ARXIV.2303.10837. URL `https://doi.org/10.48550/arXiv.2303.10837`.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank J. Reddi, Sebastian U. Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5132–5143. PMLR, 2020.

Salma Kharrat, Marco Canini, and Samuel Horváth. Dpfl: Decentralized personalized federated learning. In *International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 5086–5094, 2025.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *ICML*, 2017.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning (ICML)*, 2019.

Namhoon Lee, Thalaiyasingam Ajanthan, and Philip H. S. Torr. Snip: Single-shot network pruning based on connection sensitivity. In *ICLR*, 2019.

Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. In *International Conference on Learning Representations (ICLR)*, 2017.

Wenqian Li, Shuran Fu, Fengrui Zhang, and Yan Pang. Data valuation and detections in federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24128–24137, 2024.

Ruixuan Liu, Yang Cao, Masatoshi Yoshikawa, and Hong Chen. Fedsel: Federated sgd under local differential privacy with top-k dimension selection. *arXiv preprint arXiv:2003.10637*, 2020.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Jie Lu, Anjin Liu, Fan Dong, Feng Gu, João Gama, and Guangquan Zhang. Learning under concept drift: A review. *IEEE Transactions on Knowledge and Data Engineering*, 31(12):2346–2363, 2018.

Hidde Lycklama, Lukas Burkhalter, Alexander Viand, Nicolas Küchler, and Anwar Hithnawi. Rofl: Robustness of secure federated learning. In *2023 IEEE Symposium on Security and Privacy (SP)*, pp. 453–476. IEEE, 2023.

Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Exploiting unintended feature leakage in collaborative learning. In *IEEE Symposium on Security and Privacy*, 2019.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, et al. Mixed precision training. In *ICLR*, 2018.

Ari S. Morcos, Haonan Yu, Michela Paganini, and Yuandong Tian. One ticket to win them all: Generalizing lottery ticket initializations across datasets and optimizers. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology – EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pp. 223–238. Springer, 1999. doi: 10.1007/3-540-48910-X_16.

Kaveena Persand, Andrew Anderson, and David Gregg. Taxonomy of saliency metrics for channel pruning. *IEEE Access*, 9:120110–120126, 2021.

Tahseen Rabbani, Brandon Feng, Marco Bornstein, Kyle Rui Sang, Yifan Yang, Arjun Rajkumar, Amitabh Varshney, and Furong Huang. Comfetch: Federated learning of large networks on memory-constrained clients via sketching. *arXiv preprint arXiv:2109.08346*, 2021.

Murtaza Rangwala, Farag Azzedin, Richard O. Sinnott, and Rajkumar Buyya. Sketchguard: Scaling byzantine-robust decentralized federated learning via sketch-based screening. *arXiv preprint arXiv:2510.07922*, 2025.

Daniel Rothchild, Ashwinee Panda, Enayat Ullah, Nikita Ivkin, Ion Stoica, Vladimir Braverman, Joseph Gonzalez, and Raman Arora. Fetchsgd: Communication-efficient federated learning with sketching. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, 2020.

Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE transactions on neural networks and learning systems*, 32(8):3710–3722, 2020.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy*, 2017.

J Tan, Y Zhou, G Liu, JH Wang, and S Yu. pfedsim: Similarity-aware model aggregation towards personalized federated learning. arxiv e-prints, p. *arXiv preprint arXiv:2305.15706*, 2023.

Yue Tan, Guodong Long, Lu Liu, Tianyi Zhou, Qinghua Lu, Jing Jiang, and Chengqi Zhang. Fedproto: Federated prototype learning across heterogeneous clients. In *AAAI*, 2022.

Lucas Theis, Iryna Korshunova, Alykhan Tejani, and Ferenc Huszár. Faster gaze prediction with dense networks and fisher pruning. *arXiv preprint arXiv:1801.05787*, 2018.

Saeed Vahidian, Mahdi Morafah, Weijia Wang, Vyacheslav Kungurtsev, Chen Chen, Mubarak Shah, and Bill Lin. Efficient distribution similarity identification in clustered federated learning via principal angles between client data subspaces. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2023.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019.

Lin Wang, Yongxin Guo, Tao Lin, and Xiaoying Tang. DELTA: Diverse client sampling for fast federated learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. On layer normalization in the transformer architecture. In *ICML*, 2020.

Kai Yue, Richeng Jin, Chau-Wai Wong, Dror Baron, and Huaiyu Dai. Gradient obfuscation gives a false sense of security in federated learning. *arXiv preprint arXiv:2206.04055*, 2022.

Michael Zhang, Karan Sapra, Sanja Fidler, Serena Yeung, and Jose M. Alvarez. Personalized federated learning with first order model optimization. In *International Conference on Learning Representations (ICLR)*, 2021.

Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018.

Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. In *NeurIPS*, 2019.

A MOTIVATION: UTILITY UNDER PERTURBATION AND ENCRYPTED COMPUTATION

Similarity-based FL methods often construct client affinities from individual update vectors, creating tension with mechanisms that make those vectors unavailable or deliberately perturb them. We study two resulting systems effects: the sensitivity of dense-update neighborhoods to clipped Gaussian perturbation and the cost of evaluating dense similarities under homomorphic encryption. These experiments compare utility and systems cost; they do not compare mechanisms with matched formal privacy guarantees.

Gaussian perturbation can erase dense-update relatedness. For this diagnostic, each client update $u_i \in \mathbb{R}^M$ is clipped to ℓ_2 norm C and independently perturbed as

$$\widetilde{u}_i = \mathrm{clip}_C(u_i) + \mathcal{N}(0, \sigma^2 C^2 I).$$

This is a controlled noise-sensitivity diagnostic, not a complete differential-privacy evaluation. We do not specify an adjacency relation, sampling rate, number of composed rounds, or privacy accountant, and therefore do not assign an (ϵ, δ) guarantee to this experiment. The parameter σ should be interpreted only as the perturbation level used in the diagnostic. We evaluate cosine and negative Euclidean similarity on $\widetilde{u}_i$, and also test extracting a Top-K support *after* the same perturbation. Oracle relatedness is $1 - \mathrm{JS}(\pi_i, \pi_j)$ between client label distributions. We measure oracle-neighbor recall (R@6), the JS divergence between each client and the mixture of its six selected neighbors (JS@6), and Spearman correlation over candidate rankings.

Without perturbation, cosine achieves R@6 of 0.650, 0.700, and 0.756 and Spearman correlations of 0.784, 0.778, and 0.851 on FMNIST, CIFAR-10, and 20News, respectively. Already at $\sigma = 0.1$, cosine and Euclidean R@6 fall to 0.178–0.250 and Spearman to -0.071–0.046. At $\sigma = 1$, Table 7 shows that signals derived from the perturbed update approach the random-neighbor R@6 reference,

$$\frac{6}{30 - 1} = 0.207,$$

with near-zero rank correlation. In contrast, locally constructed *SPOS* retains R@6 of 0.640–0.760 and Spearman of 0.760–0.860. Its JS@6 is also lower than the strongest signal obtained from the perturbed update by 0.259 on FMNIST, 0.207 on CIFAR-10, and 0.244 on 20News.

Applying support construction after perturbation does not recover the lost neighborhood utility: post-noise support remains close to the degraded dense-update signals. If $\widetilde{u}_i$ were produced by a properly calibrated DP mechanism, a support computed solely from $\widetilde{u}_i$ would be post-processing of that output and could inherit its guarantee. Raw local *SPOS* performs better because it is instead computed from the unperturbed update. Releasing that raw support creates an additional unnoised disclosure and is not covered by the guarantee of a separately noised update. *SPOS* constructs its support locally from the unperturbed learning signal and shields the dense update separately; the support channel can then be protected independently as described in Sec. 3.3.

HE preserves dense similarity but at substantial cost. We next evaluate actual CKKS operations in TenSEAL rather than using a simulated cryptographic proxy. With polynomial-modulus degree 8192, coefficient-modulus bits $[60, 40, 40, 60]$, scale 2^{40}, and 4096 packed coordinates, encrypted dot products have maximum absolute error below 2.5×10^{-6} across all evaluated architectures (Table 8). Thus, CKKS preserves the dense similarity computation accurately in this microbenchmark; the primary limitation is systems cost rather than numerical fidelity.

Linearly projecting the measured packed-block cost to a full model update gives 32.5–8449.7 MiB of serialized ciphertext per client and 2.3–603.4 seconds per client pair. The scaling becomes particularly pronounced for large pretrained models. For Wav2Vec2/SpeechCommands ($M = 90.4$M trainable update coordinates), the projection reaches 6972.5 MiB per client and 504.0 seconds for one encrypted pairwise dot product. For BERT-base/20News ($M = 109.5$M), the corresponding costs are 8449.7 MiB and 603.4 seconds per pair. With 30 clients, corresponding to $\binom{30}{2} = 435$ distinct client pairs, the projected pairwise homomorphic dot-product time is 60.90 hours for Wav2Vec2 and 72.9 hours for BERT-base.

The SpeechCommands result shows that this cost regime is not specific to language models: a large pretrained speech architecture exhibits similarly steep dense-HE scaling. Across the evaluated

Table 7: Relatedness under local clipping and Gaussian perturbation at $\sigma = 1$ (30 clients; $k = 6$). Higher R@6 and Spearman are better; lower JS@6 is better. 'Post-noise support" is computed from $\widetilde{u}_i$ and is therefore post-processing of the perturbed vector. 'Raw local *SPOS*" is computed before perturbation and is a non-DP utility reference. The comparison is not privacy-matched, and no (ϵ, δ) guarantee is claimed.

Dataset	Relatedness signal	R@6 ↑	JS@6 ↓	Spearman ↑
FMNIST	Cosine on $\widetilde{u}$	.191	.378	-.019
	Euclidean on $\widetilde{u}$	.206	.386	.055
	Post-noise support	.198	.399	-.076
	Raw local *SPOS* (non-DP)	**.640**	**.119**	**.760**
CIFAR-10	Cosine on $\widetilde{u}$	.183	.342	-.006
	Euclidean on $\widetilde{u}$	.172	.354	-.014
	Post-noise support	.261	.309	-.012
	Raw local *SPOS* (non-DP)	**.680**	**.102**	**.760**
20News	Cosine on $\widetilde{u}$	.206	.411	.032
	Euclidean on $\widetilde{u}$	.204	.408	.013
	Post-noise support	.206	.399	.009
	Raw local *SPOS* (non-DP)	**.760**	**.155**	**.860**

Table 8: CKKS/TenSEAL microbenchmark. M denotes the number of trainable update coordinates considered by *SPOS*. Error and per-block time are measured on 4096-coordinate ciphertext blocks using four clients (ten products including self-products). Payload and full-vector dot time are linear projections from measured blocks; the 30-client pairwise time further projects the per-pair cost over $\binom{30}{2} = 435$ distinct client pairs.

Model / Dataset	M	Dot/block (s)	Max abs. error	MiB/client	Dot/pair (s)	30-client pairs (h)
CNN / FMNIST	421,642	.023	8.74×10^{-7}	32.5	2.3	.28
ResNet18 / CIFAR-10	11,173,962	.023	2.48×10^{-6}	862.8	62.4	7.54
BERT-base / 20News	109,497,620	.023	8.78×10^{-7}	8449.7	603.4	72.9
Wav2Vec2 / SpeechCommands	90,377,123	.023	1.20×10^{-6}	6972.5	504.0	60.90

models, projected 30-client pairwise time increases from 0.28 hours for CNN/FMNIST to 7.54 hours for ResNet18/CIFAR-10, 60.90 hours for Wav2Vec2/SpeechCommands, and 72.9 hours for BERT-base/20News. Hence, although CKKS accurately preserves the evaluated arithmetic, constructing dense encrypted client-to-client affinities becomes increasingly expensive as model dimension grows.

These measurements illustrate a utility–cost–disclosure tension. Gaussian perturbation can severely distort the neighborhood structure required by relatedness-driven FL, whereas CKKS preserves the evaluated arithmetic but incurs costs that grow rapidly with model dimension. The projected quantities are controlled extrapolations from measured ciphertext-block operations and should not be interpreted as end-to-end deployment latency; transport, decryption, and protocol orchestration would introduce additional costs.

Raw *SPOS* occupies a different operating point. Rather than evaluating dense pairwise functions over protected real-valued updates, each client constructs its salient support locally and exposes only the resulting coordinate identities for relatedness estimation. This avoids dense encrypted pairwise comparison while retaining useful neighborhood structure. The comparison is therefore one of systems tradeoffs rather than equivalent privacy guarantees: CKKS protects dense values cryptographically, whereas raw *SPOS* reduces value exposure but releases a client-specific discrete support and does not by itself provide formal privacy.

B MinHash–LSH for Scalable Candidate Retrieval

Why MinHash–LSH. Computing all-pairs overlap between n clients' supports $\{I_i\}_{i=1}^n$ costs $O(n^2K)$ with sorted K-lists (two-pointer intersection), and $O(n^2 \cdot M/w)$ with bitsets (word size w), which is prohibitive at large n. We use MinHash with locality-sensitive hashing (LSH) to

avoid exhaustive all-pairs candidate evaluation. Each client publishes a short signature (or the b band keys), the server (or a peer) looks up candidates that collide in at least one band, and exact similarity is computed only on this small candidate set. When the number of clients is large, we use MinHash/LSH only as a *candidate prefilter*. In each round, clients upload their Top-K index set $I_i^{(t)}$ (or an equivalent compressed representation) *and* a MinHash signature derived from $I_i^{(t)}$. The server uses the signatures to retrieve a small candidate list via LSH, and then computes the ***SPOS*** similarity *exactly* for those candidates using the corresponding index sets, i.e.,

$$S(i,q) = \frac{|I_i^{(t)} \cap I_q^{(t)}|}{K}.$$

Thus MinHash reduces the search cost, while the reported ***SPOS*** similarities remain exact. This preserves ***SPOS***'s index-only property (only indices/signatures, no real-valued parameters) and scales to large n.

For a support set $I \subseteq [M]$, define h independent MinHash values

$$\phi_k(I) = \min\{\ H_k(j) : j \in I\ \},$$

where j denotes a parameter-coordinate index and $H_k : [M] \to \{0, \dots, 2^{32}-1\}$ are 32-bit hash functions that approximate minwise permutations. The signature is

$$\Phi(I) = (\phi_1(I), \dots, \phi_h(I)) \in \mathbb{N}^h.$$

MinHash collisions estimate *Jaccard* similarity. For any two sets I_i, I_q, define

$$J := J(I_i, I_q) = \frac{|I_i \cap I_q|}{|I_i \cup I_q|}.$$

Then

$$\Pr\big[\phi_k(I_i) = \phi_k(I_q)\big] = J,$$

$$\hat{J} = \frac{1}{h}\sum_{k=1}^{h} \mathbf{1}\{\phi_k(I_i) = \phi_k(I_q)\},$$

$$\mathrm{Var}[\hat{J}] = \frac{J(1-J)}{h}.$$

When $|I_i| = |I_q| = K$, ***SPOS*** overlap

$$s = \frac{|I_i \cap I_q|}{K}$$

is a monotone transform of J, so MinHash–LSH can be used as a candidate prefilter for high-***SPOS*** pairs. After LSH returns candidates, the coordinator obtains the corresponding K-index lists (uploaded once per round or fetched on-demand) to compute exact ***SPOS*** on candidates. Signature cost is $O(hK)$ time and $O(h)$ words per client.

B.1 MinHash–LSH facts and end-to-end complexity

Throughout, each client submits a K-subset $I_i \subseteq [M]$ (the Top-K index set), and our ***SPOS*** overlap is

$$\mathrm{SPOS}(i,q) = \frac{|I_i \cap I_q|}{K} \in [0,1].$$

MinHash–LSH is naturally parameterized by *Jaccard* similarity, which is a monotone transform of ***SPOS*** for equal-size sets $|I_i| = |I_q| = K$:

$$J(I_i, I_q) = \frac{|I_i \cap I_q|}{|I_i \cup I_q|} = \frac{K\,\mathrm{SPOS}(i,q)}{2K - K\,\mathrm{SPOS}(i,q)} = \frac{\mathrm{SPOS}(i,q)}{2 - \mathrm{SPOS}(i,q)}.$$

Thus, MinHash banding can be used as a candidate prefilter for high-***SPOS*** pairs.

Lemma B.1 (MinHash collision probability; unbiased Jaccard estimate)**.** *Let $I_1, I_2 \subseteq [M]$ with $|I_1| = |I_2| = K$, and define*

$$J := J(I_1, I_2) = \frac{\mathrm{SPOS}(1,2)}{2 - \mathrm{SPOS}(1,2)}.$$

With h independent MinHash functions, the estimator

$$\hat{J} = \frac{1}{h} \sum_{u=1}^{h} \mathbf{1}\{\phi_u(I_1) = \phi_u(I_2)\}$$

satisfies

$$\mathbb{E}[\hat{J}] = J, \qquad \mathrm{Var}[\hat{J}] = \frac{J(1-J)}{h}.$$

Equivalently, a plug-in estimator of SPOS *overlap is*

$$\widehat{\mathrm{SPOS}} = \frac{2\hat{J}}{1+\hat{J}},$$

which is monotone in $\hat{J}$.

Proof. MinHash collision equivalence gives

$$\Pr[\phi_u(I_1) = \phi_u(I_2)] = J(I_1, I_2) = J.$$

Hence each collision indicator is Bernoulli(J). Averaging h independent indicators yields $\mathbb{E}[\hat{J}] = J$ and $\mathrm{Var}[\hat{J}] = J(1-J)/h$. □

Lemma B.2 (Banding retrieval probability in terms of *SPOS*)**.** *Arrange $h = br$ MinHash values into b bands of r rows. For two clients (i, q) with* SPOS *overlap*

$$s_{iq} := \mathrm{SPOS}(i, q)$$

and corresponding Jaccard similarity

$$J_{iq} := J(I_i, I_q) = \frac{s_{iq}}{2 - s_{iq}},$$

the probability that they collide in a particular band is J_{iq}^r, and the probability that they are retrieved as candidates is

$$p_{iq} = 1 - \left(1 - J_{iq}^r\right)^b = 1 - \left(1 - \left(\frac{s_{iq}}{2 - s_{iq}}\right)^r\right)^b.$$

Proof. Standard MinHash–LSH banding analysis with independent bands. □

Theorem B.3 (Candidate set size and total per-round cost)**.** *Let $n = |S_t|$ clients participate in a round, each with a K-set I_i. Using $h = br$ MinHashes and b hash tables, the expected candidate set size for query client i (excluding i) is*

$$\mathbb{E}[C_i] = \sum_{q \neq i} p_{iq} = \sum_{q \neq i} \left[1 - \left(1 - J_{iq}^r\right)^b\right],$$

$$J_{iq} = \frac{\mathrm{SPOS}(i,q)}{2 - \mathrm{SPOS}(i,q)}.$$

The total server cost per round is

$$O(nh) + O\left(\sum_{i=1}^{n} \mathbb{E}[C_i]\right) + O\left(\sum_{(i,q) \in \mathrm{cands}} K\right),$$

i.e., signature handling, candidate lookups, and final exact SPOS *overlaps. Equivalently, since each exact overlap costs $O(K)$, the final term is*

$$O\left(\sum_{i=1}^{n} \mathbb{E}[C_i]\, K\right).$$

Signature memory is $O(nh)$.

Algorithm 1 *SPOS*–MinHash–LSH (Build and Query)

Require: Supports $\{I_i\}_{i=1}^{n}$, hash functions $\{H_k\}_{k=1}^{h}$, number of bands b, rows per band r
Ensure: Candidate set or top-k neighbors for a query

Build phase:
for $i = 1$ to n **do**
  Compute MinHash signature $\Phi(I_i)$:
  $\phi_k(I_i) = \min_{j \in I_i} H_k(j), \quad k = 1, \ldots, h$
  for $u = 1$ to b **do**
    Form band key
    $B_u(I_i) = \big(\phi_{(u-1)r+1}(I_i), \ldots, \phi_{ur}(I_i)\big)$
    Insert i into hash table $\mathcal{T}_u[B_u(I_i)]$
  end for
end for

Query phase:
Require: Query support I_q
Compute $\Phi(I_q)$ and band keys $\{B_u(I_q)\}_{u=1}^{b}$
$C \leftarrow \bigcup_{u=1}^{b} \mathcal{T}_u[B_u(I_q)] \setminus \{q\}$
for each candidate client $c \in C$ **do**
  Retrieve I_c if not locally available
  Compute similarity score
  $S(q, c) = \frac{|I_q \cap I_c|}{K}$
end for
OUTPUT: top-k candidates ranked by $S(q, c)$ (or all c with $S(q, c) \geq s_{\min}$)

Proof. Linearity of expectation yields the candidate-count expression. Exact overlap on two K-sets costs $O(K)$, for example via a two-pointer merge on sorted indices or by hashing one set, which yields the stated total cost bound. □

C ROUND-KEYED COORDINATE PSEUDONYMIZATION: CONSTRUCTION, ASSUMPTIONS, AND LIMITS

Although *SPOS* transmits no floating-point update values, raw coordinate IDs can reveal which parameters are salient and allow direct matching of the same coordinate across rounds. Round-keyed tokenization addresses this specific semantic channel: it replaces coordinate labels by round-scoped pseudonyms while preserving every equality relation required by *SPOS*. It is therefore coordinate-semantic pseudonymization, not client anonymity or general unlinkability. In particular, it preserves the full within-round support-incidence structure and similarity graph.

Let $I_i^{(t)} \subseteq [M]$ be client i's Top-K support at round t. We transmit

$$\widetilde{I}_i^{(t)} = \{\pi_{\mathsf{rk}_t}(k) : k \in I_i^{(t)}\},$$

where $\pi_{\mathsf{rk}_t} : [M] \to [M]$ is a format-preserving keyed permutation. The round key rk_t is available to every participating client and is withheld from the coordinator. We assume $M < 2^{32}$, so both raw indices and tokens can be represented by 32-bit words. Because π_{rk_t} is bijective, token collisions do not occur and token equality is equivalent to equality of the underlying coordinates. A truncated keyed hash is not sufficient for this purpose because it need not be injective.

Correctness: invariance of *SPOS*. *SPOS* uses only intersection cardinalities. For fixed K_t, the keyed remapping is injective, so equality is preserved exactly: $\pi_{\mathsf{rk}_t}(k) = \pi_{\mathsf{rk}_t}(k') \iff k = k'$. Therefore, for any clients i, j in the same round t, $|\widetilde{I}_i^{(t)} \cap \widetilde{I}_j^{(t)}| = |I_i^{(t)} \cap I_j^{(t)}|$. Consequently, keyed-token *SPOS* yields exactly the same similarity values:

$$S_{\mathrm{tok}}(i,j) := \frac{|\widetilde{I}_i^{(t)} \cap \widetilde{I}_j^{(t)}|}{K} = \frac{|I_i^{(t)} \cap I_j^{(t)}|}{K} = S(i,j).$$

Thus any downstream algorithm that depends only on *SPOS*, including neighbor graphs, rankings, and clustering, is unchanged. Let $X^{(t)} \in \{0,1\}^{N \times M}$ be the client–coordinate incidence matrix, where $X_{ik}^{(t)} = 1$ exactly when $k \in I_i^{(t)}$, and let P_t be the permutation matrix induced by π_{rk_t}. Tokenization produces

$$\widetilde{X}^{(t)} = X^{(t)} P_t.$$

Consequently,

$$\widetilde{X}^{(t)} \widetilde{X}^{(t)\top} = X^{(t)} P_t P_t^\top X^{(t)\top} = X^{(t)} X^{(t)\top}.$$

Thus tokenization preserves the complete within-round client Gram matrix, not only individual *SPOS* values. Any attack based only on within-round overlap, graph topology, neighborhoods, rankings, or clustering receives exactly the same input before and after tokenization.

Disclosure effect and limitations. Round-keyed tokenization should be interpreted only as coordinate-semantic pseudonymization. Because the mapping is a common within-round permutation, it preserves the complete client–token incidence structure and hence the full within-round *SPOS* matrix. The mechanism also relies on a non-collusion assumption: disclosure of the round key by any participating client restores the raw-coordinate mapping for that round. We do not evaluate frequency-based inference of the underlying coordinate mapping. The purpose of tokenization is removing direct coordinate semantics under the stated key-separation assumption.

C.1 ROUND-KEYED TOKENIZATION: KEY ESTABLISHMENT, COMPROMISE, AND CHURN

Round-keyed tokenization requires a fresh key K_t that is available to every participating client but unavailable to the coordinator. Possible mechanisms include a contributory group-key protocol, derivation from a client-held group secret, or a TEE-backed service that exposes only a non-exportable permutation interface. In the ordinary group-key variants, this is a strong non-collusion assumption:

any participating client that reveals K_t allows the coordinator to invert the token mapping for every client report in round t. The oracle-key experiment in Appx. I models this single-participant key-disclosure condition.

If $K_t = \mathrm{HKDF}(\mathsf{rk}_{\mathrm{grp}}, t)$ is derived directly from a long-lived group secret, compromise of K_{grp} can expose all derivable round keys. Limiting compromise to one round requires a forward-secure group-key ratchet, authenticated membership changes, deletion of expired key material, and a policy for excluding compromised clients. These mechanisms are deployment requirements rather than properties supplied by *SPOS*.

The key assumption affects only coordinate-semantic pseudonymization. *SPOS* utility and within-round similarities remain unchanged if raw IDs are used or if the coordinator learns the key. Tokenization does not hide authenticated sender identity and does not conceal the within-round similarity graph.

Binding tokenization to report verification. A verifier that does not know K_t cannot independently check that the transmitted tokens were generated from the client's raw support. Appendix F therefore binds a domain-separated commitment to K_t and a Merkle root of the exact public token list into the signed proof transcript. The zero-knowledge circuit proves the relation $y_j = \pi_{\mathsf{rk}_t}(j)$ for audited private coordinates and proves membership or non-membership of y_j in the committed token set, without disclosing K_t, j, or the raw support.

Churn and partial participation. Let $\mathcal{C}_t$ be the sampled client set at round t. Only clients in $\mathcal{C}_t$ need K_t; a client not selected (or offline) simply does not contribute tokenized indices in that round. If a selected client fails to obtain K_t (dropout), it is treated as a standard straggler and omitted from *SPOS* computation for that round. Importantly, tokenization is *round-local*: missing K_t does not affect future rounds, and does not require replay or key recovery.

Overhead. Client-side computation is one keyed-remapping evaluation per selected index, giving time $O(K)$ per client per round, which is negligible compared with local training. Communication remains $O(K)$ per client per round: each pseudonym is a 32-bit word, so the payload is $32K$ bits. Thus, tokenized *SPOS* has the same communication cost as transmitting raw 32-bit coordinate IDs, while replacing direct coordinate labels with round-scoped pseudonyms from the coordinator's view.

D More Theoretical Analysis

D.1 Actual score, scope, and signed variant

For client i at round t, the score and support used by *SPOS* are

$$z_i^{(t)} = \mathcal{N}_\epsilon\left(\bar{s}_i^{(t)}\right) \geq 0, \quad \bar{s}_i^{(t)} = \beta \bar{s}_i^{(t-1)} + (1-\beta)|\Delta w_i^{(t)}|, \quad I_i^{(t)} = \mathrm{TopK}\left(z_i^{(t)}\right), \tag{2}$$

where $\mathcal{N}_\epsilon$ is the blockwise RMS normalization from Sec. 3; ties are broken deterministically. *SPOS* measures where learning concentrates and need not approximate signed update cosine.

For completeness, the signed-support baseline appearing in our experiments uses

$$I_i^+ = \mathrm{TopK}((\Delta w_i)_+, K_+), \qquad I_i^- = \mathrm{TopK}((-\Delta w_i)_+, K_-), \tag{3}$$

and, for common K_+, K_- across clients,

$$S_\pm(i,j) = \frac{|I_i^+ \cap I_j^+| + |I_i^- \cap I_j^-| - |I_i^+ \cap I_j^-| - |I_i^- \cap I_j^+|}{K_+ + K_-}. \tag{4}$$

Using $K_+ + K_- = K$ gives a split-budget variant; other rows in our ablations use the budgets stated in their captions. We retain sign-free *SPOS* as the main method because it reveals less information and uses a single support.

D.2 Mask geometry, null overlap, and calibration

Let $m_i \in \{0,1\}^M$ be the indicator of I_i.

Proposition D.1 (Mask geometry). *For K-subsets $I_i, I_j \subseteq [M]$,*

$$S(i,j) = \frac{\langle m_i, m_j \rangle}{\|m_i\|_2 \|m_j\|_2}, \qquad d_S(i,j) := 1 - S(i,j) = \frac{|I_i \triangle I_j|}{2K}.$$

Moreover, d_S is a metric on the family of K-subsets.

Proof. $\langle m_i, m_j \rangle = |I_i \cap I_j|$ and $\|m_i\|_2 = \|m_j\|_2 = \sqrt{K}$. Equal set sizes give $|I_i \triangle I_j| = 2K - 2|I_i \cap I_j|$. Symmetric-difference cardinality is a metric, and positive scaling preserves the metric axioms. □

Proposition D.2 (Null overlap). *Let independent random K-subsets have marginal inclusion probabilities $p_{i,u} = \Pr(u \in I_i)$ and $p_{j,u} = \Pr(u \in I_j)$. Then*

$$\mathbb{E}[S(i,j)] = \frac{1}{K} \sum_{u=1}^{M} p_{i,u} p_{j,u}. \tag{5}$$

If the two clients share marginals p_u, then $\mathbb{E}[S(i,j)] = K^{-1} \sum_u p_u^2 \geq K/M$. Here $\sum_u p_u = K$ because $|I_i| = K$ almost surely and $\sum_u \mathbb{E}[\mathbf{1}\{u \in I_i\}] = \mathbb{E}|I_i|$. Under uniform fixed-size sampling, with $\rho_K = K/M$,

$$\mathbb{E}[S(i,j)] = \rho_K, \qquad \mathrm{Var}[S(i,j)] = \frac{(1-\rho_K)^2}{M-1}. \tag{6}$$

Proof. Independence across clients gives Eq. (5); Cauchy–Schwarz gives $\sum_u p_u^2 \geq K^2/M$. Under uniform sampling, $|I_i \cap I_j|$ is hypergeometric with population size M, K successes, and K draws. Dividing its mean and variance by K and K^2 proves Eq. (6). □

For N observed supports, let $c_u = \sum_{i=1}^{N} \mathbf{1}\{u \in I_i\}$. The coordinate-frequency-preserving permutation null for two distinct clients is

$$\widehat{S}_0 = \frac{1}{K} \sum_{u=1}^{M} \frac{c_u(c_u - 1)}{N(N-1)}. \tag{7}$$

This without-replacement form is preferable to $K^{-1} \sum_u (c_u/N)^2$ when N is small. It also equals the mean *SPOS* over all ordered distinct client pairs, so deviations from it—not its magnitude alone—carry pair-specific information.

Proposition D.3 (What global chance correction can change). *For any fixed $S_0 < 1$, define*

$$S_{\mathrm{cc}}(i,j) = \frac{S(i,j) - S_0}{1 - S_0}. \tag{8}$$

Then S_{cc} has exactly the same pairwise ordering as S, and $1 - S_{\mathrm{cc}} = (1-S)/(1-S_0)$. Consequently, it preserves nearest-neighbor rankings, Kendall rank correlation, and any agglomerative dendrogram invariant to uniform positive rescaling of all distances.

Proof. Equation (8) is strictly increasing in S, and its distance identity follows by rearrangement. □

Thus Eq. (8) is useful for calibration but cannot by itself recover a ranking gap. A popularity-sensitive, no-extra-communication variant is centered-mask cosine:

$$S_{\mathrm{cen}}(i,j) = \frac{\langle m_i - \widehat{p}, m_j - \widehat{p} \rangle}{\|m_i - \widehat{p}\|_2 \|m_j - \widehat{p}\|_2}, \qquad \widehat{p}_u = c_u/N. \tag{9}$$

Under independent clients with known marginals p, the numerator centered by p has expectation zero. In experiments, p should be estimated on a held-out round or by cross-fitting to avoid using a target pair in its own baseline.

D.3 SWEPT QUANTILE SALIENCE BOUND

For nonnegative z_i, write $h_i = z_i \odot m_i$ and $\alpha_i = \|h_i\|_2^2/\|z_i\|_2^2$. For $\tau \in [0,1)$, let $q_i(\tau)$ be the $(\lfloor \tau K \rfloor + 1)$st smallest head value and set

$$\kappa_i(\tau) = \begin{cases} \max\left\{1, \dfrac{\|h_i\|_2}{q_i(\tau)\sqrt{K}}\right\}, & q_i(\tau) > 0, \\ \infty, & q_i(\tau) = 0. \end{cases} \tag{10}$$

Proposition D.4 (Optimized quantile lower bound)**.** *For every pair of nonnegative scores,*

$$\cos(z_i, z_j) \geq \sqrt{\alpha_i \alpha_j} \sup_{\tau \in [0,1)} \frac{[S(i,j) - 2\tau]_+}{\kappa_i(\tau)\kappa_j(\tau)}. \tag{11}$$

The supremum may be computed exactly over the finite set of head-order breakpoints, or approximately on a fixed grid; a ratio with an infinite $\kappa_i(\tau)$ is interpreted as zero.

Proof. For fixed τ, at least $K - \lfloor \tau K \rfloor$ head coordinates of client i are at least $q_i(\tau)$, and similarly for j. Their regular-set intersection with $I_i \cap I_j$ has size at least $K[S(i,j) - 2\tau]_+$. Every coordinate in this intersection contributes at least $\|h_i\|_2\|h_j\|_2/(\kappa_i(\tau)\kappa_j(\tau)K)$. Because $z_i, z_j \geq 0$, all omitted inner-product terms are nonnegative. Dividing by $\|z_i\|_2\|z_j\|_2$ gives the bound for each τ; taking the supremum preserves it. □

This remains a one-sided interpretation bound and does not control ranking. We report its optimized numerical value in Appx. E.

D.4 GRADED TOP-K STABILITY AND NEIGHBOR RANKING

For $a \in \mathbb{R}^M$, let $I_a = \mathrm{TopK}(a)$, let $a_{(K+1)}$ be the largest score outside I_a, and define

$$B_a(r) = \big|\{u \in I_a : a_u - a_{(K+1)} \leq r\}\big|.$$

Theorem D.5 (Graded Top-K perturbation)**.** *If $\|a - b\|_\infty \leq \delta$, then*

$$|I_a \setminus I_b| = \frac{|I_a \triangle I_b|}{2} \leq B_a(2\delta), \qquad \frac{|I_a \cap I_b|}{K} \geq 1 - \frac{B_a(2\delta)}{K}. \tag{12}$$

Proof. If $u \in I_a$ satisfies $a_u - a_{(K+1)} > 2\delta$, then for every $v \notin I_a$, $b_u \geq a_u - \delta > a_v + \delta \geq b_v$. Suppose for contradiction that $u \notin I_b$. Since $|I_a \setminus \{u\}| = K - 1 < |I_b| = K$, some $v \in I_b \setminus I_a$ must exist; Top-K membership would require $b_v \geq b_u$, contradicting the strict inequality. Hence only the $B_a(2\delta)$ ambiguous coordinates can leave. Equal support sizes give $|I_a \setminus I_b| = |I_b \setminus I_a| = |I_a \triangle I_b|/2$. □

Corollary D.6 (Score-level partial converse)**.** *For any $\delta \geq 0$, if*

$$\frac{|I_a \cap I_b|}{K} < 1 - \frac{B_a(2\delta)}{K}, \tag{13}$$

then $\|a - b\|_\infty > \delta$.

Proof. This is the contrapositive of Theorem D.5. □

Corollary D.6 certifies score dissimilarity when the reference boundary is sharp. It is not a distributional converse: a large score difference need not imply a large distance between data distributions.

Let μ_i be client i's population score, $I_i^* = \mathrm{TopK}(\mu_i)$, and let $\widehat{z}_i$ produce $\widehat{I}_i = \mathrm{TopK}(\widehat{z}_i)$. Define

$$e_i = \frac{|\widehat{I}_i \triangle I_i^*|}{2K}, \quad \widehat{S}(i,j) = \frac{|\widehat{I}_i \cap \widehat{I}_j|}{K}, \quad S^*(i,j) = \frac{|I_i^* \cap I_j^*|}{K}.$$

Corollary D.7 (*SPOS* estimation and ranking recovery)**.** *If $\|\widehat{z}_i - \mu_i\|_\infty \le \delta_i$, then*

$$e_i \le \frac{B_{\mu_i}(2\delta_i)}{K}, \qquad |\widehat{S}(i,j) - S^*(i,j)| \le e_i + e_j. \tag{14}$$

For fixed target i, let $\eta_i = \max_{j\neq i}(e_i+e_j)$. If the gap between its rth and $(r+1)$th largest population similarities exceeds $2\eta_i$, its observed Top-r neighbor set is exact.

Proof. The first claim is Theorem D.5. Replacing one K-set in q coordinates changes a normalized intersection by at most q/K; applying this to both clients proves the second claim. Each observed similarity is within η_i of its population value, so a gap greater than $2\eta_i$ cannot be reversed. □

Corollary D.8 (Cluster separation)**.** *Let $\mathcal{C}$ be a population client partition and suppose $|\widehat{S}(i,j) - S^*(i,j)| \le \eta$ for every pair. If*

$$\min_{i\neq j \text{ in the same block of } \mathcal{C}} S^*(i,j) - \max_{i,j \text{ in different blocks of } \mathcal{C}} S^*(i,j) > 2\eta, \tag{15}$$

then every observed within-block similarity exceeds every observed between-block similarity. Consequently, a threshold between the two ranges recovers the partition, and single, complete, or average agglomerative linkage merges within true blocks before making a cross-block merge.

Proof. Subtracting at most η from a within-block score and adding at most η to a between-block score preserves their strict order. The stated linkage rules aggregate pairwise values by a maximum, minimum, or average, each of which remains inside the corresponding separated range. □

D.5 A GRADED DISTRIBUTION–SUPPORT MECHANISM

Corollary D.9 (Local distribution continuity)**.** *Let $\mu(P) \in \mathbb{R}^M_{\ge 0}$ be a population normalized-salience map. If $\|\mu(P_i) - \mu(P_j)\|_\infty \le LD(P_i,P_j)$, then*

$$S^*(i,j) \ge 1 - \frac{B_{\mu(P_i)}(2LD(P_i,P_j))}{K}. \tag{16}$$

Proof. Apply Theorem D.5 with $a = \mu(P_i)$, $b = \mu(P_j)$, and $\delta = LD(P_i,P_j)$. □

As an idealized example, suppose pre-normalization population salience is $r(\pi) = A\pi = \sum_c \pi(c)a_c$. With $G_A = \max_c \|a_c\|_\infty$,

$$\|r(\pi_i) - r(\pi_j)\|_\infty \le G_A \|\pi_i - \pi_j\|_1.$$

If normalization is $L_{\mathcal{N}}$-Lipschitz on the operating region, Corollary D.9 applies with $L = L_{\mathcal{N}} G_A$. This example does not model finite samples, multiple SGD epochs, momentum, weight decay, or normalization-layer dynamics. The map $P \mapsto \mathrm{TopK}(\mu(P))$ is generally non-injective.

D.6 WHAT BLOCK NORMALIZATION GUARANTEES

For a block with n_ℓ coordinates, write

$$\mathcal{N}_{\epsilon,\ell}(x) = \frac{x}{r_\ell(x) + \epsilon}, \qquad r_\ell(x) = \mathrm{RMS}(x) = \frac{\|x\|_2}{\sqrt{n_\ell}}.$$

Proposition D.10 (Block-scale invariance and local Lipschitz bound)**.** *For $a > 0$ and nonzero x, $\mathcal{N}_{0,\ell}(ax) = \mathcal{N}_{0,\ell}(x)$. For $\epsilon > 0$,*

$$\|\mathcal{N}_{\epsilon,\ell}(ax) - \mathcal{N}_{\epsilon,\ell}(x)\|_\infty = \frac{\epsilon|a-1|\,\|x\|_\infty}{(ar_\ell(x)+\epsilon)(r_\ell(x)+\epsilon)}. \tag{17}$$

If $r_\ell(x) + \epsilon, r_\ell(y) + \epsilon \ge d_0 > 0$ and $\|y\|_\infty \le G$, then

$$\|\mathcal{N}_{\epsilon,\ell}(x) - \mathcal{N}_{\epsilon,\ell}(y)\|_\infty \le L_{\mathcal{N}} \|x - y\|_\infty, \qquad L_{\mathcal{N}} := d_0^{-1} + G d_0^{-2}. \tag{18}$$

Proof. The first two claims follow by substitution. For the last, add and subtract $y/(r_\ell(x)+\epsilon)$ and use $|r_\ell(x) - r_\ell(y)| \le \|x-y\|_2/\sqrt{n_\ell} \le \|x-y\|_\infty$. □

Thus block scaling is removed exactly only for $\epsilon = 0$; for positive ϵ, Eq. (17) quantifies the residual. Any such score perturbation can be inserted into Theorem D.5.

D.7 EMA STABILITY WITH TEMPORALLY DEPENDENT ROUNDS

After EMA burn-in (or with the finite-initialization term retained), for one client and coordinate write $s^{(t)} = \mu^{(t)} + \xi^{(t)}$ and

$$\widetilde{s}^{(t)} = (1-\beta)\sum_{q\geq 0}\beta^q s^{(t-q)}, \qquad \bar{\mu}^{(t)} = (1-\beta)\sum_{q\geq 0}\beta^q \mu^{(t-q)}.$$

We do not assume independent rounds. Instead, assume each coordinate sequence is jointly sub-Gaussian with covariance proxy C satisfying $|C_{qr}| \leq \sigma^2\omega^{|q-r|}$ for $\omega \in [0,1)$; that is, for every finitely supported vector c,

$$\mathbb{E}\exp\left(\lambda\sum_q c_q\xi^{(t-q)}\right) \leq \exp\left(\frac{\lambda^2}{2}\sum_{q,r}c_q c_r C_{qr}\right).$$

Proposition D.11 (EMA noise under temporal dependence)**.** *The EMA noise is sub-Gaussian with proxy variance at most*

$$\sigma^2_{\beta,\omega} = \sigma^2\frac{1-\beta}{1+\beta}\frac{1+\beta\omega}{1-\beta\omega}. \tag{19}$$

Consequently,

$$\Pr\Big(\|\widetilde{s}^{(t)} - \bar{\mu}^{(t)}\|_\infty > u\Big) \leq 2M\exp\left(-\frac{u^2}{2\sigma^2_{\beta,\omega}}\right). \tag{20}$$

Proof. With $c_q = (1-\beta)\beta^q$, the proxy variance is bounded by

$$\sigma^2(1-\beta)^2\sum_{q,r\geq 0}\beta^{q+r}\omega^{|q-r|} = \sigma^2\frac{1-\beta}{1+\beta}\frac{1+\beta\omega}{1-\beta\omega}.$$

Apply a two-sided sub-Gaussian tail bound and union-bound over coordinates; cross-coordinate independence is unnecessary. □

At $\omega = 0$, Eq. (19) recovers $\sigma^2(1-\beta)/(1+\beta)$; as $\omega \uparrow 1$, the upper bound tends to σ^2. A lag-one autocorrelation estimate alone does not certify the geometric envelope. One must estimate an upper envelope over relevant lags or directly estimate the EMA quadratic form $\sum_{q,r} c_q c_r C_{qr}$.

Let $q^{(t)} = \mathcal{N}_\epsilon(\bar{\mu}^{(t)})$ and $H^{(t)} = \mathrm{TopK}(q^{(t)})$. Define

$$\Delta^{(t)} = \min_{u\in H^{(t)},\, v\notin H^{(t)}}\big(q_u^{(t)} - q_v^{(t)}\big).$$

For round r, define the explicit normalization event

$$\mathcal{E}_r(d_0, G_\mathcal{N}) := \Big\{\min_\ell \min\big(r_\ell(\widetilde{s}_\ell^{(r)}) + \epsilon, r_\ell(\bar{\mu}_\ell^{(r)}) + \epsilon\big) \geq d_0, \\ \max_\ell \|\bar{\mu}_\ell^{(r)}\|_\infty \leq G_\mathcal{N}\Big\}. \tag{21}$$

Corollary D.12 (Consecutive-round exact stability)**.** *Assume $H^{(t)} = H^{(t-1)} =: H$, let $\Delta_{\min} = \min\{\Delta^{(t)}, \Delta^{(t-1)}\}$, and suppose $\Pr(\mathcal{E}_r(d_0, G_\mathcal{N})^c) \leq \nu_r$ for $r \in \{t-1, t\}$. Let $L_\mathcal{N} = d_0^{-1} + G_\mathcal{N} d_0^{-2}$ and let $\bar{\sigma}^2_{\beta,\omega}$ upper-bound Eq.* (19) *at both rounds. Then*

$$\Pr[I^{(t)} = I^{(t-1)} = H] \geq 1 - \nu_t - \nu_{t-1} - 4M\exp\left(-\frac{\Delta^2_{\min}}{8L^2_\mathcal{N}\bar{\sigma}^2_{\beta,\omega}}\right). \tag{22}$$

Proof. On $\mathcal{E}_r(d_0, G_\mathcal{N})$, Proposition D.10 bounds normalized score error by $L_\mathcal{N}$ times raw EMA error. Raw error below $\Delta_{\min}/(2L_\mathcal{N})$ therefore preserves the support. Apply Eq. (20) at both rounds and union-bound these failures and the two normalization-event failures. □

Exact recovery is conservative. We denote observed temporal self-overlap by

$$T_i^{(t)} := \frac{|I_i^{(t)} \cap I_i^{(t-1)}|}{K}, \tag{23}$$

avoiding collision with the covariance proxy C.

D.8 A DIRECT, ASSUMPTION-FREE TEMPORAL CHECK

The observed consecutive scores themselves yield a stronger diagnostic than fitting the EMA noise model. Define

$$\delta_i^{(t)} = \|z_i^{(t)} - z_i^{(t-1)}\|_\infty.$$

Corollary D.13 (Observable turnover certificate)**.** *For every client and adjacent-round pair,*

$$1 - T_i^{(t)} \le \frac{1}{K} \min\Big\{ B_{z_i^{(t-1)}}(2\delta_i^{(t)}), B_{z_i^{(t)}}(2\delta_i^{(t)}) \Big\}. \tag{24}$$

Proof. Apply Theorem D.5 in both directions. The number of coordinates replaced is $K(1 - T_i^{(t)})$. □

Equation (24) uses no distributional or noise assumption. Appx. E compares its right-hand side with measured turnover and reports the bound-to-observation ratio rather than claiming tightness a priori.

D.9 COMPUTATION AND AN EXPLICIT COMMUNICATION MODEL

Salience construction, EMA, and block normalization cost $O(M)$ once per client round. Top-K extraction is $O(M)$ expected time by partial selection (or $O(M \log K)$ with a heap). Communication depends on the encoding and must be stated consistently. For $M < 2^{32}$:

$$\begin{aligned} B_{\text{uint32}} &= 32K, \\ B_{\text{packed}} &= K\lceil \log_2 M \rceil, \\ B_{\text{enum}} &= \left\lceil \log_2 \binom{M}{K} \right\rceil = MH_2(\rho_K) + O(\log M), \qquad \rho_K = K/M. \end{aligned} \tag{25}$$

where $H_2(x) = -x \log_2 x - (1-x)\log_2(1-x)$ is binary entropy. Relative to a dense FP32 vector, their idealized bit ratios are respectively ρ_K, $\rho_K \lceil \log_2 M \rceil / 32$, and approximately $H_2(\rho_K)/32$. Thus *SPOS* sends K index items instead of M real values. At fixed ρ_K every lossless representation remains $\Theta(M)$ bits. The benefit is a potentially large constant-factor reduction and removal of real-valued entries, not sublinear asymptotic communication at fixed density. Sublinear bits require $K = o(M)$ or an approximate sketch such as Appx. B.

E EMPIRICAL DIAGNOSTICS AND THEORY CALIBRATION

The theory gives sufficient conditions under which salient-support overlap is informative. Because several conditions are intentionally worst-case, we calibrate them using the actual layer-normalized *SPOS* scores and supports produced during training. We distinguish between (i) empirical properties associated with reliable support comparison—salience concentration and temporal persistence—and (ii) the tightness of the deterministic sufficient-condition bounds on the observed models.

E.1 SALIENCE CONCENTRATION AND TEMPORAL PERSISTENCE

For client i at round t, define

$$\alpha_i^{(t)} = \frac{\|z_i^{(t)} \odot m_i^{(t)}\|_2^2}{\|z_i^{(t)}\|_2^2}, \qquad T_i^{(t)} = \frac{|I_i^{(t)} \cap I_i^{(t-1)}|}{K}, \tag{26}$$

where $z_i^{(t)}$ is the layer-normalized *SPOS* salience, $m_i^{(t)}$ is its binary Top-K mask, and $I_i^{(t)}$ is the corresponding support. The head-energy ratio α measures how much normalized salience energy is retained by the transmitted support, while T measures support persistence across consecutive rounds.

Table 9 shows meaningful concentration in the selected supports. Across the three models included in this diagnostic, *SPOS* retains 56.8–72.9% of the normalized salience energy while selecting only 10% of coordinates. Relative to the selected-coordinate fraction, the support therefore contains approximately $7.3\times$, $5.8\times$, and $5.7\times$ its proportional share of salience energy on FMNIST, CIFAR-10, and 20News, respectively.

Table 9: Diagnostics computed from the actual layer-normalized *SPOS* scores and supports. Head energy is the fraction of normalized salience energy retained by the Top-K support; temporal overlap is the fraction of selected coordinates retained between consecutive rounds.

Model / Dataset	Head energy $\alpha \uparrow$	Temporal overlap $T \uparrow$
CNN / FMNIST	0.7285	0.8874
ResNet18 / CIFAR-10	0.5811	0.7953
BERT-base / 20News	0.5676	0.7066

Table 10: Empirical calibration of the *SPOS* theoretical quantities at the fixed support ratio $K/M = 10\%$. "Turnover ratio" is the deterministic turnover upper bound divided by the observed turnover, reported as the median over cases with nonzero observed turnover. The dependence proxy $\widehat{\omega}$ is defined in Eq. (27).

Model / Dataset	$\widehat{S}_0$	$\kappa(0.05)$ med.	$\kappa(0.05)$ 95th	Turnover ratio	$\widehat{\omega}$
CNN / FMNIST	.386	1.672	1.821	9.84	.822
ResNet18 / CIFAR-10	.261	1.451	1.536	4.93	.813
BERT-base / 20News	.198	1.401	1.447	3.56	.753

The supports are also temporally persistent. Consecutive-round overlap is 0.887 on FMNIST, 0.795 on CIFAR-10, and 0.707 on 20News; thus, approximately 71–89$\%$ of the selected coordinates survive from one round to the next despite continued local training. If the Top-K boundary were dominated by round-specific noise, this overlap would be substantially less stable. The results instead indicate that the EMA-smoothed, layer-normalized supports contain a persistent core across the three architectures evaluated in this diagnostic.

E.2 Calibration of the Sufficient-Condition Bounds

Head concentration and temporal persistence establish the empirical mechanism but do not determine the constants in the sufficient-condition results. We therefore estimate the coordinate-popularity null overlap $\widehat{S}_0$, the turnover bound from saved client/round scores. We also report an empirical temporal-dependence proxy $\widehat{\omega}$. Specifically, if $\widehat{\rho}(\ell)$ is the pooled lag-ℓ autocorrelation of the sampled, temporally centered salience trajectories, we use

$$\widehat{\omega} = \max_{1 \le \ell \le 5} |\widehat{\rho}(\ell)|^{1/\ell}. \tag{27}$$

This quantity diagnoses temporal dependence; it does not by itself verify the population covariance envelope assumed by the dependent-EMA result.

Table 10 separates empirical scale from theorem tightness. First, the coordinate-popularity null is 0.386 on FMNIST, 0.261 on CIFAR-10, and 0.198 on 20News. These values exceed the uniform-support baseline $K/M = 0.10$ in all three cases, showing that some coordinates are selected disproportionately often. We therefore use $\widehat{S}_0$ only as a calibration reference, not as evidence that the observed retrieval signal is entirely client-specific. A global chance correction cannot resolve this issue because it is an affine transformation of S and leaves neighbor rankings unchanged.

Second, the quantile head-dispersion factors are moderate. Across the three models evaluated here, the median $\kappa(0.05)$ ranges from 1.401 to 1.672, and its 95th percentile remains below 1.83. This complements the head-energy result: although coordinates near the Top-K boundary can change, the selected head is not governed by a small number of arbitrarily extreme coordinates.

Third, the deterministic turnover certificate is valid but conservative in magnitude. Its median ratio to observed nonzero turnover is approximately 9.84, 4.93, and 3.56 on FMNIST, CIFAR-10, and 20News, respectively. The bound therefore identifies boundary ambiguity as the relevant stability quantity but should not be interpreted as a tight predictor of empirical turnover. The dependence proxies $\widehat{\omega} = 0.822$, 0.813, and 0.753 also show substantial serial dependence in the sampled salience trajectories, supporting the use of a temporally dependent analysis rather than an i.i.d. round

approximation. As noted above, these empirical values diagnose dependence but do not certify the theorem's population covariance assumption.

Finally, the optimized quantile-based lower bound is strictly positive at both the median and fifth percentile for all architectures evaluated in this diagnostic. Its median (fifth percentile) is 0.097 (0.083) on FMNIST, 0.072 (0.064) on CIFAR-10, and 0.059 (0.055) on 20News. Thus, the quantile formulation avoids the numerically zero lower bound produced by an uncontrolled worst-case head ratio on these scores. It nevertheless remains a conservative one-sided interpretation bound, not an estimator of realized cosine similarity or a ranking guarantee.

Overall, the empirical calibration supports the qualitative chain underlying *SPOS*: normalized salience is concentrated, the selected support persists across rounds, and the relevant continuity and dependence quantities remain finite in practice. At the same time, the turnover ratios in Table 10 show that the worst-case sufficient conditions remain substantially more conservative than the empirical behavior. We therefore use the theory to explain *why* support overlap can preserve relatedness and to identify its stability conditions, rather than to claim that the deterministic bounds tightly predict the observed retrieval metrics.

F Verifiable *SPOS* Reporting: Input Binding and Sampled Consistency

This section presents a protocol design and soundness analysis. We do not report prover/verifier runtime or end-to-end proof overhead; those quantities depend on a concrete proof-system implementation and are left to future systems work. The similarity plane exposes each client's support directly, so an authenticated client can manipulate the affinity graph by reporting coordinates other than those prescribed by *SPOS*. A digital signature authenticates the sender and the report bytes, but not the computation that produced them. Likewise, a proof that the report is the Top-K of a score vector chosen and committed by the same client is self-referential: for any desired support I, the client can choose a compatible score vector and prove the resulting statement. A meaningful integrity statement must therefore bind the reported support to a previously fixed protocol input and must state explicitly what that input represents.

Guarantee and anchoring boundary. Let $c_{\Delta,i}^{(t)}$ be a binding and hiding vector commitment to the local update $\Delta w_i^{(t)}$. In a task with an aggregation plane (T1), we require $c_{\Delta,i}^{(t)}$ to be either the same commitment accepted for the client's secure-aggregation input or linked to that commitment by an equality proof. Systems such as RoFL and ACORN illustrate privacy-preserving validation of secure-aggregation inputs (Lycklama et al., 2023; Bell et al., 2023); a concrete integration must additionally implement the stated commitment reuse or equality link. Under that integration, the *SPOS* report is certified against the update that the client actually contributes to aggregation.

In an affinity-only task (T2 and T3), no externally consumed update exists. A commitment to a client-chosen update or local model can therefore certify only *internal consistency*: the support was computed from the committed input. It cannot certify that the input resulted from honest local training, and it does not prevent a malicious client from choosing an update that induces a target support. Obtaining the stronger statement in this regime would require an additional anchor, such as an attested training execution, a proof of the local training computation, or later protocol use of the same committed model. We do not assume such an anchor here.

Fixed-point statement. The proof concerns the fixed-point *SPOS* implementation, not ideal real arithmetic. Let Q be the public scale, and let all signed update coordinates, EMA values, block normalizers, and normalized scores use a specified integer encoding and rounding rule. The circuit range-checks every opened value so that absolute value, multiplication, division, and comparison agree with signed integer arithmetic and cannot wrap modulo the proof field. If q_j is the quantized normalized score, deterministic tie-breaking uses

$$a_j = q_j(M+1) + (M-j), \tag{28}$$

whose values are also range-checked. This construction proves Top-K for the same quantized scores used by the implementation.

State and report commitments. Let $c_{\bar{s},i}^{(t-1)}$ and $c_{\bar{s},i}^{(t)}$ commit to the previous and current EMA vectors, respectively, and let $c_{q,i}^{(t)}$ commit to the quantized normalized score vector. The initial EMA commitment is fixed by the protocol, and every accepted round publishes $c_{\bar{s},i}^{(t)}$ as the state commitment consumed at the client's next participating round. This prevents a client from silently resetting or changing its EMA history. Let $c_{r,i}^{(t)}$ commit to the per-block fixed-point normalizers, and let

$$d_{I,i}^{(t)} = \mathsf{MerkleRoot}\Big(\mathrm{sort}(\widetilde{I}_i^{(t)})\Big)$$

bind the proof to the exact public token list. The coordinator independently checks that this list contains exactly K distinct tokens in $[M]$ and recomputes $d_{I,i}^{(t)}$.

Tokenization uses a specified format-preserving keyed permutation $\pi_{\mathsf{rk}_t} : [M] \to [M]$; a truncated keyed hash is not sufficient because collisions would invalidate support equality. The high-entropy round key is bound through the domain-separated tag

$$c_{\mathsf{rk}_t} = \mathsf{Poseidon}\big(\texttt{SPOS-KEY}, \mathsf{sid}, t, \mathsf{rk}_t\big),$$

where sid is a session identifier. All accepted reports in the round must carry the same tag, and the proof demonstrates knowledge of a key opening the tag and producing the report tokens. The signed transcript is

$$\mathsf{Tr}_i^{(t)} = \big(\mathsf{sid}, t, i, K, Q, \beta, \mathsf{algID}, c_{\Delta,i}^{(t)}, c_{\bar{s},i}^{(t-1)}, c_{\bar{s},i}^{(t)}, c_{r,i}^{(t)}, c_{q,i}^{(t)}, c_{\mathsf{rk}_t}, d_{I,i}^{(t)}, \widetilde{I}_i^{(t)}\big). \tag{29}$$

where algID fixes the commitment schemes, fixed-point rules, normalization, tie-breaking, and token permutation.

Exact pipeline subproof. Sampling only the ranking predicate is not enough to anchor the report: a client could otherwise alter a small number of unsampled EMA coordinates, change a block normalizer, and thereby move many coordinates across the global Top-K boundary. We therefore require a zero-knowledge proof Π_{pipe} of the complete score-construction pipeline. It proves, for every coordinate, that the committed update and previous EMA open $c_{\Delta,i}^{(t)}$ and $c_{\bar{s},i}^{(t-1)}$, that

$$\bar{s}_{i,j}^{(t)} = \beta \bar{s}_{i,j}^{(t-1)} + (1-\beta)|\Delta w_{i,j}^{(t)}|, \tag{30}$$

and that the resulting vector opens $c_{\bar{s},i}^{(t)}$. For every block $\ell \in \mathcal{L}$, the same proof establishes

$$U_\ell = \sum_{j \in \ell} \big(\bar{s}_{i,j}^{(t)}\big)^2.$$

If r_ℓ is the fixed-point floor of the block RMS, the argument also proves the corresponding rounding inequalities

$$n_\ell r_\ell^2 \le U_\ell < n_\ell (r_\ell + 1)^2,$$

after applying the public scale convention. Finally, it proves that the normalizers open $c_{r,i}^{(t)}$ and that every $q_{i,j}^{(t)} = \mathsf{NormalizeQ}(\bar{s}_{i,j}^{(t)}, r_{\ell(j)}, Q, \epsilon)$ under the public rounding rule, with the resulting score vector opening $c_{q,i}^{(t)}$. The exact RMS and division inequalities must be adjusted mechanically if another public rounding rule is used. The proof may be instantiated with a suitable zero-knowledge polynomial IOP, sum-check system, or a full arithmetic-circuit proof, but it must include batched range and sign constraints for all coordinates. A single polynomial opening at one random point does not establish these relations.

Unpredictable audit challenge. The audit set must be unknown when the client fixes Eq. (29). Hashing only client-chosen commitments is insufficient because a last mover can grind its commitment. We instead use a public beacon sampled after the commitment deadline, or an equivalent commit–reveal seed to which the coordinator commits before receiving client transcripts. After the seed R_t becomes public, the verifier derives s distinct uniform coordinates as

$$S = \mathsf{HashToDistinct}\big([M], s; R_t, \mathsf{sid}, t, \mathsf{H}(\mathsf{Tr}_1^{(t)} \| \cdots \| \mathsf{Tr}_N^{(t)})\big).$$

An abort after the challenge is treated as a rejected report; the same session identifier and transcript may not be retried with fresh randomness.

Table 11: Lower bound $1 - e^{-\eta\rho s}$ on detecting a fixed support manipulation of relative size η, shown for $\rho = K/M = 0.10$. This table reports only the sampling component of soundness; cryptographic soundness and pipeline-proof errors are additional negligible terms.

	$s = 1024$	$s = 4096$	$s = 16384$
$\eta = 0.01$	0.641	0.983	> 0.999
$\eta = 0.05$	0.994	> 0.999	> 0.999
$\eta = 0.10$	> 0.999	> 0.999	> 0.999

Sampled zero-knowledge relation. For each $j \in S$, the proof opens $q_{i,j}^{(t)}$ against $c_{q,i}^{(t)}$ inside the zero-knowledge circuit. The correctness of this score relative to the aggregation input and the chained EMA state is supplied by Π_{pipe}. The sampled proof then computes a_j from Eq. (28) and $y_j = \pi_{\mathsf{rk}_t}(j)$, and proves against the authenticated token-set root $d_{I,i}^{(t)}$ that

$$\begin{cases} y_j \in \widetilde{I}_i^{(t)} & \Longrightarrow \ a_j \geq \tau, \\ y_j \notin \widetilde{I}_i^{(t)} & \Longrightarrow \ a_j < \tau, \end{cases} \tag{31}$$

for one witness threshold τ shared by all sampled coordinates. Membership and non-membership can be proven with authenticated paths in a sorted Merkle-set representation, without revealing y_j, its position, any update coordinate, or any salience value. The public statement of the proof is a digest of Eq. (29), the beacon value R_t, the audit size s, and the pipeline-proof transcript. The coordinator accepts only if the signature, session and round identifiers, cardinality and distinctness checks, key-tag consensus, Π_{pipe}, and the sampled proof all verify.

Proposition F.1 (Soundness of the sampled support audit)**.** *Let $I^\star = \mathrm{TopK}(a, K)$ be the support prescribed by the committed fixed-point state, and let the preimage of the reported token set under the committed permutation be I, where $|I \triangle I^\star| = 2\eta K$ and ηK is an integer. If S consists of s coordinates sampled uniformly without replacement after the report is committed, then*

$$\Pr[\text{accept} \mid I] \leq \frac{\binom{M-\eta K}{s}}{\binom{M}{s}} + \mathrm{negl}(\lambda) \leq \left(1 - \frac{\eta K}{M}\right)^s + \mathrm{negl}(\lambda) \leq e^{-\eta\rho s} + \mathrm{negl}(\lambda), \qquad \rho = K/M, \tag{32}$$

where λ is the cryptographic security parameter. Moreover, for every other K-element support J,

$$\left| \frac{|I \cap J|}{K} - \frac{|I^\star \cap J|}{K} \right| \leq \eta.$$

Proof. Order $I^\star \setminus I = \{u_1, \ldots, u_{\eta K}\}$ and $I \setminus I^\star = \{v_1, \ldots, v_{\eta K}\}$ in decreasing augmented score. Because the augmented scores are distinct, $a_{u_m} > a_{v_m}$ for every m. For any fixed τ, if $a_{u_m} \geq \tau$, then the omitted coordinate u_m violates the non-membership implication in Eq. (31); otherwise $a_{v_m} < a_{u_m} < \tau$, and the included coordinate v_m violates the membership implication. The pairs are disjoint, so at least ηK coordinates violate the sampled predicate. The first term in Eq. (32) is the probability that a size-s sample without replacement misses all violators; the remaining inequalities are standard. The negligible term accounts for the soundness errors of the commitments and proof systems. Replacing ηK elements of a K-set changes its intersection with any fixed K-set by at most ηK, proving the similarity bound. □

For target sampling failure probability δ_{audit}, it is sufficient to choose

$$s \geq \frac{\log(1/\delta_{\text{audit}})}{\eta\rho}.$$

Thus the audit size depends jointly on the smallest manipulation fraction η to be detected and the support fraction ρ. In our experiments, $\rho = K/M = 0.10$ for all models, and Table 11 reports the corresponding sampling-detection bounds.

Table 12: Components that must be included in an end-to-end overhead measurement. A constant-size core SNARK does not make the complete protocol constant-cost.

Component	Dominant prover work	Required reporting
Vector and EMA commitments	$\Omega(M)$	Commitment construction and sampled-opening costs
Pipeline proof (EMA, ranges, and normalization)	$\Omega(M)$	Proof bytes, prover time, verifier time, and block count
Sampled consistency SNARK	Depends on s, openings, PRP, and comparison circuit	Total constraints and setup assumptions for each supported s
Report authentication	Linear in the public token list for hashing/signing	Signature, roots, tags, beacon data, and complete serialized metadata

Cost accounting. Sampling reduces the number of coordinate-wise *SPOS* relations placed in the SNARK, but it does not make total client work independent of model size. The client must still compute *SPOS* and the vector commitments over M coordinates, and Π_{pipe} must process every update and EMA coordinate. With randomized-leaf Merkle commitments, the sampled selection circuit contains $O(s(\log M + \log K + C_\pi + C_{\mathrm{cmp}}))$ constraints, where C_π is the cost of the chosen permutation and C_{cmp} covers range-safe score construction and threshold comparison. The prover work for Π_{pipe} is at least linear in M, while its proof size and verification cost depend on the selected proof system, range-proof strategy, and number of parameter blocks.

For example, a Groth16 core proof contains three group elements, but quoting only its compressed size omits Π_{pipe}, commitment roots and openings, the token-set root, key tag, beacon binding, and signature. Consequently, total metadata and runtime must be measured from a concrete end-to-end implementation; they cannot be inferred from the core SNARK size. Groth16 also requires a circuit-specific setup for each supported circuit shape unless a compatible setup strategy is specified.

Security and privacy scope. The sampled audit gives the following narrow guarantee. In T1, and only when $c^{(t)}_{\Delta,i}$ is linked to the accepted aggregation input, a client that commits a report differing from the prescribed support by $2\eta K$ is accepted with probability at most Eq. (32), subject to the stated cryptographic assumptions. In T2 and T3, the same bound certifies consistency only with a client-chosen committed input. The protocol does not prove honest local training, benign data, a benign update, or truthful client metadata, and it does not prevent a client from aborting after commitment.

Zero knowledge hides the committed update, EMA values, block normalizers, token preimages, and threshold beyond what follows from the public statement. It does not hide the public token set or the within-round *SPOS* matrix and is not a differential-privacy mechanism. The claim also relies on binding and hiding commitments, a zero-knowledge knowledge-sound proof system, an injective secure token permutation, correct fixed-point range constraints, and an unbiased post-commitment challenge. These assumptions and the concrete implementation parameters must be reported before practical overhead or deployment security is claimed.

G Extended Experimental Results

G.1 Hyperparameter Tuning Protocol

Experiments run on a GPU cluster node equipped with an AMD EPYC 7543 CPU (128 cores), 500 GB RAM, and $4\times$ NVIDIA A100 (40,GB) GPUs. We use a validation-only hyperparameter tuning protocol. For each dataset/model/split, 10% of each client's local data is reserved for validation diagnostics and hyperparameter selection and is excluded from final test reporting. Once selected, hyperparameters are fixed and all reported results are evaluated on the held-out test set and averaged over five random seeds. The common *SPOS* support ratio $K/M = 10\%$ is fixed across the main experiments rather than selected separately using the relatedness oracle; the support-size sweep in Appx. H.2 is a retrospective sensitivity analysis.

For CNN/ResNet models, we use SGD with momentum 0.9, weight decay 5×10^{-4}, learning rate 10^{-3}, seven local epochs, batch size 64, and cosine learning-rate decay. BERT-base uses AdamW with learning rate 2×10^{-5}, weight decay 10^{-2}, 10% warmup, maximum sequence length 128, and linear decay. Wav2Vec2 on SpeechCommands is fine-tuned using its associated audio feature extractor, with raw waveforms converted to model inputs and padded or truncated to a common length within each batch. FMNIST uses normalization only, CIFAR-10 uses random crop, horizontal flip, and normalization, and 20News uses the BERT-base tokenizer with standard truncation and padding.

For *SPOS*, salience is computed from the magnitude of the final local update and smoothed with EMA coefficient $\beta = 0.6$. We fix $K/M = 10\%$ for all downstream CPFL, neighbor-selection, and distribution-alignment experiments. Main utility results use raw coordinate IDs; tokenized IDs are evaluated separately in the security analysis and preserve within-round *SPOS* similarities. Downstream task hyperparameters, including the number of clusters and neighbor cutoff, are held fixed across similarity methods, so the test-time comparison changes only the similarity primitive.

Implementation details. The primary client-relatedness and CPFL experiments use $N = 30$, while scalability analysis evaluate *SPOS* with up to 400 clients. We use common parameter coordinate system within each model. For *SPOS*, client i forms the local update $\Delta w_i^{(t)} = w_i^{(t+1)} - w^{(t)}$, uses $s_i^{(t)} = |\Delta w_i^{(t)}|$ as salience, and updates the EMA once per communication round as $\bar{s}_i^{(t)} = 0.6\, \bar{s}_i^{(t-1)} + 0.4\, s_i^{(t)}$. Each layer-wise parameter block is normalized independently by its RMS before concatenation, after which a single global Top-K operation selects $K/M = 10\%$ of all coordinates. Top-K ties are resolved deterministically by parameter index. Thus all clients of a given model use the same architecture, parameter ordering, support size, and coordinate space, and only the selected coordinate identities enter the *SPOS* similarity computation.

We evaluate two label-heterogeneity families. Dir(α) draws client class proportions from a Dirichlet distribution, using $\alpha = 0.3$ and 0.1; the smaller value produces more concentrated client label distributions and hence stronger label skew. Patho($p\%$) restricts each client to $p\%$ of the label space, using $p = 20\%$ and 30%; therefore Patho(20%) is the more label-restricted setting. Client partitions are generated once for a seed and reused across all compared similarity methods, together with the same initialization and training budget. Each client reserves 10% of its local data for validation, and this validation subset is excluded from final held-out test reporting.

For CNN/FMNIST and ResNet18/CIFAR-10, local optimization uses SGD with learning rate 10^{-3}, momentum 0.9, weight decay 5×10^{-4}, $E = 7$ local epochs, batch size 64, and cosine learning-rate decay. BERT-base/20News uses AdamW with learning rate 2×10^{-5}, weight decay 10^{-2}, 10% warmup, maximum sequence length 128, and linear decay. Wav2Vec2/SpeechCommands is fine-tuned from the pretrained speech model using its associated feature extractor, with waveforms padded or truncated to a common length within each batch. All reported configurations are evaluated over five random seeds.

For CPFL, we fix $G = 3$ clusters for the 30 clients and recompute the partition every 5 communication rounds. CPFL runs for 60 rounds, giving 12 scheduled clustering refresh points, whereas affinity-only experiments run for 10 rounds. The EMA is updated from the corresponding common-reference client updates and is maintained across rounds rather than recomputed per mini-batch. All affinity methods use identical client partitions, model initialization, training budget, cluster count, refresh frequency, and five-seed protocol, so the principal experimental variable is the client-relatedness measure.

For CPFL, all client models are evaluated on the common global test set rather than on client-specific test distributions. Reported accuracy is averaged across clients and then across the five matched seeds. We therefore interpret this metric as global predictive performance of the cluster-specialized models, rather than as accuracy specifically on each client's local distribution.

The factorial representation diagnostic uses CNN/FMNIST, $N = 30$ clients, Dir(0.3), 10 communication rounds, seven local epochs, $K/M = 10\%$, $\beta = 0.6$, and five prespecified seeds. The heterogeneity and checkpoint-relative experiments also use five seeds. Feature-, concept-, and quantity/compute-shift experiments use exactly equal label marginals, and the checkpoint experiment compares 15 models adapted from one shared reference checkpoint on disjoint data from five domains.

Table 13: Full relatedness comparison on CNN/FMNIST under Dir(0.3).

Method	R@6↑	R@12↑	JS@6↓	JS@12↓	Spearman↑	Kendall↑
SPOS	.656	.681	.124	.143	.786	.582
Cosine update	.650	.769	.124	.142	.784	.598
Euclidean update	.644	.714	.129	.151	.658	.479
Cosine full salience	.672	.689	.140	.159	.627	.461
Euclidean full salience	.594	.622	.155	.169	.471	.330
Cosine Top-K values	.639	.761	.124	.140	.773	.587
Euclidean Top-K values	.628	.706	.134	.153	.645	.468
FedFomo (Zhang et al., 2021)	.522	.608	.134	.170	.449	.362
pFedSim-head (Tan et al., 2023)	.594	.683	.139	.160	.597	.429
CKA representation (Kornblith et al., 2019)	.361	.508	.183	.192	.270	.180
Signed support (same K)	.683	.764	.121	.140	.799	.614
Signed support (same budget)	.683	.761	.121	.140	.798	.613
Signed support (split budget)	.683	.764	.121	.141	.799	.614
Sparse cosine, 4-bit	.639	.739	.128	.144	.743	.555
Sparse cosine, 8-bit	.633	.761	.125	.141	.771	.584
Distribution cosine	.800	.886	.105	.123	.941	.791
Distribution Hellinger	.944	.933	.101	.125	.990	.919
Oracle distribution (JS)	1.000	1.000	.101	.124	1.000	1.000

For clustered personalized FL, we fix $G = 3$ clusters and refresh the client partition every five rounds; our experimental wrapper is therefore not a reimplementation of the recursive CFL procedure of Sattler et al. (2020). CPFL runs for 60 federated rounds, whereas affinity-only experiments run for 10 rounds. All compared methods use the same client partitions, initialization, training budget, and five random seeds.

G.2 Full Relatedness Comparisons

This section extends Sec. 5.1 with all relatedness baselines evaluated by the current suite. We group methods into (i) update/value-based similarities, (ii) FL-specific and representation similarities, and (iii) distribution-based evaluation references. Distribution-based methods use private label histograms and are not deployable baselines. All tables use Dir(0.3).

Comparison with raw-update cosine across architectures. Against the strongest deployable baseline, *SPOS* equals or beats full-update cosine on ResNet18, BERT-base, and Wav2Vec2 (R@12 $+.052$, $+.010$, $+.018$) and trails on CNN/FMNIST ($-.088$). The sign flips at the smallest model, consistent with the head–tail premise: a 0.42M-parameter CNN offers little redundant tail for a support to summarize. Retrieval and divergence are not perfectly aligned—on ResNet18 cosine attains lower JS@6 ($.105$ vs $.109$) while retrieving fewer oracle neighbors ($.700$ vs $.756$).

Normalization and truncation, not magnitude alone, carry the sign-free signal. "Cosine full salience" discards signs without block normalization or Top-K selection and is the most erratic family here: best on Wav2Vec2 ($.772/.822$), collapsing on ResNet18 ($.406/.567$, Kendall $.323$), where *SPOS* reaches $.756/.808$ from the same magnitude signal. Signed supports agree from the other side: they improve on *SPOS* only for CNN/FMNIST ($.683/.764$), so sign information helps where the support itself is least informative.

FL-specific baselines trail. FedFomo's R@12 falls below its R@6 on two models, indicating that utility probing ranks close peers but not the tail; pFedSim-head is competitive only where the classifier head carries the signal (ResNet18 $.750/.775$) and needs real-valued head parameters; CKA is weakest throughout (R@6 $.361$–$.461$). The distribution rows are near-ceiling by construction and bound the metric, not the task.

Table 14: Full relatedness comparison on ResNet18/CIFAR-10 under Dir(0.3).

Method	R@6↑	R@12↑	JS@6↓	JS@12↓	Spearman↑	Kendall↑
SPOS	.756	.808	.109	.136	.780	.588
Cosine update	.700	.756	.105	.136	.778	.586
Euclidean update	.508	.647	.139	.149	.405	.278
Cosine full salience	.406	.567	.182	.189	.461	.323
Euclidean full salience	.439	.558	.182	.185	.417	.287
Cosine Top-K values	.700	.756	.105	.136	.777	.584
Euclidean Top-K values	.511	.653	.139	.149	.426	.292
FedFomo (Zhang et al., 2021)	.606	.603	.129	.171	.531	.437
pFedSim-head (Tan et al., 2023)	.750	.775	.116	.142	.766	.585
CKA representation (Kornblith et al., 2019)	.439	.583	.183	.184	.468	.324
Signed support (same K)	.650	.739	.122	.146	.736	.546
Signed support (same budget)	.650	.739	.122	.146	.737	.546
Signed support (split budget)	.650	.742	.122	.146	.737	.546
Sparse cosine, 4-bit	.661	.733	.116	.137	.764	.572
Sparse cosine, 8-bit	.700	.753	.115	.136	.776	.584
Distribution cosine	.794	.906	.115	.125	.943	.792
Distribution Hellinger	.956	.936	.103	.127	.990	.916
Oracle distribution (JS)	1.000	1.000	.101	.125	1.000	1.000

Table 15: Full relatedness comparison on BERT-base/20News under Dir(0.3).

Method	R@6↑	R@12↑	JS@6↓	JS@12↓	Spearman↑	Kendall↑
SPOS	.777	.824	.152	.176	.848	.658
Cosine update	.756	.814	.156	.176	.851	.656
Euclidean update	.756	.814	.165	.183	.830	.636
Cosine full salience	.756	.744	.161	.189	.707	.515
Euclidean full salience	.650	.669	.194	.212	.622	.447
Cosine Top-K values	.744	.822	.154	.178	.848	.653
Euclidean Top-K values	.750	.797	.169	.184	.789	.595
FedFomo (Zhang et al., 2021)	.706	.669	.168	.193	.640	.509
pFedSim-head (Tan et al., 2023)	.717	.803	.173	.189	.841	.656
CKA representation (Kornblith et al., 2019)	.450	.575	.215	.220	.419	.285
Signed support (same K)	.756	.822	.155	.176	.850	.655
Signed support (same budget)	.756	.822	.155	.176	.850	.655
Signed support (split budget)	.756	.822	.155	.176	.850	.655
Sparse cosine, 4-bit	.744	.825	.154	.176	.847	.652
Sparse cosine, 8-bit	.750	.828	.153	.176	.845	.650
Distribution cosine	.822	.844	.152	.173	.914	.747
Distribution Hellinger	.906	.942	.150	.173	.987	.906
Oracle distribution (JS)	1.000	1.000	.152	.175	1.000	1.000

G.3 Coordinate-system controls for checkpoint relatedness

The primary checkpoint experiment intentionally uses a shared reference model and parameter ordering. We test the necessity of this condition with three controls. First, we apply a hidden-unit permutation together with the inverse permutation in the following layer, preserving the represented function. Second, we train the same architecture from an independent reference initialization. Third, we compare behavior across different architectures, where no one-to-one parameter-index correspondence exists.

The permutation control leaves predictions unchanged but sharply reduces raw support overlap; applying the known inverse alignment restores overlap exactly. The independent-initialization control

Table 16: Full relatedness comparison on Wav2Vec2/SpeechCommands under Dir(0.3).

Method	R@6↑	R@12↑	JS@6↓	JS@12↓	Spearman↑	Kendall↑
SPOS	.770	.818	.186	.198	.882	.673
Cosine update	.733	.800	.186	.200	.839	.650
Euclidean update	.650	.764	.202	.210	.732	.537
Cosine full salience	.772	.822	.189	.202	.848	.665
Euclidean full salience	.628	.744	.209	.220	.686	.503
Cosine Top-K values	.706	.806	.187	.199	.853	.668
Euclidean Top-K values	.639	.781	.203	.209	.747	.553
FedFomo (Zhang et al., 2021)	.711	.719	.190	.210	.702	.557
pFedSim-head (Tan et al., 2023)	.678	.769	.199	.214	.756	.563
CKA representation (Kornblith et al., 2019)	.461	.617	.235	.239	.475	.328
Signed support (same K)	.683	.775	.192	.203	.786	.592
Signed support (same budget)	.683	.772	.192	.203	.786	.591
Signed support (split budget)	.683	.772	.192	.203	.786	.591
Sparse cosine, 4-bit	.711	.797	.186	.200	.845	.658
Sparse cosine, 8-bit	.711	.808	.187	.198	.856	.672
Distribution cosine	.794	.836	.185	.200	.909	.738
Distribution Hellinger	.906	.928	.179	.199	.986	.901
Oracle distribution (JS)	1.000	1.000	.179	.200	1.000	1.000

Table 17: Coordinate-system stress tests, averaged over five seeds. Functional similarity is behavior-based. Raw and aligned columns report *SPOS*; an em dash means that a common parameter-index space is unavailable.

Control	Functional similarity	Raw *SPOS*	Aligned *SPOS*
Function-preserving hidden permutation	1.000	.114	1.000
Independent reference initialization	.957	.125	–
Different architecture	.907	–	–

likewise shows that behavioral similarity does not imply coordinate correspondence. For different architectures, parameter-index overlap is not merely low but undefined. The method should therefore be described as checkpoint-relative relatedness in a shared coordinate system, not as a permutation-invariant or cross-architecture model similarity.

G.4 Full Results of Clustered Personalized Federated Learning

We provide the complete results for the clustered personalized federated learning (CPFL) task in Table 18. Clients are clustered from their pairwise relatedness and aggregate only within the resulting cluster. We compare *SPOS* against unpersonalized FedAvg, cosine and Euclidean affinities computed from dense update values, Oracle-JS computed directly from the true client label distributions, and Global *SPOS*, which applies a single global Top-K selection without layer normalization. Oracle-JS is an evaluation-only reference because it assumes access to private client label distributions. In contrast, *SPOS* constructs relatedness using only layer-normalized salient coordinate identities.

***SPOS* retains the utility of value-bearing affinities.** Despite removing update magnitudes and signs from the relatedness channel, *SPOS* remains close to affinities computed from full dense updates. Across all settings, its mean accuracy remains within 0.93 percentage points of the stronger dense baseline whenever it does not exceed it.

The correspondence is particularly clear under Dir(0.3). *SPOS* achieves 86.82% on FMNIST versus 86.72%/85.42% for Euclidean/cosine, 65.71% on CIFAR-10 versus 64.09%/65.59%, and 71.26% on 20News versus 71.12%/70.96%. On SpeechCommands, it reaches 96.75%, compared with 96.63%

Table 18: T1: CPFL test accuracy (%), means over five paired seeds across non-IID splits; paired-difference uncertainty is reported in Appx. G.4.

Model / Dataset	**Method**	**Dir**(0.3)	**Dir**(0.1)	**Patho**(20%)	**Patho**(30%)
CNN / FMNIST	Local only	77.28	81.32	54.92	62.82
	FedAvg	86.40	83.92	57.63	66.48
	Cosine update	85.42	84.27	70.97	79.55
	Euclidean update	86.72	84.87	60.97	77.33
	Oracle-JS	86.98	85.48	73.15	79.85
	Global *SPOS* (w/o LN)	86.32	84.68	71.23	74.62
	SPOS	86.82	85.13	72.18	78.62
ResNet18 / CIFAR-10	Local only	61.13	55.85	29.86	29.82
	FedAvg	62.95	56.13	31.88	32.35
	Cosine update	65.59	62.40	33.85	34.95
	Euclidean update	64.09	58.28	33.89	37.25
	Oracle-JS	65.97	64.01	35.85	37.29
	Global *SPOS* (w/o LN)	64.17	60.94	32.10	30.14
	SPOS	65.71	61.54	33.36	36.63
BERT-base / 20News	Local only	68.86	65.26	54.30	51.93
	FedAvg	70.00	70.61	58.31	59.65
	Cosine update	70.96	73.68	64.21	70.88
	Euclidean update	71.12	71.84	63.92	71.32
	Oracle-JS	71.47	74.72	67.33	76.93
	Global *SPOS* (w/o LN)	69.91	69.47	59.04	62.46
	SPOS	71.26	73.32	64.58	72.09
Wav2Vec2 / SpeechCommands	Local only	81.63	86.25	80.51	84.20
	FedAvg	85.69	86.14	81.80	84.06
	Cosine update	97.04	96.50	90.92	94.13
	Euclidean update	96.63	96.30	89.80	94.51
	Oracle-JS	97.15	96.98	92.70	96.90
	Global *SPOS* (w/o LN)	95.81	96.42	86.52	91.81
	SPOS	96.75	96.63	90.78	94.68

Table 19: Paired CPFL test-accuracy differences in percentage points across five matched seeds. Each entry gives the mean paired difference and its 95% confidence interval computed from the five seed-level differences. Positive values favor *SPOS*.

Model / split	***SPOS*** $-$ **cosine, 95% interval**	***SPOS*** $-$ **Global *SPOS*, 95% interval**
CNN/FMNIST, Dir(0.3)	$+1.40$; $[+0.92, +1.88]$	$+0.50$; $[+0.08, +0.92]$
CNN/FMNIST, Dir(0.1)	$+0.86$; $[+0.34, +1.38]$	$+0.45$; $[-0.03, +0.93]$
CNN/FMNIST, Patho(20%)	$+1.21$; $[+0.60, +1.82]$	$+0.95$; $[+0.38, +1.52]$
CNN/FMNIST, Patho(30%)	-0.93; $[-1.61, -0.25]$	$+4.00$; $[+3.37, +4.63]$
ResNet18/CIFAR-10, Dir(0.3)	$+0.12$; $[-0.44, +0.68]$	$+1.54$; $[+1.03, +2.05]$
ResNet18/CIFAR-10, Dir(0.1)	-0.86; $[-1.48, -0.24]$	$+0.60$; $[+0.03, +1.17]$
ResNet18/CIFAR-10, Patho(20%)	-0.49; $[-1.23, +0.25]$	$+1.26$; $[+0.57, +1.95]$
ResNet18/CIFAR-10, Patho(30%)	$+1.68$; $[+0.90, +2.46]$	$+6.49$; $[+5.75, +7.23]$
BERT-base/20News, Dir(0.3)	$+0.30$; $[-0.14, +0.74]$	$+1.35$; $[+0.95, +1.75]$
BERT-base/20News, Dir(0.1)	-0.36; $[-0.85, +0.13]$	$+3.85$; $[+3.40, +4.30]$
BERT-base/20News, Patho(20%)	$+0.37$; $[-0.21, +0.95]$	$+5.54$; $[+5.01, +6.07]$
BERT-base/20News, Patho(30%)	$+1.21$; $[+0.58, +1.84]$	$+9.63$; $[+9.04, +10.22]$
Wav2Vec2/SpeechCommands, Dir(0.3)	-0.29; $[-0.60, +0.02]$	$+0.94$; $[+0.65, +1.23]$
Wav2Vec2/SpeechCommands, Dir(0.1)	$+0.13$; $[-0.21, +0.47]$	$+0.21$; $[-0.10, +0.52]$
Wav2Vec2/SpeechCommands, Patho(20%)	-0.14; $[-0.57, +0.29]$	$+4.26$; $[+3.86, +4.66]$
Wav2Vec2/SpeechCommands, Patho(30%)	$+0.55$; $[+0.08, +1.02]$	$+2.87$; $[+2.43, +3.31]$

for Euclidean and 97.04% for cosine. Hence, *SPOS* exceeds both dense baselines on three of four Dir(0.3) settings and is only 0.29 points below the strongest dense baseline on SpeechCommands.

The same pattern largely persists under Dir(0.1). On FMNIST, *SPOS* reaches 85.13%, exceeding cosine (84.27%) and Euclidean (84.87%). On CIFAR-10 and 20News, its gaps to the strongest dense baseline are 0.86 and 0.36 points, respectively. On SpeechCommands, *SPOS* reaches 96.63%, exceeding cosine (96.50%) and Euclidean (96.30%).

Relatedness is most useful under severe heterogeneity. *SPOS* outperforms FedAvg in all 16 settings. On FMNIST, the gains over FedAvg are 14.55 points on Patho(20%) and 12.14 on Patho(30%); on CIFAR-10 they are 21.48 and 19.28 points. On 20News Patho(30%), accuracy increases from

59.65% to 72.09%, a gain of 12.44 points. On SpeechCommands, *SPOS* improves over FedAvg by 8.98 points on Patho(20%) and 10.62 on Patho(30%).

SPOS also remains competitive with dense similarities in these difficult settings. It exceeds both cosine and Euclidean on FMNIST Patho(20%) and on both 20News pathological splits. On SpeechCommands, it is only 0.14 points below cosine on Patho(20%) and exceeds both dense baselines on Patho(30%), reaching 94.68%. Where a dense baseline is stronger, the largest remaining gap is 0.93 points on FMNIST Patho(30%).

Layer normalization matters under heterogeneous data. The comparison against Global *SPOS* isolates the effect of layer-wise normalization before global Top-K selection. Layer-normalized *SPOS* improves over Global *SPOS* in all settings. The gains are especially large under pathological heterogeneity: $+4.00$ points on FMNIST Patho(30%), $+1.26$/$+6.49$ on CIFAR-10, $+5.54$/$+9.63$ on 20News, and $+4.26$/$+2.87$ on SpeechCommands for Patho(20%)/Patho(30%). On BERT Dir(0.1), normalization also improves accuracy from 69.47% to 73.32%.

These results are consistent with the motivation for layer normalization: raw global Top-K can be biased toward blocks with larger native salience scales, whereas normalization reduces this block-scale bias. The effect is not unconditional, but it substantially improves support quality in most heterogeneous settings.

Comparison with the label-distribution oracle. Oracle-JS is a privileged reference because it directly uses each client's true label distribution. Nevertheless, *SPOS* approaches it closely in several settings: the gaps are 0.16 points on FMNIST Dir(0.3), 0.26 on CIFAR-10 Dir(0.3), 0.21 on 20News Dir(0.3), and 0.35 on SpeechCommands Dir(0.1). Larger gaps appear under pathological partitions, most prominently BERT-base Patho(30%), where Oracle-JS reaches 76.93% versus 72.09% for *SPOS*. We therefore treat Oracle-JS as an evaluation reference rather than a deployable competitor or an upper bound.

Paired CPFL accuracy differences. Table 19 reports paired accuracy differences across the five seeds. *SPOS* remains close to dense cosine across majority of the settings, with mean differences ranging from -0.93 to $+1.68$ percentage points. In contrast, *SPOS* improves over Global *SPOS*, with gains from $+0.21$ to $+9.63$ points, supporting the benefit of layer normalization while preserving utility comparable to the dense baseline.

Overall, the full CPFL results support two complementary conclusions. First, salient-support overlap retains sufficient relatedness information to provide personalization comparable to dense value-bearing similarities. Second, layer normalization materially strengthens the representation across many heterogeneous settings, including ResNet18, BERT, and Wav2Vec2. Together, these results support *SPOS* as a compact, index-only affinity primitive rather than a replacement for the underlying FL aggregation protocol.

G.5 Results for the Task T3: Distribution Alignment

T3: Distribution Alignment. This task tests whether the neighbors selected by a similarity measure are distributionally compatible with the target client. Oracle-JS ranks clients directly from their private label distributions and is therefore used only as an evaluation reference. We report JS divergence between each client and the mixture of its top-6 and top-12 selected neighbors (lower is better), and Kendall rank correlation with the Oracle-JS ordering (higher is better).

Table 20 shows that *SPOS* often preserves distributional compatibility without observing client label distributions or update values. On BERT Dir(0.1), *SPOS* improves all three metrics over cosine, reducing JS@6/JS@12 from .249/.281 to .246/.280 while increasing Kendall from .655 to .663. On Wav2Vec2 Dir(0.3), *SPOS* improves Kendall from .650 to .673 and JS@12 from .200 to .198, while matching cosine on JS@6 at .186. On Dir(0.1), cosine is stronger on all three metrics (.279/.310/.735 versus .284/.312/.708 for *SPOS*). CIFAR-10 Dir(0.3) is also closely matched (.109/.136/.588 for *SPOS* versus .105/.136/.586 for cosine). Performance is more mixed on FMNIST and under pathological partitions, where label supports are highly discrete and several methods produce nearly identical JS values. Overall, these results show that support overlap captures

Table 20: T3: Distribution alignment across non-IID splits. Each entry reports JS@6 / JS@12 / Kendall. Lower JS and higher Kendall are better.

Model / Dataset	**Method**	**Dir**(0.3)	**Dir**(0.1)	**Patho**(20%)	**Patho**(30%)
CNN / FMNIST	Cos.	.124/.142/.598	.193/.232/.552	.317/.452/.336	.316/.451/.315
	Eucl.	.129/.151/.479	.202/.246/.385	.317/.452/.264	.316/.451/.326
	Oracle-JS	.101/.124/1	.142/.203/1	.316/.451/1	.314/.449/1
	SPOS	.124/.143/.582	.191/.222/.560	.317/.451/.341	.315/.450/.316
ResNet18 / CIFAR-10	Cos.	.105/.136/.586	.157/.207/.655	.315/.451/.523	.315/.448/.586
	Eucl.	.139/.149/.278	.241/.267/.585	.315/.451/.522	.314/.448/.560
	Oracle-JS	.101/.125/1	.144/.204/1	.315/.451/1	.313/.448/1
	SPOS	.109/.136/.588	.153/.205/.659	.315/.452/.520	.313/.448/.588
BERT-base / 20News	Cos.	.156/.176/.656	.249/.281/.655	.413/.504/.278	.428/.511/.257
	Eucl.	.165/.183/.636	.252/.288/.613	.417/.509/.263	.427/.512/.228
	Oracle-JS	.152/.175/1	.240/.278/1	.399/.496/1	.415/.502/1
	SPOS	.152/.176/.658	.246/.280/.663	.410/.499/.277	.418/.510/.230
Wav2Vec2 / SpeechCommands	Cos.	.186/.200/.650	.279/.310/.735	.432/.524/.360	.441/.532/.335
	Eucl.	.202/.210/.537	.288/.318/.521	.438/.531/.285	.446/.538/.265
	Oracle-JS	.179/.200/1	.275/.304/1	.425/.518/1	.434/.526/1
	SPOS	.186/.198/.673	.284/.312/.708	.430/.522/.350	.438/.530/.320

Table 21: Bandwidth-derived one-way communication time for dense FP32 relatedness vectors and *SPOS* support messages at $K/M = 10\%$. Times are in seconds.

	100 Mbps		200 Mbps		300 Mbps		400 Mbps	
Model / Dataset	Dense	*SPOS*	Dense	*SPOS*	Dense	*SPOS*	Dense	*SPOS*
CNN / FMNIST	0.135	0.014	0.067	0.007	0.045	0.005	0.034	0.003
ResNet18 / CIFAR-10	3.575	0.358	1.788	0.179	1.192	0.119	0.894	0.089
BERT-base / 20News	35.039	3.504	17.520	1.752	11.680	1.168	8.760	0.876
Wav2Vec2 / SpeechCommands	28.921	2.892	14.460	1.446	9.640	0.964	7.230	0.723

useful distributional structure, although it does not uniformly reproduce the privileged Oracle-JS ordering.

G.6 Bandwidth Sensitivity

We evaluate the communication latency implied by the fixed $K/M = 10\%$ similarity payload under link rates of 100, 200, 300, and 400 Mbps. With 32-bit coordinate IDs, the *SPOS* message contains $32K$ bits compared with $32M$ bits for a dense FP32 relatedness vector; hence its transmission time is $K/M = 0.10$ of the dense baseline at any bandwidth. Table 21 reports the resulting one-way transmission times.

The relative saving is independent of bandwidth because *SPOS* reduces the transmitted similarity-plane payload itself. At the fixed $K/M = 10\%$ budget and using 32-bit coordinate IDs, the similarity-plane representation occupies one tenth the payload of the uncompressed dense FP32 relatedness vector at any link rate. The absolute savings are largest for the larger models: for BERT-base, one-way latency decreases from 35.04 to 3.50 s at 100 Mbps and from 8.76 to 0.88 s at 400 Mbps; for Wav2Vec2, it decreases from 28.92 to 2.89 s and from 7.23 to 0.72 s, respectively. Increasing bandwidth reduces absolute latency for both representations but leaves the $10\times$ similarity-plane advantage unchanged.

G.7 Per-Round Runtime Across Bandwidths

We visualize the per-round runtime composition of ResNet18/CIFAR-10 under Dir(0.3) for bandwidths of 100, 200, 300, and 400 Mbps. Each round is decomposed into local training, *SPOS* support extraction (layer normalization and Top-K), tokenization/MinHash–LSH, and communication. Communication includes both the model update required for aggregation and the *SPOS* index side channel.

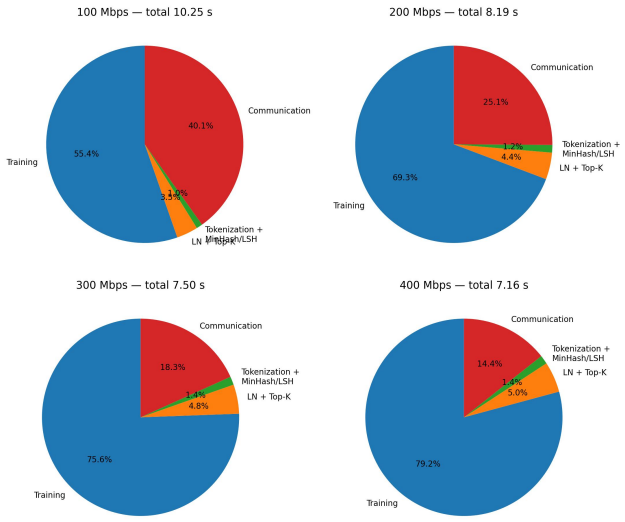


Figure 4: Per-round runtime composition for ResNet18/CIFAR-10 under Dir(0.3) at 100, 200, 300, and 400 Mbps. Communication includes the aggregation update and *SPOS* side channel. Increasing bandwidth reduces the communication share from 40.1% to 14.4%, while the local *SPOS* processing overhead remains small and nearly constant.

Figure 4 shows that communication dominates under lower bandwidth, accounting for 40.1% of the round at 100 Mbps, but drops to 14.4% at 400 Mbps. Accordingly, total round time decreases from 10.25 s to 7.16 s. In contrast, the local *SPOS* overhead remains small and bandwidth-independent: LN+Top-K contributes only 3.5–5.0% of the round, while tokenization/MinHash–LSH contributes about 1.0–1.4%. As bandwidth increases, local training becomes the dominant cost, reaching 79.2% of the round at 400 Mbps.

G.8 Scalability with Client Population

We evaluate how *SPOS* scales with the number of participating clients under Dir(0.3) while keeping the support budget fixed at $K/M = 10\%$. We focus on CNN/FMNIST and ResNet18/CIFAR-10 because these models permit controlled client-count scaling within our available compute and memory budget. We scale CNN/FMNIST to larger populations because its substantially smaller model size makes exhaustive all-pairs evaluation feasible at higher N. For each population size, we compute the complete similarity matrix and compare the induced neighborhoods with the JS-based oracle using R@6, R@12, Spearman, and Kendall. Table 22 reports both relatedness quality and the wall-clock cost of evaluating all $\binom{N}{2}$ client pairs.

Across independently constructed client populations, relatedness metrics vary non-monotonically with N but remain substantially informative over the evaluated range. On CNN/FMNIST, R@12 ranges from 0.689 to 0.783 and Spearman from 0.623 to 0.727 for $N = 50$–400. On ResNet18/CIFAR-10, Spearman remains between 0.714 and 0.759, while R@12 decreases from 0.613 at $N = 50$ to 0.499 at $N = 200$. These experiments therefore support robustness across the tested population sizes, not a monotonic scaling law. Overall, *SPOS* retains useful relatedness structure as the client population increases, with predictable quadratic growth in exact similarity construction cost.

G.9 MinHash Recall and Candidate Pruning

MinHash–LSH is used only as a candidate-retrieval prefilter: exact *SPOS* is subsequently evaluated on the retrieved candidates. We therefore measure retrieval fidelity relative to exact all-pairs *SPOS* together with the resulting candidate-set reduction. Experiments use CNN/FMNIST under Dirichlet $\alpha = 0.3$ with the fixed $K/M = 10\%$ support budget, and $h = 128$ 32-bit MinHash values arranged as $b = 32$ bands of $r = 4$ rows, giving a 512-byte signature per client.

Table 22: Scalability of *SPOS* with increasing numbers of clients. Relatedness quality is measured against the JS-based oracle using R@6, R@12, Spearman, and Kendall. Pairwise time is the measured wall-clock time for computing all client-pair *SPOS* similarities. All runs use five seeds.

Model / Dataset	Clients N	R@6 ↑	R@12 ↑	Spearman ↑	Kendall ↑	All-pairs (s) ↓
CNN / FMNIST	50	0.697	0.783	0.727	0.532	0.18
CNN / FMNIST	100	0.665	0.754	0.653	0.503	0.62
CNN / FMNIST	150	0.632	0.728	0.695	0.512	1.50
CNN / FMNIST	200	0.613	0.716	0.686	0.504	2.48
CNN / FMNIST	250	0.593	0.735	0.691	0.509	4.09
CNN / FMNIST	300	0.618	0.700	0.668	0.471	5.58
CNN / FMNIST	350	0.631	0.689	0.634	0.441	8.76
CNN / FMNIST	400	0.586	0.727	0.623	0.469	9.99
ResNet18 / CIFAR-10	50	0.627	0.613	0.759	0.665	4.75
ResNet18 / CIFAR-10	100	0.660	0.578	0.731	0.594	22.47
ResNet18 / CIFAR-10	150	0.633	0.524	0.714	0.523	49.49
ResNet18 / CIFAR-10	200	0.602	0.499	0.727	0.530	89.25

Table 23: MinHash–LSH candidate pruning and retrieval fidelity relative to exact all-pairs *SPOS*.

N	Mean cand.	Cand./N	R@4	R@8	Sig. build (ms/client)	Query (ms/client)	Measured LSH index memory (MiB)
200	51.88	25.9%	0.9063	0.8606	58.4	0.42	1.2
300	75.14	25.0%	0.9200	0.8625	59.0	0.58	1.8
400	93.80	23.5%	0.9302	0.8816	59.3	0.73	2.4

Table 23 shows that MinHash preserves the exact-*SPOS* neighborhoods well in the evaluated range, with R@4 above 0.90 and R@8 above 0.86. The reported memory includes the LSH hash tables and implementation overhead; it is therefore larger than the raw 128×32-bit (512-byte) MinHash signature size. The mean candidate set is approximately 23–26% of the client population, reducing the number of exact overlap comparisons by roughly $4\times$ relative to exhaustive search. However, the candidate count still grows approximately proportionally with N over this range; we therefore characterize MinHash–LSH as *candidate-pruned*, not as empirically subquadratic. For this experiment, $K/M = 10\%$ gives $K \approx 42.2$k coordinates. With $h = 128$, naive signature construction requires approximately $hK = 5.40$ million hash evaluations per client. We report this cost explicitly and do not extrapolate the current MinHash implementation to multimillion-entry supports without optimized hashing or additional scalability measurements.

H Ablation Study

We use complementary ablations and controlled extensions to characterize the sources and limits of the *SPOS* relatedness signal. We isolate the effects of support size, block normalization, sign removal, and Top-K binarization; test relatedness when label marginals are exactly matched under feature, concept, and quantity/compute shift; and evaluate whether the same index-only representation extends to checkpoints adapted from a shared reference model. These experiments clarify which components drive performance and where the resulting support-based similarity remains informative.

H.1 Representation Decomposition and Layer-Scale Artifacts

Separating normalization, sign, and support. We isolate the three representation choices underlying *SPOS*: removal of update signs, block-wise RMS normalization, and conversion of a dense vector into a binary Top-K support. We use the same CNN/FMNIST Dir(0.3) setting as the main relatedness experiment, with $N = 30$ clients, 10 communication rounds, seven local epochs, $K/M = 10\%$, EMA coefficient $\beta = 0.6$, and the same five-seed evaluation protocol.

For the dense variants, cosine similarity is held fixed while the representation supplied to it is changed. These variants therefore require access to real-valued per-client update information in the similarity

Table 24: Representation decomposition on CNN/FMNIST under Dir(0.3). All variants use the same operating point as the main relatedness experiment: $K/M = 10\%$, $\beta = 0.6$, and five seeds. "Value-bearing input" indicates whether client-to-client similarity requires access to per-client real-valued update information. R@k is higher-is-better and JS@k is lower-is-better.

Representation	Value-bearing input?	R@6↑	R@12↑	JS@6↓	JS@12↓	Spearman↑	Kendall↑
***SPOS* (block-normalized Top-K support)**	**No**	.656	.681	.124	.143	.786	.582
Raw Top-K support	No	.623	.678	.130	.142	.776	.578
Block-normalized absolute-update cosine	Yes	.705	.719	.127	.149	.650	.475
Block-normalized signed-update cosine	Yes	.650	.777	.117	.138	.804	.616
Cosine on absolute update	Yes	.672	.689	.140	.159	.627	.461
Cosine on signed update	Yes	.650	.769	.124	.142	.784	.598

plane. The support variants instead compute similarity from coordinate identities only; neither raw Top-K overlap nor *SPOS* transmits update magnitudes or signs.

Table 24 separates the contribution of each representation choice. Removing update signs changes the dense similarity substantially: cosine on the absolute update has Spearman .627, compared with .784 for the signed update. Block normalization partly recovers this loss for the absolute representation, increasing Spearman from .627 to .650 and R@6 from .672 to .705. For signed dense updates, normalization gives a smaller improvement, from .784 to .804 in Spearman and from .598 to .616 in Kendall. The support comparison isolates the effect most directly relevant to *SPOS*. Without block normalization, raw Top-K overlap reaches Spearman .776 and Kendall .578; applying block normalization before support extraction gives .786 and .582, respectively, while improving R@6 from .623 to .656. Thus, normalization provides a modest benefit within the index-only representation rather than explaining its performance by itself.

The comparison also makes the information tradeoff explicit. Dense cosine variants retain real-valued per-client information and, in the signed case, the complete update direction. *SPOS* instead operates using only the identities of salient coordinates. Its purpose is therefore not to dominate every value-bearing representation numerically, but to retain useful relatedness after update magnitudes and signs have been removed from the similarity channel.

Support allocation across parameter blocks. The previous decomposition measures the effect of normalization on relatedness quality. We next examine the mechanism directly. A global Top-K over raw magnitudes can favor parameter blocks with systematically larger scales, even when those blocks contain only a small fraction of the model parameters. *SPOS* instead normalizes each block before applying a single global Top-K.

For block ℓ, define

$$E_\ell = \frac{\text{support share}_\ell}{\text{parameter share}_\ell},$$

where $E_\ell = 1$ denotes proportional allocation. Table 25 groups Wav2Vec2 parameter blocks for compactness. Raw global Top-K under-selects some transformer and encoder blocks while strongly over-selecting the task head. Block normalization contracts this imbalance toward proportional allocation: raw enrichment ranges from 0.623 to 6.192, whereas layer-normalized *SPOS* reduces the range to 0.933–1.439.

This more balanced allocation is consistent with the downstream CPFL ablation. On Wav2Vec2/SpeechCommands under Dir(0.3), layer-normalized *SPOS* reaches 96.75% test accuracy, compared with 95.81% for Global *SPOS* without block normalization. Together, the representation and allocation ablations show that normalization has a modest effect on relatedness metrics while substantially reducing layer-scale bias in the coordinates selected for the index-only support.

H.2 Sensitivity to the Support Size K

We evaluate the sensitivity of *SPOS* to the support ratio $K/M \in [2.5\%, 30\%]$. Communication is computed using the implementation used throughout the paper, where each selected coordinate

Table 25: Support allocation on Wav2Vec2/SpeechCommands under Dir(0.3). "Raw" denotes unnormalized global Top-K, while "LN" applies the block-wise RMS normalization used by *SPOS*. $E_\ell = 1$ corresponds to support allocation proportional to block size.

Block	Param. share	Raw support	Raw E_ℓ	LN support	LN E_ℓ
Transformer L0–3	31.37%	20.27%	0.646	29.78%	0.949
Transformer L4–7	31.37%	22.83%	0.728	30.48%	0.972
Transformer L8–11	31.37%	51.87%	1.653	32.39%	1.032
Wav2Vec2 encoder	5.22%	3.25%	0.623	6.61%	1.266
Feature projection	0.44%	0.36%	0.824	0.41%	0.933
Task head	0.23%	1.42%	6.192	0.33%	1.439

Table 26: Sensitivity to the support ratio K/M. Each model column reports R@6 / Spearman / Kendall. Higher is better for all metrics.

K/M	CNN / FMNIST	ResNet18 / CIFAR-10	BERT-base / 20News
2.5%	.602/.711/.486	.662/.701/.502	.681/.742/.561
5%	.624/.754/.548	.702/.731/.531	.724/.788/.598
7.5%	.643/.774/.569	.734/.758/.562	.754/.821/.631
10%	.656/.786/.582	.756/.780/.588	.777/.848/.658
12.5%	.661/.791/.587	.764/.789/.596	.786/.862/.674
15%	.664/.794/.590	.768/.795/.602	.791/.873/.683
17.5%	.663/.795/.591	.770/.798/.606	.794/.879/.688
20%	.661/.794/.594	.771/.800/.608	.799/.878/.687
22.5%	.659/.792/.588	.770/.799/.607	.790/.874/.683
25%	.656/.789/.585	.768/.797/.605	.786/.869/.678
27.5%	.652/.785/.581	.765/.794/.602	.781/.863/.672
30%	.648/.781/.577	.762/.790/.598	.776/.857/.666

is transmitted as a 32-bit integer ID; hence the payload grows linearly with K. We report R@6, Spearman, and Kendall τ_b, with the latter two measuring agreement with the Oracle-JS client ranking.

Table 26 shows a consistent diminishing-return pattern. Most of the relatedness quality is recovered by $K/M = 10\%$, while increasing the support further yields only modest, architecture-dependent gains. On CNN/FMNIST, Spearman/Kendall are $.786/.582$ at 10% and peak at only $.795/.594$; on ResNet18/CIFAR-10, they increase from $.780/.588$ to $.800/.608$ around 20%. BERT-base/20News shows the largest benefit from a denser support, increasing from $.848/.658$ at 10% to $.879/.688$ at 17.5%, after which both metrics decline. R@6 follows the same pattern, with only small gains beyond the 10% operating point.

This saturation is expected from Top-K selection. The highest-salience coordinates capture the strongest and relatively stable client-specific update structure, whereas increasing K progressively admits lower-margin coordinates that are more sensitive to local and temporal variation. Moreover, denser supports increase incidental overlap between otherwise unrelated clients. Consequently, communication grows linearly with K, whereas ranking quality saturates and can eventually decrease. At $K/M = 10\%$, *SPOS* already retains at least 96.5% of the peak Spearman and 95.6% of the peak Kendall observed across these sweeps. We therefore use $K/M = 10\%$ as a common operating point for all models: it captures most of the attainable relatedness quality, uses one tenth the similarity-plane payload of an uncompressed FP32 vector under the 32-bit coordinate-ID representation, and avoids architecture-specific tuning.

H.3 Checkpoint-Relative Model Relatedness

We next test whether the signal extends beyond recovery of dataset statistics. Starting from one shared reference checkpoint, we adapt 15 same-architecture models on disjoint data from five domains, with three replicas per domain. We compare parameter similarities against two behavior-based oracles that do not use label histograms: mean predictive-distribution agreement on held-out multi-domain probes and symmetrized cross-domain transfer accuracy. All models in the primary experiment share

Table 27: Checkpoint-relative relatedness from a shared reference model. Entries are means over five seeds. R@3 measures functional-oracle neighbor retrieval and AP measures recovery of checkpoints adapted to the same domain. All models share architecture, parameter ordering, and reference checkpoint.

Behavioral oracle	Representation	Spearman↑	R@3↑	Same-domain AP↑
Predictive agreement	Signed dense cosine	**.740**	**.733**	**1.000**
	Absolute dense cosine	.634	.711	**1.000**
	Block-normalized signed dense	.730	.716	.993
	Block-normalized absolute dense	.657	.716	**1.000**
	Raw Top-K support	.525	.709	.978
	SPOS	.539	.716	.991
Cross-domain transfer	Signed dense cosine	**.670**	.720	**1.000**
	Absolute dense cosine	.594	.702	**1.000**
	Block-normalized signed dense	.662	**.724**	.993
	Block-normalized absolute dense	.613	.707	**1.000**
	Raw Top-K support	.484	.687	.971
	SPOS	.508	.702	.991

architecture, parameter ordering, and the reference initialization, so their parameter indices have common semantics.

Against predictive agreement, *SPOS* reaches R@3 $= .716$ and same-domain AP $= .991$; against cross-domain transfer, it reaches R@3 $= .702$ and AP $= .991$. Its full-ranking Spearman correlations are $.539$ and $.508$, below signed dense cosine at $.740$ and $.670$. Using unrounded seed-level scores, the paired Spearman differences between *SPOS* and signed dense cosine are $-.201$ with 95% CI $[-.327, -.076]$ for predictive agreement and $-.163$ with CI $[-.262, -.063]$ for transfer. From the rounded table entries, the corresponding R@3 gaps are $-.017$ and $-.018$, respectively; the paired seed-level intervals include zero. Thus, index-only support preserves top-neighbor and domain retrieval better than it preserves the full behavioral ordering.

This claim is coordinate-relative. A function-preserving hidden-unit permutation reduces raw *SPOS* from 1.000 after known alignment to $.114$ before alignment. Independent initialization produces functional similarity $.957$ but raw *SPOS* $.125$, and parameter-index similarity is undefined across different architectures. These controls are reported in Appx. G.3.

H.4 Relatedness Beyond Label Skew

The preceding experiments define relatedness primarily through client label distributions. We therefore construct three controls in which all clients have identical label marginals: feature shift is induced by preregistered image transformations, concept shift by agreement between label permutations, and quantity/compute imbalance by independently varying local sample counts and local epochs while retaining a feature-domain oracle. The maximum observed difference between client label marginals is 4.16×10^{-17}, and none of these oracles depends on client label histograms.

Table 28 shows that *SPOS* retains relatedness signal even when label marginals are uninformative. Under feature shift, absolute dense cosine has the highest Spearman ($.566$), while *SPOS* achieves the highest R@5 ($.757$) and ties the best group AP ($.817$). Under concept shift, signed dense cosine is strongest, but *SPOS* remains close, reaching Spearman $= .781$, R@5 $= .973$, and group AP $= .985$ versus $.791$, $.988$, and $.993$ for signed dense cosine. Under quantity/compute imbalance, *SPOS* attains the highest R@5 ($.597$) and group AP ($.738$), while signed dense cosine has the highest full-ranking agreement ($.628$ versus $.617$). These results show that *SPOS* captures relatedness beyond label-distribution similarity and remains competitive with dense value-based representations across feature, concept, and quantity/compute heterogeneity, without implying uniform superiority over dense signed geometry.

Table 28: Relatedness with exactly equal label marginals. Entries are means over five seeds. Spearman measures full oracle-ranking agreement, R@5 measures oracle-neighbor retrieval, and AP measures recovery of clients in the same oracle group. Highest values within each scenario and metric are bold.

Scenario	Representation	Spearman↑	R@5↑	Group AP↑
Feature shift	Signed dense cosine	.523	.741	.801
	Absolute dense cosine	**.566**	.752	**.817**
	Block-normalized absolute dense	.553	.745	.808
	Raw Top-K support	.547	.755	.816
	SPOS	.556	**.757**	**.817**
Concept shift	Signed dense cosine	**.791**	**.988**	**.993**
	Absolute dense cosine	.769	.965	.975
	Block-normalized absolute dense	.743	.937	.955
	Raw Top-K support	.759	.937	.955
	SPOS	.781	.973	.985
Quantity/compute imbalance	Signed dense cosine	**.628**	.564	.715
	Absolute dense cosine	.498	.541	.677
	Block-normalized absolute dense	.570	.568	.717
	Raw Top-K support	.505	.587	.721
	SPOS	.617	**.597**	**.738**

Table 29: Leakage from raw and round-keyed *SPOS* messages. Lower W1 indicates stronger histogram inference; higher AUC* and MRR indicate stronger attribute inference and cross-round linkage. Values are means over five seeds. Tokenization is evaluated under the non-colluding unknown-key condition.

	W1 ↓			AUC* ↑			MRR ↑		
Dataset	Raw	Tok.	Uninf.	Raw	Tok.	Chance	Raw	Tok.	Chance
FMNIST	.187	.185	.190	.556	.542	.500	.940	.138	.133
CIFAR-10	.178	.175	.176	.644	.564	.500	.880	.131	.133
20News	.129	.128	.128	.633	.553	.500	.800	.139	.133
SpeechCommands	.089	.088	.089	.604	.502	.500	.840	.134	.133

I Security Evaluation

Threat model and attacks. We evaluate information available from the *SPOS* support message itself. The coordinator knows the public model, training procedure, support size, and within-round similarity graph, but not update magnitudes or signs. In the round-keyed condition, the coordinator does not know the round key and no participating client discloses it; compromise of one participant reveals the mapping for that round. Thus, tokenization is evaluated under a non-collusion assumption.

We consider three attacks under $\mathrm{Dir}(0.3)$. Histogram inference predicts a client's normalized label histogram and is measured by W1 (lower is stronger); the uninformed baseline predicts the training-client prior. Attribute inference uses AUC* (higher is stronger; 0.5 is chance). For cross-round linkage, sender metadata is removed and a round-t report is ranked against the $N = 30$ reports from round $t + 1$ using message similarity. Exactly one candidate corresponds to the same client, so random ranking gives $\mathbb{E}[\mathrm{MRR}] = H_{30}/30 = 0.133$. Independently keyed permutations are used across consecutive rounds in the tokenized condition.

Interpretation. Histogram reconstruction remains close to the uninformed reference, whereas raw coordinate identities expose measurable attribute information (AUC$^* = 0.556$–0.644). Raw supports are also strongly linkable across rounds (MRR $= 0.800$–0.940), consistent with their observed temporal persistence. Independently round-keyed tokenization removes direct persistent coordinate labels and reduces this direct-overlap linkage diagnostic to MRR $= 0.131$–0.139, near the random-ranking reference 0.133.

These results do not establish anonymity or formal privacy. Tokenization preserves the complete within-round *SPOS* matrix and does not protect against attacks exploiting token frequencies, graph structure, or a disclosed round key.

I.1 Reconstruction Stress Test from Dense Gradients and *SPOS* Messages

Purpose and scope. The previous attacks target aggregate client properties rather than individual training examples. We therefore add an example-level reconstruction stress test. This is intentionally a strong leakage setting: a single-step, small-batch inversion attack, where gradient reconstruction is known to be most effective. It is *not* intended to simulate the full E-epoch client update. Across all message types, we keep the public checkpoint, target example, batch size, initialization distribution, optimization budget, and restart rule fixed. Primary results use batch size one with the true label given to all attackers, isolating input leakage from label inference. Restarts are selected only by attack objective, never by the ground-truth image metrics reported in Table 30.

Dense unprotected gradient. Let $g = \nabla_\theta \ell(f_\theta(x), y)$ be the complete observed per-example or mini-batch gradient. Following gradient-inversion attacks, the adversary optimizes a dummy input $\hat{x}$ (and optionally a dummy label) to match this dense signal. This is the unprotected value-sharing baseline and represents the strongest reconstruction surface considered here.

Raw-*SPOS* support-constrained inversion. Dense inversion is not directly applicable to *SPOS*, since the attacker does not observe gradient values, signs, or magnitudes. Instead, the attacker sees only the layer-normalized Top-K support $I = \mathrm{TopK}(z(g))$ and optimizes a dummy input so that its induced salience reproduces this support. Concretely, we use a pairwise ranking surrogate that pushes sampled positive coordinates in I above sampled negatives outside I, without revealing any unreported values. We therefore refer to this as a *support-constrained inversion attack*, rather than a dense gradient attack.

Tokenized *SPOS* under unknown-key and key-disclosure conditions. Under the non-colluding unknown-key assumption, the coordinator observes $\{\pi_{\mathsf{rk}_t}(k) : k \in I_i^{(t)}\}$ but cannot directly map the tokens to model coordinates, so it cannot instantiate the raw coordinate-level support objective. In the unknown-key condition, the evaluated attacker assigns a random size-K coordinate preimage to the tokens. In the key-disclosure condition, the mapping is revealed and the attacker runs the same support-constrained attack as for raw *SPOS*. Because every participating client possesses or can evaluate the round mapping, this condition models disclosure by one compromised or colluding participant, not merely an abstract oracle. We also report an uninformed prior-only baseline.

Metrics. We report MSE, PSNR, and SSIM. Lower MSE and higher PSNR/SSIM indicate stronger reconstruction and therefore greater leakage. Observation fit is gradient cosine for the dense condition and final support overlap $|I \cap \widehat{I}|/K$ for support-based conditions. For the uninformed prior, observation fit is not defined. Quantitative reconstruction results are reported in Table 30.

Dense-gradient exposure enables substantially stronger reconstruction than support-only communication on both datasets. On FMNIST, the dense-gradient attack reaches SSIM 0.798 and PSNR 31.38 dB, whereas raw *SPOS* drops to SSIM 0.342 and PSNR 13.40 dB. Importantly, this degradation occurs even though the raw-*SPOS* attacker recovers 87.7% of the observed support. Thus, high support agreement does not imply recovery of the underlying example. On CIFAR-10, the uninformed prior reaches SSIM 0.007, compared with 0.011 under unknown-key tokenization and 0.026 from raw *SPOS*.

Under the non-colluding unknown-key condition, tokenization reduces the effectiveness of the evaluated coordinate-level reconstruction attack. On FMNIST, SSIM falls to 0.163 and observation fit to 0.120; on CIFAR-10, SSIM falls to 0.011 and observation fit to 0.150. These results show that the evaluated attack depends on mapping support tokens back to model coordinates. Once the round mapping is disclosed, reconstruction returns to raw-SPOS-like behavior: support fit is 0.877 versus 0.877 on FMNIST and 0.474 versus 0.481 on CIFAR-10. The experiment therefore demonstrates conditional protection against coordinate-semantic inversion, together with complete loss of that protection after one round-key disclosure. It does not evaluate attacks that use the preserved within-round similarity graph without recovering coordinate identities.

Table 30: Example-level reconstruction stress test on one held-out target per dataset. The attacker is given the true label and differs only in the observed client message. Dense exposes the complete gradient; raw *SPOS* exposes layer-normalized Top-K coordinate identities; tokenized *SPOS* hides their coordinate mapping using a fresh secret round key. The oracle-key row reveals this mapping as a key-compromise control. Lower MSE and higher PSNR/SSIM indicate stronger reconstruction. Observation fit is gradient cosine for Dense and support overlap for support-based attacks.

Dataset	Observed message	MSE ↓	PSNR ↑	SSIM ↑	Obs. fit ↑
FMNIST/CNN	Dense gradient	7.27×10^{-4}	31.38	.798	.986
	Raw *SPOS*	.0457	13.40	.342	.877
	Tokenized *SPOS* (unknown key)	.1572	8.03	.163	.120
	Tokenized *SPOS* (key disclosed)	.0481	13.17	.323	.877
	Uninformed prior	.1950	7.10	.021	–
CIFAR-10/ResNet18	Dense gradient	.0245	16.10	.361	.795
	Raw *SPOS*	.1158	9.36	.026	.481
	Tokenized *SPOS* (unknown key)	.1304	8.85	.011	.150
	Tokenized *SPOS* (key disclosed)	.1140	9.43	.019	.474
	Uninformed prior	.1447	8.39	.007	–

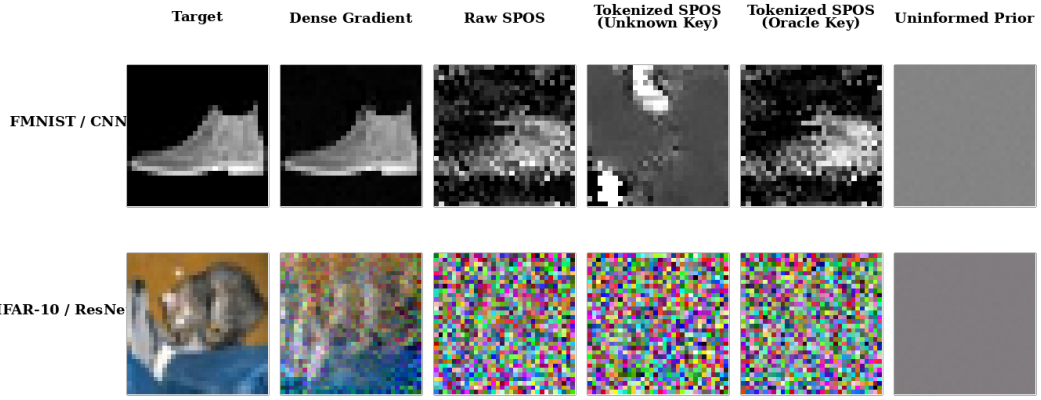


Figure 5: Representative reconstruction attacks on FMNIST/CNN and CIFAR-10/ResNet18. Detailed reconstruction methodology and quantitative results are reported in Appx. I.1.

J LIMITATIONS

SPOS requires a shared parameter coordinate system: parameter indices must have the same semantic meaning across compared models. It is therefore not permutation invariant and does not directly apply across heterogeneous architectures. In T1, cluster-specific models can diverge in parameter values; we empirically compare their coordinate supports but do not claim that the shared-reference theory covers this reference drift.

The method communicates support rather than full signed update geometry. Consequently, it can discard directional information required by tasks that depend on precise update alignment. At the fixed $K/M = 10\%$ operating point, the support can also remain large for modern pretrained models; our $10\times$ communication statement refers specifically to 32-bit IDs versus an uncompressed dense FP32 affinity vector.

Finally, index-only communication is not a privacy mechanism. Support identities, coordinate frequencies, and the induced similarity graph can reveal information. Round-keyed tokenization hides direct coordinate labels only under a non-collusion assumption and does not hide the within-round incidence structure. Our baseline study is also representation-focused rather than an exhaustive comparison with all personalized-FL, data-summary, sketch-based, or random-projection alternatives.