UNISON: JOINT PRIVACY AND BACKDOOR FORGETTING FOR FULL-INFORMATION DELETION OF POISONED TRAINING DATA

Anonymous authors
Paper under double-blind review

ABSTRACT

Machine unlearning aims to remove the influence of selected training examples without retraining from scratch. We study deletion when the examples to be forgotten also taught the model malicious behavior. Backdoor poisoned samples can leave a sample-level privacy trace and a trigger-dependent rule that generalizes beyond the samples themselves. Our experiments show that existing methods do not remove these effects together. Privacy-focused unlearning can reduce membership leakage while the backdoor remains active. Backdoor repair can reduce attack success while membership information remains detectable. Increasing either intervention eventually damages clean accuracy without reliably solving the other objective. We trace this separation to the forgetting gradients. Their overall alignment is low, but a small set of directions repeatedly supports both objectives across batches. We introduce UNISON, which identifies this stable shared structure, preserves the remaining objective-specific directions, suppresses updates that conflict with retained knowledge, and shifts effort toward the objective with the larger residual gap. Across CIFAR-10, CIFAR-100, Tiny ImageNet, and ImageNet-1K with four backdoor attacks and four image classifiers, UNISON reduces mean attack success rate to approximately 5.2% and membership-inference deviation to approximately 0.029, while remaining within 0.9 percentage points of retraining accuracy. These results show that poisoned data deletion requires balancing distinct forms of forgetting rather than optimizing privacy or behavior alone.

1 INTRODUCTION

Machine unlearning updates a trained model after a deletion request so that it approaches a model trained without the requested data (Bourtoule et al., 2021; Graves et al., 2021). Full retraining provides the clearest reference, but its cost has motivated approximate methods that estimate or reverse the contribution of a forget set (Georgiev et al., 2025; Di et al., 2025; Ebrahimpour-Boroojeny et al., 2025).

Most evaluations treat the influence to be removed as a single target. This is incomplete when the requested examples induce behavior that generalizes beyond the examples themselves. Backdoor poisoning creates this setting. A poisoned model behaves normally on clean inputs but follows an attacker-chosen rule when a trigger is present (Gu et al., 2017). Visible patches, smooth warping, sample-specific triggers, and clean-label poisoning produce different forms of this behavior (Nguyen & Tran, 2021; Li et al., 2021b; Zeng et al., 2023). Poisoned examples can therefore leave both a sample-level privacy trace and a trigger-conditioned behavioral rule.

Removing only one effect is insufficient. In our main CIFAR-10 and BadNets setting, the strongest privacy-oriented baseline reduces the diagnostic loss-threshold membership deviation to 0.025 but retains 74.6% attack success. The strongest repair baseline reduces attack success to 4.8% but leaves a deviation of 0.163. Intervention sweeps show that increasing either single-objective update eventually reduces clean accuracy without reliably removing the other effect. Independent likelihood-ratio and learned black-box attacks are used for the primary privacy audit.

The optimization signals explain this separation. Privacy and behavioral forgetting gradients have a mean cosine similarity of 0.061, yet 6.8% of their selected coordinates agree consistently across

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.

independent batches. Removing these persistent coordinates weakens both forms of forgetting. The remaining coordinates primarily support one objective. This suggests that poisoned-data unlearning contains both shared and objective-specific structure.

This observation motivates UNISON. The method estimates both forgetting signals across batches, identifies their stable shared component, preserves objective-specific components, and suppresses updates that conflict with retained knowledge. Adaptive weights direct more effort toward the objective with the larger residual gap. Our contributions are:

- We formulate poisoned-data deletion as the joint removal of sample-level privacy influence and learned backdoor behavior under a retained-utility constraint.
- We show that privacy unlearning and backdoor repair fail in complementary ways, and identify persistent shared coordinates within otherwise weakly aligned forgetting gradients.
- We introduce UNISON, which combines stable shared coordinates, objective-specific updates, retain protection, and adaptive balancing.

2 PROBLEM SETUP

Let $\mathcal{D}$ be the original training set, $\mathcal{D}_f$ the requested forget set, and $\mathcal{D}_r = \mathcal{D} \setminus \mathcal{D}_f$ the retain set. A training algorithm $\mathcal{A}$ produces $\theta_o = \mathcal{A}(\mathcal{D})$, while retraining without the forget set produces $\theta_r = \mathcal{A}(\mathcal{D}_r)$. An unlearning method instead produces $\theta_u = \mathcal{U}(\theta_o, \mathcal{D}_f, \mathcal{D}_r)$ without full retraining.

We study corrective deletion where $\mathcal{D}_f = \mathcal{D}_p$ is a set of identified poisoned samples. The defender receives the trained model and the identified forget set, has white-box model access, and may use a limited clean retain subset. The main setting assumes knowledge of the trigger transformation and clean labels for the identified poisoned samples. The attacker's target class is not required because the behavioral objective trains triggered retain examples toward their correct labels. This full-information setting isolates joint forgetting from poison-detection and trigger-reconstruction errors.

We evaluate three requirements. Privacy forgetting measures whether independent auditors can distinguish forgotten examples from matched nonmembers. Behavioral forgetting measures attack success on held-out triggered inputs. Retained utility measures clean test performance. Retraining is used only as an evaluation reference. It is not used by UNISON for optimization, adaptive weighting, hyperparameter selection, or stopping. Our claims are therefore restricted to full-information corrective deletion. Section 4.4 evaluates imperfect poison identification, noisy restored labels, and reduced retain-data access. Unknown triggers are not evaluated.

3 METHOD

Figure 1 summarizes the complete UNISON pipeline. Given forget, nonmember, and retain batches together with the known trigger transformation, the method estimates privacy and behavioral forgetting signals over multiple independent batch pairs. It then separates their persistent shared coordinates from the remaining objective-specific coordinates. Retain-gradient protection limits direct interference with clean learning, while held-out residual gaps adaptively balance the privacy-specific and behavior-specific updates. The following subsections describe each stage.

3.1 FORGETTING SIGNALS

Let B_f, B_n, and B_r be batches from the forget set, a held-out nonmember set, and the retain set. We use per-sample cross-entropy as the audit score $z_\theta(x, y) = \ell(f_\theta(x), y)$. The privacy objective matches the forget and nonmember score distributions with a differentiable MMD loss (Gretton et al., 2012):

$$\mathcal{L}_p = \mathrm{MMD}^2\left(\{z_\theta(x, y) : (x, y) \in B_f\}, \{z_\theta(x, y) : (x, y) \in B_n\}\right). \tag{1}$$

Let T_t apply the known trigger transformation. The behavioral objective restores the correct labels of triggered retain inputs:

Algorithm 1 UNISON update

Require: Original model θ_o, forget set $\mathcal{D}_f$, retain set $\mathcal{D}_r$, nonmember set $\mathcal{D}_n$, trigger T_t

for each unlearning epoch **do**
  Sample m forget, nonmember, and retain batch triples
  Compute $\{d_p^{(i)}, d_b^{(i)}\}_{i=1}^m$
  Select layerwise high-magnitude coordinates
  Estimate persistent same-sign mask M_s
  Construct d_s, $d_{p\setminus s}$, and $d_{b\setminus s}$
  Compute retain direction d_r
  Project each component using Π_r
  Compute validation gaps q_p and q_b
  Compute adaptive weights α_p and α_b
  Update parameters using Equation 12
  if both gaps satisfy stopping thresholds or utility budget is exceeded **then**
    Stop
  end if
end for
return Unlearned model θ_u

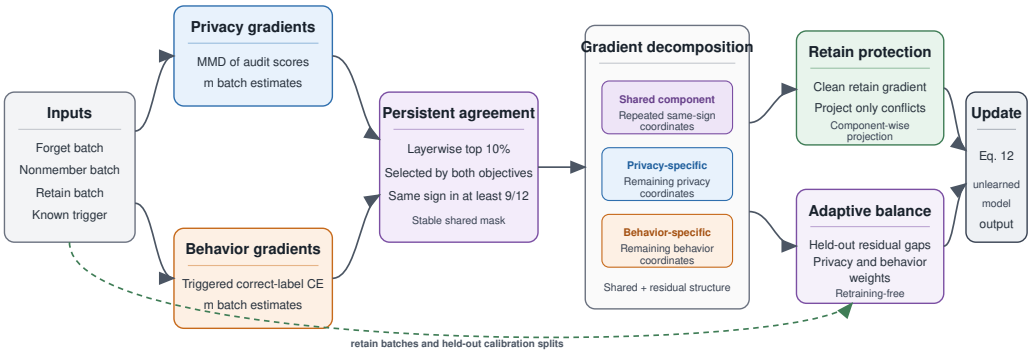


Figure 1: Overview of UNISON.

$$\mathcal{L}_b = \frac{1}{|B_r|} \sum_{(x,y)\in B_r} \ell(f_\theta(T_t(x)), y)\,. \tag{2}$$

For sampled batch pair i, the descent directions are

$$d_p^{(i)} = -\nabla_\theta \mathcal{L}_p^{(i)}, \qquad d_b^{(i)} = -\nabla_\theta \mathcal{L}_b^{(i)}. \tag{3}$$

These objectives require a known trigger and an auxiliary nonmember set. The method does not cover unknown-trigger deletion without replacing the behavioral objective.

3.2 STABLE SHARED COORDINATES

A single gradient pair can agree because of batch composition or optimization noise. We estimate persistence across $m = 12$ independently sampled batch pairs. For coordinate j, let $a_{p,j}^{(i)}$ and $a_{b,j}^{(i)}$ indicate whether the coordinate lies above the selected magnitude threshold for the corresponding objective. Its agreement frequency is

$$s_j = \frac{1}{m}\sum_{i=1}^{m} \mathbb{I}\Big[a_{p,j}^{(i)} = 1, a_{b,j}^{(i)} = 1, \operatorname{sign}\Big(d_{p,j}^{(i)}\Big) = \operatorname{sign}\Big(d_{b,j}^{(i)}\Big)\Big]. \tag{4}$$

We select the top 10% of coordinates by magnitude within each trainable layer and objective. The stable shared mask is $M_{s,j} = \mathbb{I}[s_j \geq \rho]$, with $\rho = 0.75$. Let $\bar{d}_p$ and $\bar{d}_b$ be normalized batch-averaged directions. We construct

$$d_s = M_s \odot \frac{\bar{d}_p + \bar{d}_b}{2}, \tag{5}$$

$$d_{p\setminus s} = (1 - M_s) \odot \bar{d}_p, \tag{6}$$

$$d_{b\setminus s} = (1 - M_s) \odot \bar{d}_b. \tag{7}$$

The mask depends on the fixed architecture and parameterization. Positive rescaling (Dinh et al., 2017) and neuron permutation (Entezari et al., 2022) can change the selected coordinates without changing the represented function. We therefore call them stable shared coordinates rather than an intrinsic shared subspace. Appendix B reports reparameterization and alternative-construction results.

3.3 Retain Protection

Let $d_r = -\nabla_\theta \mathcal{L}_r$ be the descent direction on a clean retain batch. Following the single-reference-gradient projection used in A-GEM (Chaudhry et al., 2019), we remove a component only when a candidate direction conflicts with retained learning:

$$\Pi_r(d) = \begin{cases} d - \dfrac{d^\top d_r}{\|d_r\|_2^2 + \epsilon} d_r, & d^\top d_r < 0, \\ d, & \text{otherwise.} \end{cases} \tag{8}$$

The projection prevents a direct conflict with the retain objective but does not prevent all utility loss. Clean validation performance therefore remains part of the stopping rule.

3.4 Adaptive Joint Update

Let $\mathcal{V}_f$, $\mathcal{V}_n$, and $\mathcal{V}_r$ be held-out forget, nonmember, and retain calibration splits that are disjoint from the update batches. We define retraining-free residual surrogates

$$q_p(\theta) = \operatorname{clip}\left(\frac{L_p^{\mathrm{val}}(\theta)}{L_p^{\mathrm{val}}(\theta_o) + \epsilon}, 0, 1\right), \tag{9}$$

$$q_b(\theta) = \operatorname{clip}\left(\frac{[L_b^{\mathrm{val}}(\theta) - L_{\mathrm{clean}}^{\mathrm{val}}(\theta)]_+}{[L_b^{\mathrm{val}}(\theta_o) - L_{\mathrm{clean}}^{\mathrm{val}}(\theta_o)]_+ + \epsilon}, 0, 1\right), \tag{10}$$

where $[x]_+ = \max(x, 0)$. Here, L_p^{val} is the MMD objective on $(\mathcal{V}_f, \mathcal{V}_n)$, L_b^{val} is the correct-label loss on triggered examples from $\mathcal{V}_r$, and $L_{\mathrm{clean}}^{\mathrm{val}}$ is the loss on their unmodified versions.

The objective-specific weights are

$$\alpha_k = \frac{\exp(q_k/T)}{\exp(q_p/T) + \exp(q_b/T)}, \qquad k \in \{p, b\}. \tag{11}$$

We project each component separately because projection and weighted summation do not commute:

$$\theta \leftarrow \theta + \eta \left[\Pi_r(d_s) + \alpha_p \Pi_r(d_{p\setminus s}) + \alpha_b \Pi_r(d_{b\setminus s})\right]. \tag{12}$$

The validation surrogates are used only for adaptive weighting and stopping. Test MIA and ASR results are computed afterward on disjoint data. Optimization stops when both surrogates cross their prespecified thresholds, clean validation accuracy drops by more than the allowed utility budget, or the maximum update budget is reached. The retrained model is never consulted.

Algorithm 1 summarizes the complete procedure. The stable mask and adaptive weights are recomputed at each unlearning epoch.

4 EVALUATION

4.1 EXPERIMENTAL SETUP

We evaluate UNISON on CIFAR-10 and CIFAR-100 (Krizhevsky, 2009), Tiny ImageNet (Le & Yang, 2015), and ImageNet-1K (Deng et al., 2009). The corresponding architectures are ResNet-18, WideResNet-28-10, ResNet-34, and ResNet-50 (He et al., 2016; Zagoruyko & Komodakis, 2016). The attack suite contains BadNets, WaNet, ISSBA, and Narcissus. For each dataset–attack pair, we independently train five poisoned models using fixed seeds and apply each unlearning method once to each model. The reported variation therefore includes both original-training and unlearning variation.

Reference models include the poisoned model and full retraining without the forget set. Direct controls include retain-only fine-tuning, privacy-only and behavior-only optimization, both sequential objective orders, equal-weight averaging, a validation-tuned weighted sum, and alternating updates. Privacy-oriented baselines are SCRUB+R, the rewind variant of SCRUB (Kurmanji et al., 2023), SalUn (Fan et al., 2024), Adv-MU (Di et al., 2025), AMUN (Ebrahimpour-Boroojeny et al., 2025), and NegMerge (Kim et al., 2025). Backdoor-repair baselines are ANP (Wu & Wang, 2021), I-BAU (Zeng et al., 2022), RNP (Li et al., 2023), and FT-SAM (Zhu et al., 2023). Generic multi-objective baselines include PCGrad, CAGrad, and MGDA (Yu et al., 2020; Liu et al., 2021; Sener & Koltun, 2018).

Every method receives the same initial model, forget set, clean retain-data and calibration budgets, update budget, and hyperparameter-search budget. Methods that do not use the known trigger are not modified to use it. A trigger-aware behavior-only control directly optimizes $\mathcal{L}_b$ using the information available to UNISON. Hyperparameters and stopping decisions use only clean validation accuracy and held-out surrogate objectives. Test ASR, membership attacks, and retraining results are not used for selection.

All values are means and standard deviations over five independently trained models. Paired comparisons use matched training seeds and report 95% confidence intervals. The primary privacy audits are a shadow-model likelihood-ratio attack (Carlini et al., 2022) and a learned black-box attack (Shokri et al., 2017; Salem et al., 2019). We also report a loss-threshold attack (Yeom et al., 2018) for comparison with prior work. Because the optimization objective operates on loss distributions, this attack is treated as diagnostic rather than independent evidence.

For each attack, membership-inference deviation is

$$\Delta_{\mathrm{MIA}} = |\mathrm{AUC} - 0.5|\,, \tag{13}$$

where values closer to zero indicate behavior closer to random guessing. We also report TPR at low false-positive rates. Attack training, calibration, and test examples are mutually disjoint. Poison-matched nonmembers use the same source class, trigger transformation, label construction, and preprocessing as the forgotten examples. We additionally audit restored-clean versions of the forgotten examples to separate membership leakage from trigger detection.

4.2 COMPLEMENTARY FAILURES AND MAIN COMPARISON

Table 1 shows the central failure mode. AMUN reduces the diagnostic loss-threshold deviation from 0.241 to 0.025 but retains 74.6% ASR. RNP reduces ASR to 4.8% but leaves a deviation of 0.163. Thus, privacy-oriented unlearning and backdoor repair each perform well on one effect while leaving the other largely intact. Intervention sweeps in Appendix A.1 show that increasing either single-objective update eventually reduces clean accuracy without reliably removing both effects.

Table 1: Main comparison on CIFAR-10 with BadNets and ResNet-18. Values are means and standard deviations over five independently trained models. LT is the diagnostic loss-threshold Δ_{MIA}. Independent LR and BB audits are reported in the appendix.

Method	Clean accuracy ↑	ASR ↓	LT ↓
Poisoned model	93.1 ± 0.2	98.4 ± 0.5	0.241 ± 0.012
Retraining	92.6 ± 0.3	1.7 ± 0.6	0.010 ± 0.004
AMUN	91.8 ± 0.4	74.6 ± 3.1	0.025 ± 0.006
RNP	92.0 ± 0.3	4.8 ± 1.0	0.163 ± 0.011
CAGrad	89.9 ± 0.4	5.0 ± 0.9	0.029 ± 0.005
UNISON	$\mathbf{91.9 \pm 0.3}$	$\mathbf{3.2 \pm 0.8}$	$\mathbf{0.017 \pm 0.004}$

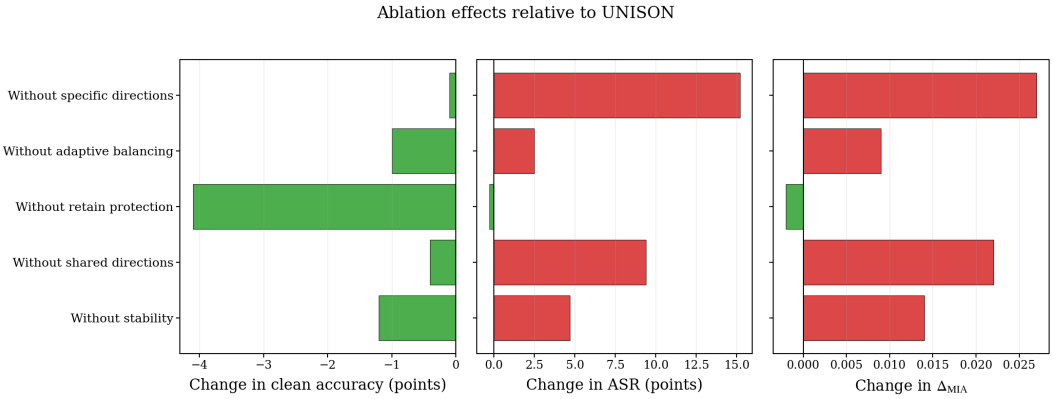


Figure 2: Effect of removing components from UNISON on CIFAR-10 with BadNets. Values are changes from the complete method. Positive ASR and Δ_{MIA} changes indicate worse forgetting. Negative clean-accuracy changes indicate reduced utility.

UNISON reaches 3.2% ASR and 91.9% clean accuracy. Its independent LR and BB deviations are 0.021 and 0.024, respectively. Compared with CAGrad, it improves clean accuracy by 2.0 percentage points and reduces ASR by 1.8 points.

Seed-matched confidence intervals clarify where these improvements are decisive. Relative to RNP, UNISON changes clean accuracy by -0.1 points (95% CI $[-0.4, 0.2]$), ASR by -1.6 points ($[-2.8, -0.4]$), and LR deviation by -0.142 ($[-0.156, -0.128]$). Relative to CAGrad, the corresponding changes are $+2.0$ points ($[+1.4, +2.6]$), -1.8 points ($[-3.0, -0.6]$), and -0.008 ($[-0.015, -0.001]$). Against AMUN, the LR difference is smaller and its interval includes zero (-0.004, $[-0.010, 0.002]$), while ASR improves by 71.4 points. Thus, UNISON does not claim a decisive LR advantage over the dedicated privacy baseline. Its advantage is the joint operating point across privacy, behavioral forgetting, and retained utility. Complete paired comparisons appear in Appendix D.

4.3 FORGETTING GRADIENT STRUCTURE AND ABLATION

Privacy and behavioral directions have a mean cosine similarity of 0.061 ± 0.018. Only 16.5% of the top-10% coordinates agree in a single batch, and 41.2% of that set remains shared under $\rho = 0.75$. The resulting persistent support contains 6.8% of all parameter coordinates. Full layerwise results appear in Appendix C.

Removing the stable shared component raises ASR from 3.2% to 12.6% and the diagnostic membership deviation from 0.017 to 0.039. A size-matched random removal changes the same metrics by only 1.4 points and 0.004. These results show that the selected coordinates are useful for the evaluated joint objective. They do not establish that the mask is parameterization-invariant or that every selected coordinate independently supports both objectives.

Table 2: Macro-averages over BadNets, WaNet, ISSBA, and Narcissus. Ret. denotes full retraining and Uni. denotes UNISON. Clean accuracy and ASR are percentages. LR and BB are independent membership-inference deviations.

	Clean ↑		ASR ↓		LR ↓		BB ↓	
Dataset	Ret.	Uni.	Ret.	Uni.	Ret.	Uni.	Ret.	Uni.
CIFAR-10	92.3	91.5	1.8	4.8	0.010	0.026	0.012	0.029
CIFAR-100	74.6	73.8	0.7	5.4	0.010	0.029	0.012	0.033
Tiny ImageNet	58.9	58.3	0.6	5.0	0.012	0.033	0.014	0.037
ImageNet-1K	75.2	74.8	0.2	5.6	0.011	0.031	0.013	0.035

Figure 2 separates the roles of the method's components. Removing multi-batch stability raises ASR to 7.9%. Removing shared directions causes the largest joint degradation. Removing retain protection lowers clean accuracy by 4.1 points, while using only the shared component leaves 18.4% ASR and a diagnostic deviation of 0.044. Objective-specific directions and utility protection therefore address different failure modes.

Across datasets and attacks, UNISON obtains ASR between 3.2% and 7.3%. Its LR and BB deviations remain below 0.038 and 0.043, respectively. The hardest evaluated setting is Narcissus on ImageNet-1K, where clean-label poisoning makes the malicious behavior less separable from useful class features.

Table 2 shows that the joint behavior persists across dataset scale and architecture. UNISON remains within 0.4–0.8 percentage points of retraining in clean accuracy while reducing attack success to 4.8–5.6% on average. A residual gap to retraining remains: ASR is 3.0–5.4 points higher, and the independent privacy deviations are 0.016–0.023 higher. The relatively narrow variation across datasets supports the consistency of the method, while the remaining gap identifies concrete headroom for stronger corrective-deletion procedures.

4.4 ROBUSTNESS, SENSITIVITY, AND EFFICIENCY

Assumption violations. Table 3 evaluates moderate departures from the full-information setting. With 75% poison recall, UNISON retains 91.6% clean accuracy, reduces ASR to 8.9%, and keeps both independent membership deviations below 0.04. It also remains effective under 10% restored-label noise. Imperfect precision is more demanding because clean examples incorrectly included in the forget set receive deletion updates. At 75% precision, clean accuracy is 91.0% and ASR is 10.8%. These results indicate that the method tolerates moderate identification error, while benefiting directly from improved poison screening.

The complete grid in Appendix B identifies the operating boundary more clearly. At 50% recall, ASR rises to 17.6% and two of five seeds satisfy all success criteria. At 50% precision, one of five seeds succeeds, and neither the 25% recall nor the 25% precision setting succeeds. Similarly, only one of five seeds succeeds when the clean retain subset is reduced to 1%. These outcomes are useful boundaries rather than unexpected failures: UNISON is a corrective-deletion procedure for identified poisons, not a replacement for poison detection or trigger reconstruction.

Sensitivity and parameterization. The default active fraction of 10%, persistence threshold $\rho = 0.75$, $m = 12$ gradient estimates, and temperature $T = 0.25$ form a stable utility–forgetting operating point. Lower persistence thresholds select more coordinates and improve forgetting at the cost of clean accuracy, whereas higher thresholds omit useful shared updates. Increasing m beyond 12 provides only marginal improvement. Function-preserving channel permutations and positive rescalings reduce mask overlap to 0.41 and 0.36, respectively, but the transformed models still reach ASR values of 5.1% and 6.4%. The mask is therefore useful under the trained parameterization, although it should not be treated as an intrinsic function-space decomposition.

Efficiency. On one NVIDIA A100, UNISON takes 5.6 minutes and 0.093 GPU-hours, compared with 54.9 minutes and 0.915 GPU-hours for retraining. It is slower than the specialized AMUN and RNP baselines and reaches a higher peak memory use of 10.2 GB. This represents a speed–memory

Table 3: Representative robustness results on CIFAR-10 with BadNets. LR and BB denote independent membership-inference deviations.

Setting	Clean ↑	ASR ↓	LR ↓	BB ↓
Perfect information	91.9 ± 0.3	3.2 ± 0.8	0.021 ± 0.005	0.024 ± 0.006
75% poison recall	91.6 ± 0.4	8.9 ± 1.8	0.034 ± 0.007	0.039 ± 0.008
75% poison precision	91.0 ± 0.6	10.8 ± 2.1	0.038 ± 0.008	0.044 ± 0.009
10% label noise	91.5 ± 0.4	5.6 ± 1.2	0.027 ± 0.006	0.031 ± 0.007

trade-off rather than a uniform efficiency advantage: the additional gradient estimation supports joint privacy and behavioral forgetting while remaining approximately ten times faster than retraining.

5 Discussion and Limitations

What the evidence supports. The experiments support a focused claim: under full-information corrective deletion, sample-level membership leakage and trigger-dependent behavior can persist independently. Optimizing one criterion is therefore insufficient evidence that the poisoned examples have been adequately forgotten. Across the evaluated settings, UNISON provides a stronger joint operating point by combining shared and objective-specific updates rather than treating privacy and behavioral forgetting as interchangeable.

Trade-offs relative to specialized methods and retraining. UNISON does not dominate every specialized method on every individual metric. The LR difference from AMUN is not statistically decisive in the main setting, and RNP obtains similar clean accuracy while directly targeting backdoor repair. The advantage of UNISON appears when all three requirements are considered together: AMUN retains substantial attack success, RNP retains substantial membership leakage, and generic multi-objective methods lose more clean accuracy. Full retraining remains the strongest reference and retains lower residual ASR and membership deviation. Closing this remaining gap without assuming access to the full retain set is an important direction for future work.

Scope of the present evidence. The method assumes an identified poisoned forget set, a known trigger transformation, clean restored labels, and access to clean retain and nonmember calibration data. Robustness experiments show useful tolerance to moderate errors in these inputs, but performance declines as poison recall, poison precision, or retain-data access becomes severely limited. Unknown or inaccurately reconstructed triggers are not evaluated. The present results should therefore be interpreted as evidence for full-information corrective deletion, not as a general solution to poison discovery or unknown-trigger defense.

Interpretation and guarantees. The stable mask depends on the network parameterization. Its functional value is supported by ablation and reparameterization experiments, but it is not an intrinsic shared subspace. Likewise, passing the reported likelihood-ratio and black-box attacks does not provide a certified deletion guarantee, and low ASR cannot rule out every trigger-related behavior. These limitations motivate function-space decompositions, stronger per-example privacy audits, unknown-trigger objectives, and certifiable joint forgetting. More broadly, the results suggest that future unlearning evaluations should measure both information retained about requested examples and behavior learned from those examples.

6 Related Work

Machine unlearning and privacy evaluation. Pipeline-based methods prepare for later deletion, while post-hoc methods update an already trained model (Bourtoule et al., 2021). Representative post-hoc approaches include Amnesiac unlearning, SCRUB, SalUn, selective synaptic dampening, Adv-MU, AMUN, and Datamodel Matching (Graves et al., 2021; Kurmanji et al., 2023; Fan et al., 2024; Foster et al., 2024; Di et al., 2025; Ebrahimpour-Boroojeny et al., 2025; Georgiev et al., 2025). These methods primarily target sample-level or counterfactual forgetting, while our setting also requires removing a rule that generalizes to unseen triggered inputs. Population-level membership attacks can overstate unlearning success, so we separate a diagnostic loss-threshold attack from

independent likelihood-ratio and learned black-box audits (Hayes et al., 2025). These audits remain empirical tests rather than formal deletion guarantees.

Backdoor repair. Post-training defenses include Fine-Pruning, Neural Attention Distillation, I-BAU, ANP, RNP, and FT-SAM (Liu et al., 2018; Li et al., 2021a; Zeng et al., 2022; Wu & Wang, 2021; Li et al., 2023; Zhu et al., 2023). They are designed primarily to reduce trigger-dependent attack success, not to make identified poisoned examples indistinguishable from nonmembers. Privacy-oriented unlearning has the complementary limitation that it does not explicitly remove generalized trigger behavior.

Sign consistency and multi-objective optimization. NegMerge retains weight coordinates whose unlearning-task-vector signs agree across hyperparameter configurations (Kim et al., 2025). UNISON instead tests persistent sign agreement between two different objectives across independent batches while preserving their objective-specific residuals. MGDA, PCGrad, and CAGrad modify currently observed gradients when objectives compete (Sener & Koltun, 2018; Yu et al., 2020; Liu et al., 2021). UNISON first estimates cross-batch persistence and then separates shared and objective-specific coordinates. We compare against NegMerge and generic multi-objective alternatives under matched budgets.

7 CONCLUSION

We studied full-information deletion of identified poisoned examples under a known-trigger and restored-label threat model. Privacy-oriented unlearning and backdoor repair often remove different parts of the poisoned data's influence. UNISON combines persistent shared coordinates, objective-specific updates, retain-gradient protection, and retraining-free adaptive weighting. Across the reported image-classification settings, it provides a stronger empirical trade-off among clean accuracy, attack success, and independently audited membership leakage than the included baselines. These results do not establish certified deletion or unknown-trigger defense. They show that corrective deletion should evaluate privacy and learned behavior jointly.

AI USE STATEMENT

Generative AI tools were used to assist with language editing and to improve clarity during manuscript preparation. They were not used to generate, modify, or interpret experimental results, statistical analyses, or figures. All mathematical derivations, implementations, experiments, and claims were verified by the authors. The authors take full responsibility for the final content of this paper.

ETHICS STATEMENT

This work studies the removal of backdoor behavior from poisoned models. Although the described attacks could be misused, all experiments use standard public datasets and controlled poisoning procedures. The method is intended for defensive model maintenance and data-deletion scenarios. No personal or sensitive user data were collected or released.

REPRODUCIBILITY STATEMENT

The experimental protocol, threat model, model architectures, attack configurations, unlearning hyperparameters, evaluation metrics, and random seeds are described in Section 4 and Appendices. Complete per-seed results, source code, and configuration files are available here: `https://anonymous.4open.science/r/unison-F5AC`.

REFERENCES

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *IEEE Symposium on Security and Privacy*, 2021.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 1897–1914, 2022. doi: 10.1109/SP46214.2022.9833649.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. In *International Conference on Learning Representations*, 2019.

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

Zonglin Di, Sixie Yu, Yevgeniy Vorobeychik, and Yang Liu. Adversarial machine unlearning. In *International Conference on Learning Representations*, 2025.

Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1019–1028. PMLR, 2017.

Ali Ebrahimpour-Boroojeny, Hari Sundaram, and Varun Chandrasekaran. Not all wrong is bad: Using adversarial examples for unlearning. In *International Conference on Machine Learning*, 2025.

Rahim Entezari, Hanie Sedghi, Olga Saukh, and Behnam Neyshabur. The role of permutation invariance in linear mode connectivity of neural networks. In *International Conference on Learning Representations*, 2022.

Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. SalUn: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *International Conference on Learning Representations*, 2024.

Jack Foster, Stefan Schoepf, and Alexandra Brintrup. Fast machine unlearning without retraining through selective synaptic dampening. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 12043–12051, 2024.

Kristian Georgiev, Roy Rinberg, Sung Min Park, Shivam Garg, Andrew Ilyas, Aleksander Madry, and Seth Neel. Attribute-to-delete: Machine unlearning via datamodel matching. In *International Conference on Learning Representations*, 2025.

Laura Graves, Vineel Nagisetty, and Vijay Ganesh. Amnesiac machine learning. In *AAAI Conference on Artificial Intelligence*, 2021.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012.

Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017.

Jamie Hayes, Ilia Shumailov, Eleni Triantafillou, Amr Khalifa, and Nicolas Papernot. Inexact unlearning needs more careful evaluations to avoid a false sense of privacy. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pp. 497–519. IEEE, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778. IEEE, 2016. doi: 10.1109/CVPR.2016.90.

Hyo Seo Kim, Dongyoon Han, and Junsuk Choe. NegMerge: Sign-consensual weight merging for machine unlearning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 30067–30085, 2025.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. CS231N course report, Stanford University, 2015. URL https://cs231n.stanford.edu/reports/2015/pdfs/yle_project.pdf.

Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. Neural attention distillation: Erasing backdoor triggers from deep neural networks. In *International Conference on Learning Representations*, 2021a.

Yige Li, Xixiang Lyu, Xingjun Ma, Nodens Koren, Lingjuan Lyu, Bo Li, and Yu-Gang Jiang. Reconstructive neuron pruning for backdoor defense. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19837–19854, 2023.

Yuezun Li, Yiming Li, Baoyuan Wu, Longkang Li, Ran He, and Siwei Lyu. Invisible backdoor attack with sample-specific triggers. In *IEEE International Conference on Computer Vision*, 2021b.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *Research in Attacks Intrusions and Defenses*, 2018.

Tuan Anh Nguyen and Anh Tuan Tran. Wanet- imperceptible warping based backdoor attack. In *International Conference on Learning Representations*, 2021.

Ahmed Salem, Yang Zhang, Mathias Humbert, Mario Fritz, Pascal Berrang, and Michael Backes. ML-Leaks: Model and data independent membership inference attacks and defenses on machine learning models. In *Network and Distributed System Security Symposium*, 2019. doi: 10.14722/ndss.2019.23119.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy*, pp. 3–18. IEEE, 2017. doi: 10.1109/SP.2017.41.

Dongxian Wu and Yisen Wang. Adversarial neuron pruning purifies backdoored deep models. In *Advances in Neural Information Processing Systems*, 2021.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium*, pp. 268–282. IEEE, 2018. doi: 10.1109/CSF.2018.00027.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference*, 2016. doi: 10.5244/C.30.87.

Yi Zeng, Si Chen, Won Park, Z. Morley Mao, Ming Jin, and Ruoxi Jia. Adversarial unlearning of backdoors via implicit hypergradient. In *International Conference on Learning Representations*, 2022.

Yi Zeng, Minzhou Pan, Hoang Anh Just, Lingjuan Lyu, Meikang Qiu, and Ruoxi Jia. Narcissus: A practical clean-label backdoor attack with limited information. In *ACM Conference on Computer and Communications Security*, 2023.

Mingli Zhu, Shaokui Wei, Li Shen, Yanbo Fan, and Baoyuan Wu. Enhancing fine-tuning based backdoor defense with sharpness-aware minimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4466–4477, 2023.

Table 4: Detailed comparison on CIFAR-10 with BadNets and ResNet-18. Values are means and standard deviations over five independently trained models. LT is the diagnostic loss-threshold membership-inference deviation. Relative time is normalized to AMUN.

Method	Clean accuracy ↑	ASR ↓	LT Δ_{MIA} ↓	Relative time ↓
Poisoned model	93.1 ± 0.2	98.4 ± 0.5	0.241 ± 0.012	0.0
Retraining	92.6 ± 0.3	1.7 ± 0.6	0.010 ± 0.004	12.8
AMUN	91.8 ± 0.4	74.6 ± 3.1	0.025 ± 0.006	1.0
RNP	92.0 ± 0.3	4.8 ± 1.0	0.163 ± 0.011	0.7
Equal-weight averaging	88.9 ± 0.6	8.4 ± 1.4	0.048 ± 0.008	1.0
Tuned weighted sum	88.3 ± 0.7	7.1 ± 1.2	0.041 ± 0.008	1.1
Alternating updates	89.1 ± 0.5	5.6 ± 0.9	0.034 ± 0.007	1.2
PCGrad	89.4 ± 0.5	5.4 ± 1.0	0.032 ± 0.006	1.3
CAGrad	89.9 ± 0.4	5.0 ± 0.9	0.029 ± 0.005	1.4
MGDA	89.7 ± 0.5	4.7 ± 0.8	0.030 ± 0.005	1.5
UNISON	$\mathbf{91.9 \pm 0.3}$	$\mathbf{3.2 \pm 0.8}$	$\mathbf{0.017 \pm 0.004}$	**1.3**

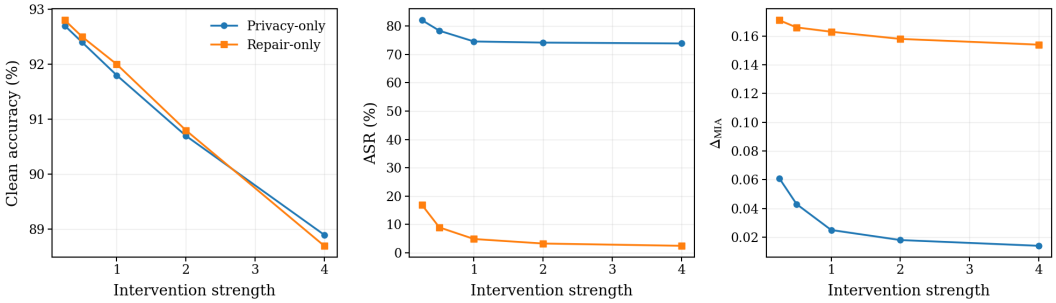


Figure 3: Effect of intervention strength on clean accuracy, attack success rate, and membership-inference deviation. Increasing a single-objective intervention eventually reduces clean accuracy without removing both effects.

A Additional Main-Setting Results

Table 4 gives the complete main-setting comparison. Table 5 reports dataset-level macro-averages. Each macro-average is computed by first averaging across five training seeds for each dataset–attack pair and then taking an unweighted mean over the four attacks.

A.1 Intervention Strength

Figure 3 shows that stronger single-objective updates do not reliably remove both effects. Privacy-oriented updates primarily reduce membership leakage, while repair-oriented updates primarily reduce ASR. At high intervention strength, both approaches reduce clean accuracy.

B Robustness, Sensitivity, and Efficiency

B.1 Violations of the Full-Information Assumptions

Table 6 evaluates incomplete poison recall, false-positive deletions, noisy restored labels, and reduced retain-data access. It does not evaluate inaccurate or unknown trigger transformations.

At 75% recall or precision, the method retains most of its clean utility and substantially reduces ASR relative to the poisoned model. Lower recall leaves residual behavior, while false positives direct deletion updates toward clean examples. Larger retain subsets and accurate restored labels improve reliability after imperfect poison identification.

Table 5: Macro-averages over BadNets, WaNet, ISSBA, and Narcissus for each dataset. LR and BB denote likelihood-ratio and learned black-box Δ_{MIA}, respectively.

Dataset	Method	Clean accuracy ↑	ASR ↓	LR ↓	BB ↓
CIFAR-10	Poisoned model	92.7	96.6	0.228	0.241
	Retraining	92.3	1.8	0.010	0.012
	SCRUB+R	91.4	70.2	0.027	0.034
	AMUN	91.7	63.5	0.023	0.030
	FT-SAM	91.9	5.6	0.146	0.159
	RNP	91.6	4.5	0.132	0.147
	CAGrad	89.4	5.9	0.035	0.042
	UNISON	**91.5**	**4.8**	**0.026**	**0.029**
CIFAR-100	Poisoned model	75.0	94.8	0.198	0.214
	Retraining	74.6	0.7	0.010	0.012
	SCRUB+R	73.4	73.1	0.032	0.040
	AMUN	73.8	66.7	0.027	0.035
	FT-SAM	73.9	6.4	0.133	0.148
	RNP	73.5	5.0	0.121	0.138
	CAGrad	71.9	6.8	0.041	0.049
	UNISON	**73.8**	**5.4**	**0.029**	**0.033**
Tiny ImageNet	Poisoned model	59.6	93.4	0.181	0.199
	Retraining	58.9	0.6	0.012	0.014
	SCRUB+R	57.5	75.0	0.037	0.046
	AMUN	57.9	68.4	0.032	0.041
	FT-SAM	58.1	7.4	0.121	0.137
	RNP	57.8	4.8	0.111	0.128
	CAGrad	56.4	6.9	0.046	0.055
	UNISON	**58.3**	**5.0**	**0.033**	**0.037**
ImageNet-1K	Poisoned model	75.6	91.8	0.159	0.177
	Retraining	75.2	0.2	0.011	0.013
	SCRUB+R	74.0	77.2	0.036	0.045
	AMUN	74.3	70.5	0.031	0.040
	FT-SAM	74.5	8.1	0.108	0.124
	RNP	74.1	5.3	0.099	0.117
	CAGrad	72.9	7.6	0.044	0.052
	UNISON	**74.8**	**5.6**	**0.031**	**0.035**

Table 6: Robustness on CIFAR-10 with BadNets. LR and BB denote likelihood-ratio and learned black-box Δ_{MIA}. A seed is successful if clean accuracy is at least 90%, ASR is at most 10%, and both LR and BB are at most 0.05.

Violation	Level	Clean accuracy ↑	ASR ↓	LR ↓	BB ↓	Successes
Default	Perfect information	91.9 ± 0.3	3.2 ± 0.8	0.021 ± 0.005	0.024 ± 0.006	$5/5$
Recall	75%	91.6 ± 0.4	8.9 ± 1.8	0.034 ± 0.007	0.039 ± 0.008	$4/5$
Recall	50%	91.3 ± 0.5	17.6 ± 2.9	0.058 ± 0.010	0.066 ± 0.012	$2/5$
Recall	25%	90.9 ± 0.7	39.2 ± 4.8	0.096 ± 0.016	0.109 ± 0.019	$0/5$
Precision	75%	91.0 ± 0.6	10.8 ± 2.1	0.038 ± 0.008	0.044 ± 0.009	$3/5$
Precision	50%	89.8 ± 0.8	18.6 ± 3.0	0.061 ± 0.011	0.070 ± 0.013	$1/5$
Precision	25%	87.5 ± 1.1	30.4 ± 4.2	0.084 ± 0.015	0.097 ± 0.017	$0/5$
Label noise	10%	91.5 ± 0.4	5.6 ± 1.2	0.027 ± 0.006	0.031 ± 0.007	$4/5$
Label noise	25%	91.0 ± 0.6	10.2 ± 2.0	0.039 ± 0.008	0.045 ± 0.009	$2/5$
Label noise	50%	90.2 ± 0.8	21.5 ± 3.5	0.063 ± 0.012	0.072 ± 0.014	$0/5$
Retain fraction	1%	89.9 ± 0.9	9.6 ± 1.9	0.045 ± 0.009	0.052 ± 0.010	$1/5$
Retain fraction	5%	91.0 ± 0.5	5.5 ± 1.1	0.028 ± 0.006	0.032 ± 0.007	$4/5$
Retain fraction	25%	92.0 ± 0.3	2.9 ± 0.7	0.019 ± 0.004	0.022 ± 0.005	$5/5$

B.2 SENSITIVITY AND REPARAMETERIZATION

The default configuration uses a 10% active fraction, $\rho = 0.75$, $m = 12$, and $T = 0.25$. Lower ρ selects more coordinates and reduces clean accuracy. Higher ρ omits useful shared updates. Increasing m beyond 12 provides little improvement relative to its additional cost.

The transformations confirm that the coordinate mask is not parameterization-invariant. The results support a narrower claim: stable coordinates are useful under the trained parameterization, not an intrinsic decomposition of function space.

Table 7: Sensitivity analysis on CIFAR-10 with BadNets. LR and BB denote likelihood-ratio and learned black-box Δ_{MIA}, respectively. Bold values indicate the default configuration.

Factor	Value	Selected fraction	Clean accuracy ↑	ASR ↓	LR ↓	BB ↓
Active fraction	1%	0.8%	91.5 ± 0.4	8.8 ± 1.6	0.034 ± 0.007	0.038 ± 0.008
Active fraction	5%	3.9%	91.8 ± 0.3	4.6 ± 1.0	0.025 ± 0.006	0.029 ± 0.007
Active fraction	**10%**	6.8%	91.9 ± 0.3	3.2 ± 0.8	0.021 ± 0.005	0.024 ± 0.006
Active fraction	20%	13.2%	90.7 ± 0.6	3.6 ± 0.9	0.019 ± 0.005	0.023 ± 0.006
Persistence ρ	0.50	13.4%	90.3 ± 0.7	2.8 ± 0.8	0.015 ± 0.004	0.018 ± 0.005
Persistence ρ	**0.75**	6.8%	91.9 ± 0.3	3.2 ± 0.8	0.021 ± 0.005	0.024 ± 0.006
Persistence ρ	0.90	2.7%	91.6 ± 0.4	6.6 ± 1.4	0.027 ± 0.006	0.031 ± 0.007
Gradient estimates m	4	5.9%	90.9 ± 0.5	6.7 ± 1.3	0.029 ± 0.006	0.033 ± 0.007
Gradient estimates m	8	6.4%	91.5 ± 0.4	4.3 ± 0.9	0.024 ± 0.005	0.027 ± 0.006
Gradient estimates m	**12**	6.8%	91.9 ± 0.3	3.2 ± 0.8	0.021 ± 0.005	0.024 ± 0.006
Gradient estimates m	20	6.9%	91.8 ± 0.3	3.0 ± 0.7	0.020 ± 0.005	0.023 ± 0.006
Temperature T	0.10	6.8%	91.1 ± 0.5	3.8 ± 0.9	0.020 ± 0.005	0.024 ± 0.006
Temperature T	**0.25**	6.8%	91.9 ± 0.3	3.2 ± 0.8	0.021 ± 0.005	0.024 ± 0.006
Temperature T	0.50	6.8%	91.6 ± 0.4	4.5 ± 1.0	0.025 ± 0.006	0.029 ± 0.007

Table 8: Parameterization and alternative-construction analysis on CIFAR-10 with BadNets. Logit error is measured before unlearning. LR and BB denote likelihood-ratio and learned black-box Δ_{MIA}, respectively.

Setting	Max logit error ↓	Mask overlap	Clean accuracy ↑	ASR ↓	LR ↓	BB ↓
None	0.0	1.00	91.9 ± 0.3	3.2 ± 0.8	0.021 ± 0.005	0.024 ± 0.006
Channel permutation	3.1×10^{-6}	0.41 ± 0.05	91.2 ± 0.5	5.1 ± 1.1	0.028 ± 0.006	0.032 ± 0.007
Positive rescaling	4.4×10^{-6}	0.36 ± 0.06	90.9 ± 0.6	6.4 ± 1.3	0.033 ± 0.007	0.039 ± 0.008
Tensor-level mask	—	0.58 ± 0.04	91.4 ± 0.4	4.4 ± 1.0	0.025 ± 0.006	0.029 ± 0.007
Low-rank subspace	—	0.62 ± 0.05	91.6 ± 0.4	4.0 ± 0.9	0.024 ± 0.005	0.028 ± 0.006

B.3 Efficiency

Runtime includes stability estimation, validation-gap evaluation, and all unlearning updates. UNISON requires more gradient computation than a single-purpose update but remains approximately ten times faster than retraining in this setting.

C Gradient Structure and Additional Privacy Results

The layerwise analysis in Table 10 shows increasing persistent agreement in later stages. Tables 11 and 12 separate residual membership leakage from trigger-pattern detection. Restored-clean inputs consistently have lower Δ_{MIA}, while Narcissus remains the hardest attack.

D Paired Comparisons

Table 13 reports seed-matched comparisons in the main CIFAR-10 and BadNets setting. The small LR difference from the dedicated privacy baseline is less decisive because that baseline directly optimizes privacy.

E Threat Model and Objective Definitions

The attacker poisons between 0.5% and 5.0% of the training set without modifying the training procedure. BadNets, WaNet, and ISSBA use an attacker-chosen target label. Narcissus uses valid clean labels. The defender receives the trained parameters and the poisoned examples selected for deletion. The main experiment assumes perfect poison identification, knowledge of the trigger transformation, clean labels for the identified examples, and a clean retain subset equal to 10% of the original retain set. The defender does not access the full retain set and does not retrain the model.

Table 9: Efficiency on CIFAR-10 with BadNets using one NVIDIA A100 80GB GPU. The cost of training the original poisoned model is excluded.

Method	Wall time (min) ↓	GPU-hours ↓	Peak memory (GB) ↓	Forward passes ↓	Backward passes ↓
Retraining	54.9 ± 1.2	0.915	8.6	1.56×10^6	0.78×10^6
AMUN	4.3 ± 0.2	0.072	7.8	0.12×10^6	0.06×10^6
RNP	3.0 ± 0.1	0.050	7.4	0.09×10^6	0.04×10^6
Tuned weighted sum	4.7 ± 0.2	0.078	8.2	0.13×10^6	0.07×10^6
PCGrad	5.5 ± 0.3	0.092	9.1	0.13×10^6	0.12×10^6
CAGrad	6.0 ± 0.3	0.100	9.5	0.14×10^6	0.13×10^6
MGDA	6.4 ± 0.3	0.107	9.8	0.14×10^6	0.14×10^6
UNISON	5.6 ± 0.3	0.093	10.2	0.15×10^6	0.13×10^6

Table 10: Layerwise gradient structure averaged across evaluated settings.

Layer group	Cosine similarity	Stable shared fraction
Stem	0.043 ± 0.016	$2.1 \pm 0.5\%$
Residual stage 1	0.051 ± 0.017	$3.4 \pm 0.6\%$
Residual stage 2	0.060 ± 0.018	$5.2 \pm 0.7\%$
Residual stage 3	0.071 ± 0.019	$7.1 \pm 0.9\%$
Final stage	0.079 ± 0.020	$8.7 \pm 1.0\%$

The poison-matched nonmember calibration set has the same size as the forget set. Membership attack training, calibration, and test examples are disjoint. The privacy objective is the MMD loss in Equation 1, using an RBF kernel whose bandwidth is the median pairwise distance between audit scores. The behavioral loss in Equation 2 applies the known trigger to clean retain inputs and restores their original labels. The retain objective is standard cross-entropy on unmodified retain inputs.

F Algorithm and Experimental Details

For each unlearning epoch, we sample 12 triples of forget, nonmember, and retain batches. We normalize each privacy and behavioral direction within its layer. A coordinate is active when its magnitude is in the top 10% of the corresponding layer and objective. It enters the shared mask when both objectives select it with the same sign in at least 9 of the 12 estimates. We average the selected directions, separate shared and objective-specific components, and project each component against a retain gradient when their inner product is negative.

We compute the held-out privacy and behavioral gaps every epoch and update the objective-specific weights with $T = 0.25$. Optimization stops when both gaps are below 0.10, clean validation accuracy drops by more than two points, or the method reaches 20 epochs.

CIFAR-10 and CIFAR-100 use their standard $50{,}000$ training and $10{,}000$ test images. Tiny ImageNet uses $100{,}000$ training and $10{,}000$ validation images. ImageNet-1K uses its standard training and validation splits. The CIFAR models train for 200 epochs, the Tiny ImageNet model for 100 epochs, and the ImageNet model for 90 epochs. All models use SGD with momentum 0.9, weight decay 5×10^{-4}, and cosine learning-rate decay. Initial learning rates are 0.1 for CIFAR and Tiny ImageNet and 0.4 for ImageNet-1K.

The main poisoning ratio is 1%. BadNets uses a fixed square trigger covering approximately 1.6% of the image area. WaNet, ISSBA, and Narcissus use the configurations of their original implementations. Each unlearning baseline is tuned over learning rates $\{10^{-4}, 5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}\}$ and budgets of 5, 10, and 20 epochs. Runtime is measured on one NVIDIA A100 80GB GPU after one warmup run.

Table 11: Triggered and restored-clean membership evaluation on CIFAR-10. LT, LR, and BB report Δ_{MIA}.

Attack	Input view	LT	LR	BB
BadNets	Triggered	0.017 ± 0.004	0.021 ± 0.005	0.024 ± 0.006
BadNets	Restored clean	0.013 ± 0.003	0.016 ± 0.004	0.018 ± 0.005
WaNet	Triggered	0.019 ± 0.005	0.024 ± 0.006	0.026 ± 0.006
WaNet	Restored clean	0.015 ± 0.004	0.019 ± 0.005	0.021 ± 0.005
ISSBA	Triggered	0.022 ± 0.005	0.027 ± 0.006	0.030 ± 0.007
ISSBA	Restored clean	0.018 ± 0.004	0.022 ± 0.005	0.025 ± 0.006
Narcissus	Triggered	0.025 ± 0.006	0.031 ± 0.007	0.034 ± 0.008
Narcissus	Restored clean	0.021 ± 0.005	0.026 ± 0.006	0.029 ± 0.007

Table 12: Low-FPR membership results corresponding to Table 11. Values are TPR percentages.

Attack	Input view	LR TPR@1%	BB TPR@1%	BB TPR@0.1%
BadNets	Triggered	2.4 ± 0.7	2.8 ± 0.8	0.4 ± 0.2
BadNets	Restored clean	1.8 ± 0.6	2.1 ± 0.7	0.3 ± 0.1
WaNet	Triggered	2.8 ± 0.8	3.0 ± 0.8	0.5 ± 0.2
WaNet	Restored clean	2.0 ± 0.6	2.3 ± 0.7	0.4 ± 0.2
ISSBA	Triggered	3.2 ± 0.9	3.6 ± 1.0	0.7 ± 0.3
ISSBA	Restored clean	2.5 ± 0.8	2.8 ± 0.8	0.5 ± 0.2
Narcissus	Triggered	3.9 ± 1.1	4.3 ± 1.2	0.9 ± 0.4
Narcissus	Restored clean	3.0 ± 0.9	3.4 ± 1.0	0.6 ± 0.3

Table 13: Paired comparisons on CIFAR-10 with BadNets. Differences are UNISON minus the named baseline. Negative ASR and LR differences favor UNISON.

Baseline	Δ clean accuracy	Δ ASR	Δ LR
AMUN	$+0.1$ $[-0.3, 0.5]$	-71.4 $[-74.9, -67.9]$	-0.004 $[-0.010, 0.002]$
RNP	-0.1 $[-0.4, 0.2]$	-1.6 $[-2.8, -0.4]$	-0.142 $[-0.156, -0.128]$
Tuned weighted sum	$+3.6$ $[+2.8, +4.4]$	-3.9 $[-5.3, -2.5]$	-0.020 $[-0.028, -0.012]$
CAGrad	$+2.0$ $[+1.4, +2.6]$	-1.8 $[-3.0, -0.6]$	-0.008 $[-0.015, -0.001]$
MGDA	$+2.2$ $[+1.5, +2.9]$	-1.5 $[-2.6, -0.4]$	-0.009 $[-0.016, -0.002]$