

BAF-Defense: A Bi-Branch Autoencoder Fusion Defense Against Label-Flipping Attacks in Federated Learning

Yunhao Wang*, Ranesh Naha†, Aniket Mahanti*, Diksha Goel‡

*University of Auckland, New Zealand; †Queensland University of Technology, Australia; ‡CSIRO’s Data61, Australia

*nway818@aucklanduni.ac.nz, a.mahanti@auckland.ac.nz; †ranesh.naha@qut.edu.au; ‡diksha.goel@csiro.au

Abstract—Federated learning (FL) lets mobile, edge, and Internet of Things (IoT) devices in 5G/6G and vehicular networks train a shared model without sharing raw data. This decentralization opens an attack surface: a poisoned minority of clients can corrupt the global model through label-flipping attacks. Clustering defenses such as LFighter detect these clients from their last-layer gradient features, but they assume malicious and benign updates separate cleanly. Under non-independent and identically distributed (non-IID) data, the norm at the edge, that assumption fails: the two populations overlap, a single hard cluster boundary turns brittle, and its binary verdict discards the confidence information that matters most for clients near the margin. We present *BAF-Defense* (Bi-Branch Autoencoder Fusion), a server-side defense that recovers this discarded signal instead of replacing the clustering rule. It augments LFighter’s binary cluster decision with a continuous per-round trust score from an autoencoder (AE) trained on the same gradient features, then fuses the two asymmetrically: the cluster decision remains dominant, while the trust score arbitrates the borderline clients a hard boundary misjudges. The defense is purely server-side and scales linearly in the number of clients per round, adding no client computation, no extra communication, and no auxiliary data. On CIFAR-10 with 40% malicious clients, BAF-Defense holds the attack success rate (ASR) to 8.6%, against 21.1% for LFighter and 26.0% for FedAvg, while recovering 9.2 and 11.3 accuracy points over LFighter and FedAvg, respectively. The advantage widens as the attacker fraction grows and closes to parity on clean data, and the accuracy gains extend to MNIST and the medical-imaging MedMNIST benchmark, reflecting the cross-silo medical FL setting we target.

Index Terms—Federated learning, label-flipping attacks, poisoning defense, robust aggregation, edge intelligence, security of machine learning.

I. INTRODUCTION

Federated learning (FL) trains across many clients that keep their data local, exchanging only model updates with a central server [1], [2]. This arrangement fits mobile and edge deployments where raw data cannot leave the device for privacy, regulatory, or bandwidth reasons, spanning healthcare [3], connected vehicles [4], and industrial IoT systems [5]. Yet the server does not control the devices that perform training. This opens an attack surface absent from centralized learning: a client can submit a crafted update in place of an honest one.

Poisoning is the most studied such threat: a small malicious set can plant backdoors or drive large accuracy drops by tampering with local training [6], [7], [8]. Label-flipping attacks (LFAs) are a stealthy variant that alters only labels

and leaves input features intact, letting poisoned updates evade defenses that look for statistical outliers. With 10–40% of clients compromised, single-pair flipping has been shown to push ASR above 25% and cut accuracy by more than 15 points [9].

Detecting label flipping under non-IID data, the norm in mobile and edge deployments [2], [5], remains difficult. LFighter [9] clusters selected last-layer gradient features to identify malicious clients, and this works when the poisoned gradients form a distinct group. Three properties of label flipping erode that assumption: (i) benign clients with different class priors produce dispersed gradients even when no attacker is present; (ii) flipping perturbs the target class while leaving inputs intact, so poisoned gradients sit closer to the benign mass than a Byzantine outlier would; and (iii) a hard cluster label collapses confidence, treating a client just inside the boundary as a clear outlier. A single k -means boundary is therefore sensitive to gradient noise, to high attacker ratios, and to nonlinear malicious structure. Our own reproduction under the LFighter protocol confirms this: LFighter’s ASR rises by a factor of 3.2 as the attacker ratio grows from 10% to 40% (from 6.5% to 21.1%), its source-class stability degrades non-monotonically as the attacker fraction increases, and clustering alone misses nonlinear anomalies that a reconstruction detector catches (an AE-only variant reaches 15.2% ASR at 40% attackers, against LFighter’s 21.1%; Section IV).

The participants that most need protection can least spare computation or bandwidth, so a practical defense should run entirely at the server, add nothing to client computation or communication, and avoid touching raw data. Many existing defenses instead lean on separable updates, round-to-round history, or distributional assumptions [8], [10], [11] that weaken under non-IID data. Our observation is that a compact nonlinear representation can separate two sources of variation that raw gradients conflate, namely benign heterogeneity and malicious drift. An autoencoder is well suited to this role because its reconstruction error provides a continuous anomaly signal, complementing rather than duplicating the cluster assignment [12]. Neither signal is sufficient on its own: a reconstruction score flags general anomalies but cannot identify *which* classes are under attack, while the cluster assignment identifies the attacked classes but discards all confidence near its boundary. Their fusion makes the decision

both class-targeted and confidence-aware.

Our contribution. We propose *BAF-Defense*, a server-side defense that turns LFighter’s hard accept/reject decision into a calibrated one. Its contributions are threefold:

- **Failure-mode analysis.** We characterize why raw-gradient clustering becomes unreliable under heterogeneity: benign and malicious drift overlap near the decision boundary, where a binary label discards the confidence information needed to separate them (Section III), with supporting feature-space evidence in Section IV.
- **Calibrated trust signal and asymmetric fusion.** We fuse a per-round AE reconstruction score, computed over LFighter’s own gradient features, with the binary cluster score, weighting the two so that the label-flipping signature stays dominant while the continuous score resolves borderline clients. The defense runs entirely server-side and adds no client cost.
- **Deployment-driven evaluation.** We motivate the defense from the constraints of wireless and mobile FL, evaluate it on CIFAR-10, MNIST, and MedMNIST under LFighter’s protocol [9], ablate each branch, and delimit its scope against adaptive attackers, secure aggregation, and differential privacy.

II. RELATED WORK

We group defenses by the signal they use to separate benign from malicious updates and position BAF-Defense against the most closely related defenses. The focus is label flipping, but we include broader robust-aggregation methods because they recur as baselines.

Aggregation-based. FedAvg (Federated Averaging) [1] averages all updates and so fails under poisoning, since it treats every client equally. Krum [13] and Multi-Krum instead select the geometrically most central updates, while Trimmed Mean [11] and coordinate-wise Median discard extreme coordinates. These assumptions are fragile under label flipping. Poisoned clients train on genuine inputs, so they need not be coordinate-wise or Euclidean outliers. Benign non-IID clients, by contrast, can themselves appear far apart in update space.

Similarity-based. FoolsGold [14] uses cosine similarity to penalize correlated Sybil updates. It avoids auxiliary data but is $O(n^2)$ and misclassifies benign clients with similar distributions, a problem that is acute under non-IID data [5].

Feature- and trust-based. Tolpegin et al. [7] project last-layer gradients with principal component analysis (PCA) and flag clients that separate from the benign majority, though a determined adversary can explicitly optimize its updates against such robust rules [15]. Reputation schemes such as CONTRA [10] accumulate trust over rounds. These approaches motivate richer trust signals, but they typically depend on round-to-round history or on assumptions about attack consistency. BAF-Defense instead computes a per-round reconstruction score directly from LFighter’s gradient features, treating reconstruction error as a continuous anomaly measure [12].

TABLE I
COMPARISON OF REPRESENTATIVE FL DEFENSES AGAINST POISONING ATTACKS.

Method	Signal	NIID	Adapt.	Key Limitation
FedAvg	Mean update	×	×	No filtering
Median/TM	Coordinates	△	×	Ignores correlations
Multi-Krum	Distances	△	×	Needs compact benign cluster
FoolsGold	Similarity	△	×	Focused on Sybil attacks
FLAME	HDBSCAN + clip	△	×	Designed for backdoor defense
LFighter	Gradient cluster	✓	×	Hard cluster decision boundary
BAF-Defense	Cluster + AE	✓	×	Non-adaptive attacker setting

✓: supported; △: partial; ×: not supported. “Adapt.” indicates robustness against defense-aware adversaries.

Hybrid. FLAME [16] combines density-based clustering (HDBSCAN) with weight clipping and noise injection for backdoor defense. LFighter [9] applies k -means to last-layer gradient features for non-IID label flipping and is our closest baseline, but its decision remains a raw-gradient cluster assignment.

Research gap. These defenses share a limitation under heterogeneity: they return a hard accept/reject decision, and the statistical and similarity signals behind it weaken precisely when poisoned updates are not outliers and benign non-IID updates overlap with attacks. BAF-Defense addresses these by learning a nonlinear AE representation of LFighter’s gradient features and fusing a continuous reconstruction score with the cluster score. Our contribution is the AE’s placement as a server-side companion to LFighter’s source/target detector. Table I summarizes how these defenses differ in detection signal, non-IID robustness, and resistance to adaptive attackers.

III. METHODOLOGY

A. Deployment Setting

BAF-Defense targets the wireless and mobile FL deployments named in the introduction, where label flipping is plausible and hard to contain at the client. A compromised or systematically mislabeling hospital site can flip diagnostic labels while the coordinating server, by design, never sees raw data [3]. An adversary controlling vehicles or roadside units can flip labels at the source, and the server aggregates over bandwidth- and energy-constrained links that rule out any client-side addition [4]. Industrial IoT adds strongly non-IID data across sensors and locations, on devices that cannot host mitigation logic [5].

These scenarios impose three design constraints: the defense runs entirely at the server, adds no client computation or communication, and accesses no raw data or labels. BAF-Defense satisfies all three by scoring only the model updates the server already receives, which distinguishes it from history-based, similarity-based, and distribution-based defenses that relax at least one of them [8], [10], [11].

B. Overview

BAF-Defense extends LFighter [9] with an autoencoder (AE) branch parallel to the clustering branch. The protocol is unchanged: each round the server collects updates, extracts last-layer gradient features following LFighter, and replaces the original FILTER step with FILTERENHANCE (Algorithm 1). Figure 1 shows the pipeline.

Threat model. We consider a cross-device or cross-silo system with n clients, up to $f < n/2$ malicious (honest majority). Malicious clients relabel local samples from a source to a target class, keep features unchanged, train normally, and submit updates; the server cannot access client data or verify labels. Our main evaluation uses a non-adaptive attacker who knows the FL protocol but not the defense parameters, the regime used by many first-stage defenses [9], [16]. We separately discuss an adaptive attacker who knows the AE branch. BAF-Defense targets non-adaptive and weakly adaptive label flipping; it is not designed for arbitrary Byzantine, backdoor, or model-replacement attacks.

Privacy scope. BAF-Defense needs no raw data, labels, or auxiliary validation set; it operates only on updates already visible to a FedAvg server, preserving data locality but providing no cryptographic privacy. It cannot be applied under secure aggregation that hides individual updates, and differential-privacy noise may reduce detection power.

System architecture. Each round t , after receiving updates $\{\Delta w_k^t\}_{k=1}^n$, the server extracts last-layer (class-logit) gradient features and, following LFighter, focuses on the two classes with highest aggregated magnitude, typically the source and target of a flip.

We write $\hat{g}_k^t$ for these features after ℓ_2 normalization; they are the shared input to both branches.

C. Autoencoder-Based Scoring and Filtering

The LFighter branch applies k -means with $k = 2$ to $\{\hat{g}_k^t\}$ and assigns a binary cluster score $\text{cluster_score}_k \in \{0, 1\}$ through the size and density heuristics of the original method [9]. Our extensions are the AE branch and the score fusion described next.

Autoencoder branch. We train a symmetric multi-layer perceptron (MLP) autoencoder (E_ϕ, D_ψ) of shape $d \rightarrow 128 \rightarrow 32 \rightarrow 128 \rightarrow d$, with rectified-linear (ReLU) activations, batch normalization, and dropout of 0.1, optimized by Adam (learning rate 10^{-3} , weight decay 10^{-5}) for 50 epochs each round under a mean-squared-error (MSE) reconstruction loss. Here $d = 2C$ for a C -class problem: LFighter forms the feature by concatenating the C -dimensional last-layer logit gradients of the two selected classes, so on CIFAR-10 ($C = 10$) this gives $d = 20$. The 128-dimensional over-complete layer followed by the 32-dimensional bottleneck retains the dominant variation while filtering noise [17], [18]. For each client we compute a per-feature reconstruction error and map it to a bounded score:

$$e_k = \frac{1}{d} \|\hat{g}_k - D_\psi(E_\phi(\hat{g}_k))\|_2^2 \quad (1)$$

$$\text{recon_score}_k = 1 - \frac{e_k - e_{\min}}{e_{\max} - e_{\min} + \epsilon} \quad (2)$$

Rationale. Let $\mathcal{M}_b$ denote the low-dimensional manifold of benign gradient features. A compact AE trained each round approximates the dominant structure of this manifold whenever benign clients form the majority. Benign non-IID updates may lie far apart in raw feature space yet still reconstruct well, provided they share the same source and target label semantics. Flipping reverses the sign and inflates the magnitude of the source and target gradients, pushing the update away from the benign manifold. The reconstruction error e_k thus measures how far a client departs from the dominant benign structure, which is why the AE branch reduces the noise sensitivity of a hard cluster boundary.

Score fusion and filtering. The trust score combines the binary cluster score and the continuous AE score:

$$s_k = \alpha \cdot \text{cluster_score}_k + (1 - \alpha) \cdot \text{recon_score}_k \quad (3)$$

Clients whose s_k falls below the q -quantile are excluded from aggregation. With $\alpha = 0.8$ the structure of this rule is worth making explicit. Because the cluster score is binary, the fused scores fall into two non-overlapping bands: clients the cluster branch deems benign occupy $[0.8, 1.0]$, and those it deems malicious occupy $[0, 0.2]$. The reconstruction score therefore re-orders clients *within* a band rather than across it, and it alters a final decision only when the q -quantile cut falls inside a band. This happens precisely in the high-attacker-ratio regime, where the cluster size disagrees with the true attacker fraction and a hard boundary is least reliable. This explains why the AE contributes most as the attacker fraction grows. We set q to the true attacker ratio for controlled evaluation, the same convention applied to LFighter, so the comparison stays controlled even though the threshold favors both. At deployment, q can be replaced by the LFighter cluster-size estimate the system already produces, and we leave a dedicated q -sensitivity study to future work. The asymmetry is intentional. The cluster score anchors the defense to the known label-flipping signature, while the reconstruction score arbitrates the boundary clients the cluster branch is least sure about (Algorithm 1).

The additional cost is incurred entirely at the server and scales as $O(n)$ in the number of clients per round. For CIFAR-10, where $d = 20$, the AE holds roughly 13K parameters, so per-round scoring is negligible beside client training, and the client protocol, payload, and local computation are all unchanged.

IV. EVALUATION

A. Experimental Setup

We evaluate BAF-Defense against targeted label flipping with 100 clients coordinated by a single server. Malicious clients flip one source class to a target class, while benign clients train normally. Each client performs 5 local epochs of stochastic gradient descent (SGD, momentum 0.9) per round across 200 rounds, and we sweep the attacker fraction from 10% to 40%. Non-IID partitions are drawn from a Dirichlet split with concentration $\alpha_{\text{Dir}} = 1.0$ [9], the moderate heterogeneity typical of cross-device and cross-silo settings [2],

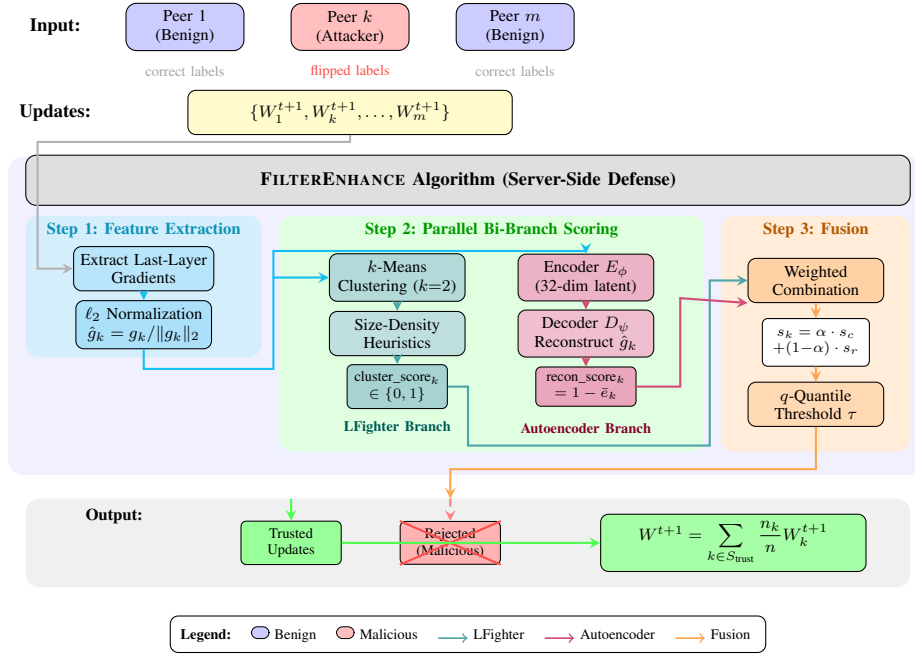


Fig. 1. BAF-Defense framework overview. The server receives gradient updates from FL clients (including potential attackers with flipped labels). **Step 1** extracts and normalizes last-layer gradient features. **Step 2** processes features through two parallel branches: the LFighter branch applies k -means clustering with size-density heuristics to produce binary cluster scores, while the Autoencoder branch computes reconstruction errors converted to continuous trust scores. **Step 3** fuses both scores using weighted combination and applies quantile-based thresholding. Only trusted updates are aggregated into the new global model W^{t+1} .

Algorithm 1 FILTERENHANCE: AE-based scoring and filtering

Require: Gradient features $\{\hat{g}_k\}_{k \in S}$, trained AE (E_ϕ, D_ψ) , α, q
Ensure: *bad_peers*
1: // LFighter branch
2: $(C_1, C_2) \leftarrow k\text{-means}(\{\hat{g}_k\}, k=2)$
3: Assign $\text{cluster_score}_k \in \{0, 1\}$ via [9]
4: // AE branch
5: **for** each $k \in S$ **do**
6: $e_k \leftarrow \frac{1}{d} \|\hat{g}_k - D_\psi(E_\phi(\hat{g}_k))\|_2^2$
7: **end for**
8: $\text{recon_score}_k \leftarrow 1 - (e_k - e_{\min}) / (e_{\max} - e_{\min} + \varepsilon)$
9: // Fusion and filtering
10: $s_k \leftarrow \alpha \cdot \text{cluster_score}_k + (1-\alpha) \cdot \text{recon_score}_k$
11: $\text{bad_peers} \leftarrow \{k \in S \mid s_k \leq q\text{-quantile}(\{s_k\})\}$
12: **return** *bad_peers*

[5]. We evaluate on three image-classification benchmarks: CIFAR-10 [19] and MNIST [20], the standard vision testbeds, and MedMNIST [21], a standardized collection of biomedical images whose Path, Blood, and OrganA subsets reflect the cross-silo medical FL deployments motivated in Section III.

We report four metrics. ASR is the fraction of source-class test samples misclassified into the attacker’s target class. COV_Src_Acc is the coefficient of variation (standard deviation over mean) of source-class accuracy over the final 50 rounds, so a lower value signals a more stable defense. We also

report F1-score and overall accuracy. We use two baselines: FedAvg, the undefended aggregator, which shows how severe the attack is with no defense; and LFighter, the state-of-the-art label-flipping defense that BAF-Defense extends [9]. Because BAF-Defense and LFighter share the same clustering stage, any improvement over LFighter isolates the contribution of the added trust signal. For completeness we additionally ran six families of robust aggregators (Median, Trimmed Mean, Multi-Krum, FoolsGold, Tolpegin, and FLAME) under the same protocol with default configurations, positioned qualitatively in Section II and Table I. Table II summarizes the settings. The implementation runs on a single NVIDIA A100 (80 GB) with Python 3.11 and PyTorch 2.x. Because each run is computationally expensive (200 rounds $\times$ 100 clients), we report single runs and rely on COV_Src_Acc to characterize stability.

B. Main Results

On CIFAR-10 with a ResNet-18 backbone, BAF-Defense improves robustness to targeted poisoning while preserving utility, as Figure 2 shows across all four metrics.

Attack success rate. BAF-Defense records the lowest ASR at every attacker fraction. At 40% attackers it holds the ASR to 8.6%, compared with 21.1% for LFighter and 26.0% for FedAvg, a relative reduction of 59% over the stronger baseline. The advantage grows with the attacker ratio, which agrees with the band analysis of Section III: the continuous score contributes most where the hard boundary is least stable.

TABLE II
SUMMARY OF EXPERIMENTAL SETUP.

Aspect	Details
Datasets	CIFAR-10 (primary), MNIST, MedMNIST subsets (Path/Blood/OrganA).
Model	ResNet-18 for the CIFAR-10 experiments; a lightweight convolutional neural network (CNN) for MNIST and the MedMNIST subsets. Architecture and initialization are shared across compared methods within each dataset.
Attackers	10/20/30/40%; single-pair source→target flip (3 → 5 on CIFAR-10), applied every round.
Local training	5 epochs, SGD (mom. 0.9); lr searched over $\{10^{-4}, 10^{-3}, 10^{-2}, 0.05, 0.1\}$.
Partition	Non-IID Dirichlet, $\alpha_{\text{Dir}}=1.0$.
BAF-Defense	$\alpha=0.8$; q = true attacker ratio (LFighter cluster-size proxy at deployment).

Stability. The defense holds COV_Src_Acc between 14.8% and 15.9% across all ratios, a relative reduction of 18.9% to 32.7% over LFighter (18.5% to 22.0%). FedAvg’s coefficient of variation also drifts downward, from 19.3% to 17.4%, but for an undesirable reason: a strong attack drives source-class accuracy to a uniformly low level, so the variation shrinks around a poor mean. BAF-Defense likewise declines, from 15.9% to 14.8%, but here the AE yields steadily more consistent representations as the attacker fraction grows, so the defense keeps both a lower coefficient of variation and a far higher absolute source-class accuracy than FedAvg at every ratio. LFighter, by contrast, behaves non-monotonically across the sweep (21.4%, 21.0%, 18.5%, then 22.0%), the signature of a filter that oscillates between over- and under-exclusion.

F1 and accuracy. At 40% attackers BAF-Defense retains 53.3% accuracy and 53.1% F1, well above FedAvg at 42.0% and LFighter at 44.1% accuracy and 44.0% F1. At 10% attackers it reaches 58.9% accuracy, close to the 61.2% achieved by FedAvg on clean data.

Table III consolidates these measurements. At 40% attackers BAF-Defense cuts ASR by 67% relative to FedAvg and by 59% relative to LFighter, while recovering 11.3 and 9.2 accuracy points respectively; on clean data it matches both references (61.0% against 61.2% and 60.5%), so the added filtering does not erode clean-task performance. The 0% row is a utility reference rather than a robustness comparison. With no attacker present all three aggregators receive the same updates, so the only question it answers is whether the filter discards useful benign contributions, and it does not, to within 0.2 accuracy points of FedAvg.

Broader baselines. We summarize the broader baseline set qualitatively for space and place each method in Table I. Median, Trimmed Mean, and Multi-Krum deliver only partial ASR reduction and trail LFighter, because flipped-label gradients are not coordinate-wise outliers, so trimming and distance-based selection make weak filters here. Fools-Gold helps at low attacker fractions but degrades as that fraction grows. Tolpegin’s PCA-based filtering achieves a meaningful ASR reduction, whereas FLAME, designed for backdoor defense, offers limited protection against flipping. Both label-flipping-specific methods outperform the general

TABLE III
SUMMARY OF RESULTS ON CIFAR-10 (RESNET-18). BEST VALUES IN **BOLD**. ↓: LOWER IS BETTER; ↑: HIGHER IS BETTER.

Metric	Attacker %	FedAvg	LFighter	BAF-Defense
ASR (%) ↓	0%	0.0	0.0	0.0
	10%	10.0	6.5	5.0
	20%	14.0	8.9	5.6
	30%	19.0	12.9	7.5
	40%	26.0	21.1	8.6
Accuracy (%) ↑	0%	61.2	60.5	61.0
	10%	57.5	58.9	58.9
	20%	54.0	56.8	56.8
	30%	48.0	52.1	55.8
	40%	42.0	44.1	53.3
COV_Src_Acc (%) ↓	0%	n/a	n/a	n/a
	10%	19.3	21.4	15.9
	20%	18.4	21.0	15.3
	30%	17.9	18.5	15.0
	40%	17.4	22.0	14.8

TABLE IV
CROSS-DATASET RESULTS AT 40% ATTACKERS UNDER NON-IID SPLITS. BAF REPORTS THE BEST AE CONFIGURATION PER DATASET, SELECTED ON HELD-OUT ROUNDS RATHER THAN THE REPORTED TEST METRIC; LF DENOTES LFIghter.

Dataset	LF ASR	BAF ASR	LF Acc.	BAF Acc.
CIFAR-10	21.15	8.59	44.07	53.34
BloodMNIST	7.04	2.08	70.15	75.36
MNIST	0.23	0.15	98.62	98.67
OrganAMNIST	5.26	2.57	80.98	82.91
PathMNIST	1.27	1.71	76.00	79.30

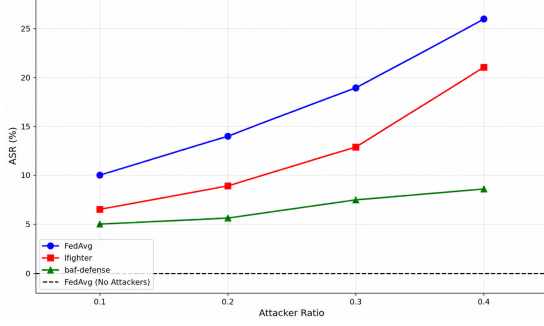
robust-aggregation baselines, and BAF-Defense is strongest under this threat model.

Table IV reports 40%-attacker results across datasets. The AE helps most where LFighter’s raw-gradient clustering is unstable (CIFAR-10, BloodMNIST); on MNIST both reach near-zero ASR. PathMNIST shows a limitation: BAF-Defense improves accuracy but not ASR at the best-accuracy setting, so the fusion weight and threshold should track the deployment objective (ASR vs. utility), especially for small medical-image datasets.

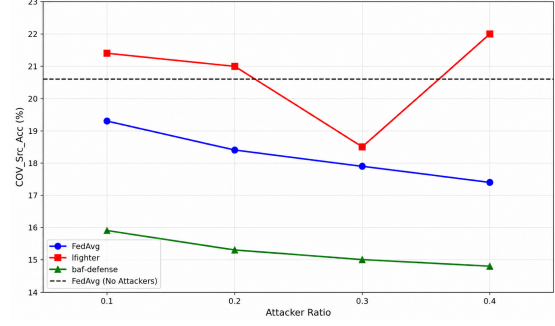
Sensitivity. The defense is stable for $\alpha \in [0.7, 0.9]$. Below 0.5 the reconstruction score dominates and performance drops, while above 0.9 the rule collapses back onto LFighter, so the default $\alpha = 0.8$ keeps both signals active. Among the bottleneck sizes 16, 32, 64, and 128, the smallest underfits the benign manifold and the largest retains too much noise, which leaves 32 as the default choice.

C. Ablation

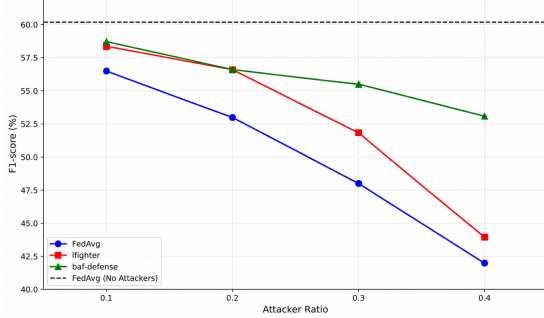
Removing the AE branch leaves LFighter alone. This raises both the ASR and the instability (a higher COV_Src_Acc) at every attacker ratio. The accuracy gap widens with attack strength, from roughly comparable at 10% to 20% attackers to as much as 9.2 points at 30% to 40%. The AE matters most when the clustering decision is ambiguous. Removing the clustering branch instead, and scoring on reconstruction alone,



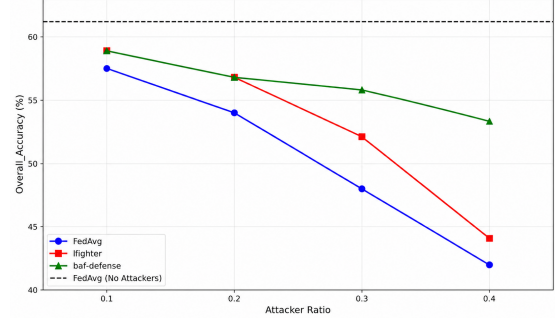
(a) ASR vs. attacker ratio



(b) COV_Src_Acc vs. attacker ratio



(c) F1-score vs. attacker ratio



(d) Overall accuracy vs. attacker ratio

Fig. 2. Primary CIFAR-10 results (ResNet-18) as the attacker ratio varies. (a) ASR (lower is better). (b) COV_Src_Acc (lower is more stable). (c) F1-score. (d) Overall accuracy. In each panel the dashed line marks the clean-data FedAvg reference (no attackers); this reference is omitted from the COV_Src_Acc rows of Table III, where the attacker ratio is the controlled variable. The reference bounds attainable utility and is not a robustness baseline.

raises the ASR to 15.2% at 40% attackers against the fused defense’s 8.6%: reconstruction error flags general anomalies but cannot identify which classes are actually under attack. The two branches are insufficient in different ways, and the gain comes from fusing them.

D. Feature-Space Analysis

Figure 3 presents a t-distributed stochastic neighbor embedding (t-SNE) of the client updates at round 200. Both methods cluster on the *raw* normalized features $\hat{g}_k$ rather than on the AE latent space. Since the AE contributes only a reconstruction score to the fusion, these plots illustrate the shared feature structure seen by both branches rather than any causal effect of the AE on clustering. Under BAF-Defense (Figure 3a) the benign and malicious clients separate cleanly. Under LFighter (Figure 3b) the clusters overlap and the boundary is ambiguous, which is consistent with the higher false-positive and false-negative rates that heterogeneity induces [9]. This panel is drawn from a separate run whose round-200 features differ from those under BAF-Defense, because the two defenses filter different clients as training proceeds. Benign clients drawn from different distributions still group together, while malicious clients occupy a distinct region. This matches the higher reconstruction error the AE assigns them. The boundary clients, which a hard decision most often misclassifies, receive intermediate scores. These intermediate scores let the fused score reach a better-calibrated

verdict. The measurable form of the same effect is the cluster assignment itself. Across rounds 150 to 200, BAF-Defense assigns up to 98% of clients to the cluster matching their true status, against 85 to 90% for LFighter, and fewer than 2% of its assignments change between consecutive rounds, against 5 to 10% for LFighter. The separation is therefore both more accurate and more stable than the projections alone convey.

V. DISCUSSION AND LIMITATIONS

Interpretation. BAF-Defense’s improvement comes from placing a continuous trust signal alongside LFighter’s discrete decision rather than from altering the space in which clustering occurs. Fusing the two scores produces a graded measure of suspicion, most informative at the boundary where a hard assignment is least reliable. The ablation supports this mechanism and the t-SNE views are consistent with it, though we present the explanation as an empirical one rather than a formal robustness proof.

Attack coverage. Our evaluation targets single-pair label flipping, following LFighter. Multi-class flipping, dynamic source/target switching, backdoors, and model replacement all deviate from the single source-to-target pattern our detector relies on [6], [15]. Backdoor gradients carry the main-task and backdoor-task signals at once, and untargeted poisoning has no source-to-target structure. Covering them would call for attack-specific AE features or a multi-task detector.

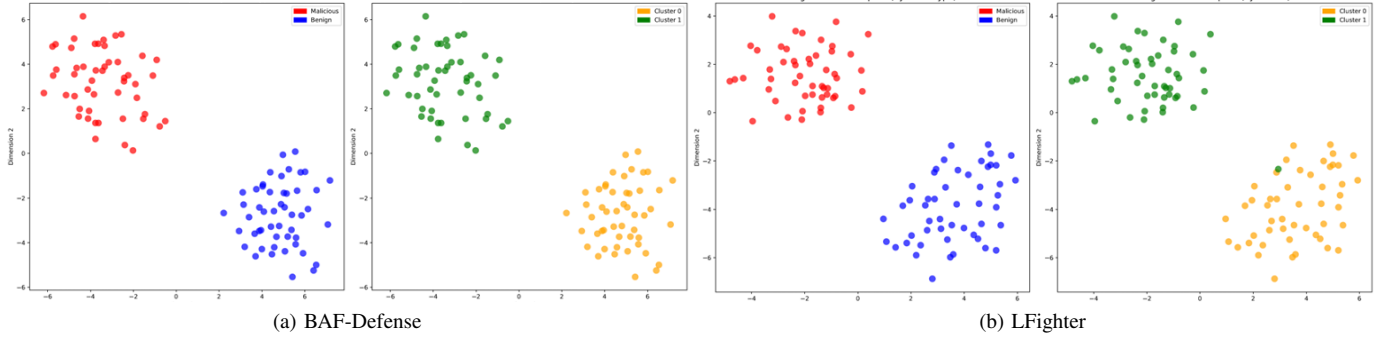


Fig. 3. t-SNE of the raw normalized gradient features $\hat{g}_k$ at round 200. (a) Under BAF-Defense benign and malicious clients separate cleanly. (b) Under LFighter boundaries are less distinct. Cluster decisions use $\hat{g}_k$; the AE contributes only a reconstruction score for fusion.

Adaptive attacker. An adversary who knows BAF-Defense could craft updates that satisfy both branches at once, for instance by minimizing

$$\mathcal{L}_{adv} = \mathcal{L}_{flip} + \lambda e_k + \gamma \|\hat{g}_k - \mu_b\|_2^2, \quad (4)$$

where $\mathcal{L}_{flip}$ is the attack objective, e_k the reconstruction error, and μ_b an estimate of the benign cluster center, so that the attacker trades attack strength against detectability. We do not claim robustness against white-box adaptive attackers. The relevant question is the rate at which detection degrades as the attacker optimizes against the AE branch. Sweeping λ and γ and charting the ASR–evasion frontier would quantify this degradation, and randomized AE retraining and ensemble detection would aim to blunt it at some added server cost. Because BAF-Defense reads each update directly, it falls outside standard secure aggregation and offers no guarantee against backdoor or model-replacement adversaries.

Threshold sensitivity. We set q to the true attacker ratio in all reported runs and applied the same convention to LFighter, so the comparison is controlled but optimistic for both defenses. The band structure of the fused score bounds what an incorrect q costs. With $\alpha = 0.8$ the fused scores occupy two disjoint bands: $[0, 0.2]$ for clients the cluster branch rejects and $[0.8, 1.0]$ for those it accepts. An estimate above the true ratio moves the cut into the upper band, where it removes the accepted clients the AE ranks as least typical, costing utility while excluding every client the cluster branch already rejected. An estimate below the true ratio moves the cut into the lower band, where it retains the rejected clients with the smallest reconstruction error, and ASR rises toward LFighter’s. The AE ordering therefore governs which clients are lost or retained first in either direction. At deployment q comes from the LFighter cluster-size estimate, so the quantity that matters is the error of that proxy under non-IID splits. This is an analytical consequence of the fusion rule rather than a measurement; charting accuracy and ASR as the estimated ratio deviates from the true one is the experiment the argument calls for, and we leave it to future work.

Overhead. BAF-Defense adds a single AE pass per round, roughly 2.3 s against the ≈ 180 s spent on client training and aggregation on an A100, about 1.3% of the round time. We

report this as indicative rather than a controlled benchmark. Scaling in the client count is linear and the ratio to aggregation is fixed. Each round the server trains the AE on the n gradient vectors and runs one forward pass per client, so the cost is $O(n)$ with a per-client constant set by $d = 2C$ and roughly 13K AE parameters, independent of the model the clients train. Aggregation over the same n updates is also $O(n)$, with a per-client constant set by the millions of parameters in the client model, so both terms grow at the same rate and the AE’s share of round time is approximately invariant in n . Memory is $O(nd)$: under 1 MB at $n = 100$ and $d = 20$, against roughly 45 MB for the ResNet-18 weights the server already holds. The AE’s share does grow when clients train a smaller model or when the server is resource-constrained. We retrain the AE from scratch each round; incremental or periodic retraining removes the training pass on most rounds and is the first cost reduction to try.

Statistical scope. All reported results are single runs of 200 rounds under a fixed seed. COV_Src_Acc characterizes within-run stability, which does not substitute for variance across seeds. The margins we rely on are large (12.5 ASR points and 9.2 accuracy points over LFighter at 40% attackers on CIFAR-10) and the accuracy ordering holds on all five datasets in Table IV, but establishing significance requires a multi-seed study with confidence intervals.

Privacy. The features scored by BAF-Defense are aggregated last-layer statistics rather than raw activations. Gradient-inversion attacks typically require intermediate-layer gradients together with architectural knowledge, so the exposure here is limited, but residual leakage still merits analysis before deployment in sensitive healthcare or financial settings.

VI. CONCLUSION

We studied the security of federated learning against label-flipping poisoning under non-IID data and proposed BAF-Defense, a server-side extension of LFighter that pairs raw-gradient clustering with an AE reconstruction trust score. BAF-Defense leaves client training and communication untouched and adds only a lightweight server-side module. On CIFAR-10 with a ResNet-18 backbone it holds the ASR to 8.6% at 40% attackers, against 21.1% for LFighter and 26.0%

for FedAvg, while preserving higher accuracy and F1. The MNIST and MedMNIST results broadly agree, although on some high-attacker medical-image cases BAF-Defense improves accuracy without reducing ASR, which points to a need for dataset-specific tuning. Our contribution is a calibrated fusion of LFighter’s cluster signal with a continuous AE trust score: recovering the confidence a hard boundary discards more than halves the attack success rate of the strongest prior defense.

Future work. The first priorities are a multi-seed study with confidence intervals and a sensitivity sweep over the estimated attacker ratio q , including the error of the LFighter cluster-size proxy under non-IID splits. Beyond these, a direct evaluation against *adaptive adversaries* that minimize reconstruction error to evade the AE branch, together with the dynamic and ensemble AE countermeasures such attacks motivate. A second direction is reducing the server-side cost through lighter or pre-trained encoders and warm-started incremental updates. Further directions include extending the method to other modalities and attack types, a theoretical account linking latent-space separation to provable ASR bounds, and validation under secure aggregation, differential privacy, and client sampling.

ACKNOWLEDGMENT

The authors used Anthropic’s Claude (Claude Opus 4.8) to assist with language editing, restructuring for clarity, and identifying factual and citation inconsistencies in the manuscript. All experiments, results, and technical claims are the authors’ own and were verified by them; the tool was not used to generate research findings.

REFERENCES

- [1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, “Communication-Efficient Learning of Deep Networks from Decentralized Data,” in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Singh and J. Zhu, Eds., vol. 54. PMLR, 20–22 Apr 2017, pp. 1273–1282. [Online]. Available: <https://proceedings.mlr.press/v54/mcmahan17a.html>
- [2] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, “Federated learning: Challenges, methods, and future directions,” *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.
- [3] M. J. Sheller, B. Edwards, G. A. Reina, J. Martin, S. Pati, A. Kotrotsou, M. Milchenko, W. Xu, D. Marcus, R. R. Colen, and S. Bakas, “Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data,” *Scientific Reports*, vol. 10, no. 1, p. 12598, Jul 2020.
- [4] S. Zhang, J. Li, L. Shi, M. Ding, D. C. Nguyen, W. Tan, J. Weng, and Z. Han, “Federated learning in intelligent transportation systems: Recent applications and open problems,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 5, pp. 3259–3285, 2024.
- [5] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, “Federated learning for internet of things: Recent advances, taxonomy, and open challenges,” *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1759–1799, 2021.
- [6] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, “How to backdoor federated learning,” in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, S. Chiappa and R. Calandra, Eds., vol. 108. PMLR, 26–28 Aug 2020, pp. 2938–2948. [Online]. Available: <https://proceedings.mlr.press/v108/bagdasaryan20a.html>
- [7] V. Tolpegin, S. Truex, M. E. Gursoy, and L. Liu, “Data poisoning attacks against federated learning systems,” in *Computer Security – ESORICS 2020*, L. Chen, N. Li, K. Liang, and S. Schneider, Eds. Cham: Springer International Publishing, 2020, pp. 480–501.
- [8] K. N. Kumar, C. K. Mohan, and L. R. Cenkeramaddi, “The impact of adversarial attacks on federated learning: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 5, pp. 2672–2691, 2024.
- [9] N. M. Jebreel, J. Domingo-Ferrer, D. Sánchez, and A. Blanco-Justicia, “LFighter: Defending against the label-flipping attack in federated learning,” *Neural Networks*, vol. 170, pp. 111–126, 2024.
- [10] S. Awan, B. Luo, and F. Li, “Contra: Defending against poisoning attacks in federated learning,” in *Computer Security – ESORICS 2021*, E. Bertino, H. Shulman, and M. Waidner, Eds. Cham: Springer International Publishing, 2021, pp. 455–475.
- [11] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, “Byzantine-robust distributed learning: Towards optimal statistical rates,” in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 5650–5659. [Online]. Available: <https://proceedings.mlr.press/v80/yin18a.html>
- [12] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, “Deep learning for anomaly detection: A review,” *ACM Comput. Surv.*, vol. 54, no. 2, Mar. 2021.
- [13] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, “Machine learning with adversaries: Byzantine tolerant gradient descent,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/f4b9ec30ad9f6f8f89b29639786cb62ef-Paper.pdf
- [14] C. Fung, C. J. M. Yoon, and I. Beschastnikh, “The limitations of federated learning in sybil settings,” in *23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020)*. San Sebastian: USENIX Association, Oct. 2020, pp. 301–316. [Online]. Available: <https://www.usenix.org/conference/raid2020/presentation/fung>
- [15] M. Fang, X. Cao, J. Jia, and N. Gong, “Local model poisoning attacks to Byzantine-Robust federated learning,” in *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, Aug. 2020, pp. 1605–1622. [Online]. Available: <https://www.usenix.org/conference/usenixsecurity20/presentation/fang>
- [16] T. D. Nguyen, P. Rieger, H. Chen, H. Yalame, H. Möllering, H. Fereidooni, S. Marchal, M. Miettinen, A. Mirhoseini, S. Zeitouni, F. Koushanfar, A.-R. Sadeghi, and T. Schneider, “FLAME: Taming backdoors in federated learning,” in *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, Aug. 2022, pp. 1415–1432. [Online]. Available: <https://www.usenix.org/conference/usenixsecurity22/presentation/nguyen>
- [17] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [18] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, “Extracting and composing robust features with denoising autoencoders,” in *Proceedings of the 25th International Conference on Machine Learning*, ser. ICML ’08. New York, NY, USA: Association for Computing Machinery, 2008, p. 1096–1103.
- [19] A. Krizhevsky, G. Hinton *et al.*, “Learning multiple layers of features from tiny images,” 2009.
- [20] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, *Object Recognition with Gradient-Based Learning*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 319–345.
- [21] J. Yang, R. Shi, D. Wei, Z. Liu, L. Zhao, B. Ke, H. Pfister, and B. Ni, “Medmnist v2 - a large-scale lightweight benchmark for 2d and 3d biomedical image classification,” *Scientific Data*, vol. 10, no. 1, p. 41, 2023.