LONG-TAILED LEARNING STRATEGIES MATTER IN MACHINE UNLEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Real-world datasets often exhibit long-tailed distributions, where head groups contain many samples and tail groups contain few. Although long-tailed learning strategies are widely used, how they affect machine unlearning remains unclear. In this paper, we study how such strategies influence unlearning difficulty. We find that strategies improving tail performance often make unlearning harder, producing larger deviations from retraining. Moreover, the same degree of tail exposure does not guarantee similar forgetting behavior. Beyond these findings, we observe domain-specific characteristics in vision and language across CIFAR-10, CIFAR-100, Tiny-ImageNet, and long-tailed variants of TOFU and ELUDe, using resampling, augmentation, and loss reweighting under class-wise and random-sample forgetting. Our findings suggest that long-tailed learning strategies should be carefully considered when comparing unlearning methods.

1 INTRODUCTION

Real-world data often follow long-tailed distributions, where a small number of head groups account for most observations while tail groups contain far fewer examples (Liu et al., 2019; Kang et al., 2019). Such imbalance is also common in privacy-sensitive domains, including medical imaging and financial fraud detection (Tschandl et al., 2018; Dal Pozzolo et al., 2014). Training on these distributions can bias models toward head groups and reduce tail performance, motivating strategies for long-tailed learning such as resampling, augmentation, and loss reweighting (Cui et al., 2019; Kang et al., 2019).

However, how strategies for long-tailed learning affect subsequent unlearning remains unclear, especially when tail or head data are later targeted for forgetting. Prior work suggests that learning-time choices, including data augmentation and knowledge encoding, can influence privacy and unlearning behavior (Kaya & Dumitras, 2021; Falcao & Cordeiro, 2025; Wu et al., 2026). This issue is relevant in long-tailed settings, where tail examples are often resampled, augmented, or assigned larger loss weights to improve their learning. If these examples are later removed, the same mechanisms that strengthened their influence during training may also affect how easily that influence can be erased. This leads to a natural question: *how do long-tailed learning strategies affect subsequent unlearning performance?*

To answer this question, we conduct a systematic empirical study of more than 2,800 trained, retrained, and unlearned models across vision and language, spanning 11 long-tailed learning strategy settings and five datasets and benchmarks. In vision, we evaluate seven strategies on CIFAR-10, CIFAR-100, and Tiny-ImageNet under class-wise and random-sample forgetting with six unlearning methods; in language, we construct long-tailed variants of TOFU and ELUDe and evaluate four strategies with five unlearning methods. Across both domains, we find that stronger tail learning tends to be followed by larger deviations from matched retraining, while the same amount of tail exposure can still yield substantially different unlearning behavior. We further find that severe imbalance amplifies strategy-dependent forgetting in vision, whereas in Large language models (LLMs), learning strategies affect both the effectiveness of unlearning methods and whether deviations from retraining extend from seen to held-out examples. These findings show that subsequent unlearning depends on how the forget data were learned, suggesting that long-tailed learning strategies and training-data distributions should be explicitly considered when evaluating and comparing unlearning methods.

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.

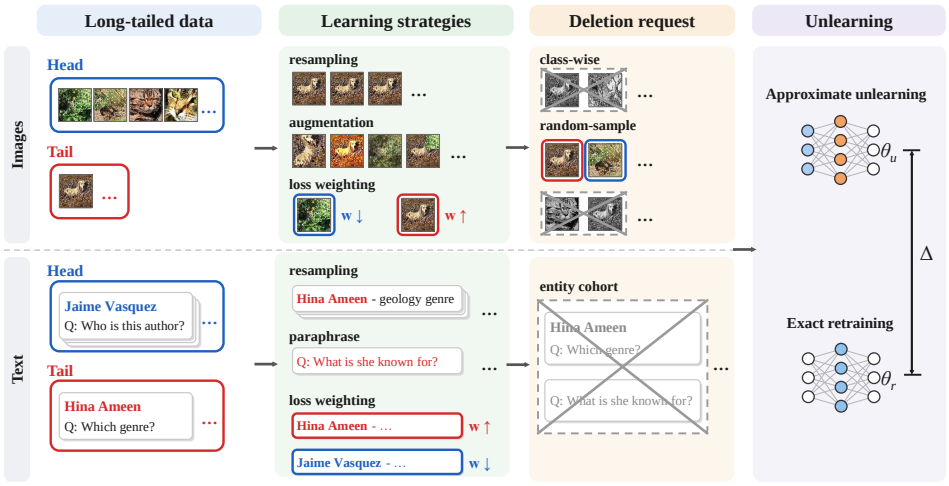


Figure 1: Overview of our experimental design. We evaluate how long-tailed learning strategies affect subsequent unlearning across forgetting settings, using strategy-matched exact retraining as the reference and evaluating performance separately on head and tail groups.

2 Related Works

2.1 Long-Tailed Learning Strategies

Long-tailed imbalance can degrade performance on underrepresented classes or groups (Liu et al., 2019; Henning et al., 2023). Common remedies include oversampling, loss reweighting, and data augmentation. ROS increases tail exposure by repeatedly sampling minority examples, while reweighting methods such as CB Loss adjust class contributions to the objective (Buda et al., 2018; Kang et al., 2019; Cui et al., 2019). Augmentation instead increases input diversity: standard image transformations include random cropping, flipping, rotation, and ColorJitter (He et al., 2016; Shorten & Khoshgoftaar, 2019; Chen et al., 2020), while Mixup and CutMix construct mixed examples (Zhang et al., 2017; Yun et al., 2019). In language, paraphrasing similarly introduces lexical or syntactic variation while preserving semantics (Feng et al., 2021). Prior work further shows that augmentation choices matter under imbalance: MetaSAug targets minority classes with learned semantic transformations (Li et al., 2021), and CUDA finds that optimal augmentation strength can vary across classes (Ahn et al., 2023). Such choices can also affect properties beyond predictive performance; Kaya & Dumitras (2021) show that improved accuracy under augmentation does not necessarily imply reduced membership leakage.

2.2 Machine Unlearning

Machine unlearning aims to remove the influence of designated training data while preserving knowledge from the remaining data (Cao & Yang, 2015). Exact retraining without the forget set provides a gold-standard reference but is often impractical, motivating approximate methods that update a pretrained model (Golatkar et al., 2020). Common approaches include fine-tuning on retained data, forget-set ascent methods such as NegGrad and GradAscent, GradDiff, and Random Label (Golatkar et al., 2020), as well as more targeted methods such as SCRUB (Kurmanji et al., 2023), SalUn (Fan et al., 2024), and UAM (Kim et al., 2026). For LLMs, NPO and SimNPO suppress forget-example likelihood, while DPO has been adapted to favor alternative responses over forget-associated outputs (Zhang et al., 2024; Fan et al., 2026; Rafailov et al., 2023; Maini et al., 2024).

Recent work has begun to examine how training-time choices shape subsequent unlearning. Wu et al. (2026) show that training-data encoding affects LLM unlearning, while Falcao & Cordeiro (2025) show that augmentation choices can reduce the gap between approximate unlearning and retraining on CIFAR-10 and CIFAR-100. Our work extends this direction by systematically studying how long-tailed learning strategies affect subsequent unlearning across vision and language.

3 EXPERIMENTAL METHODOLOGY

In this section, we develop an experimental framework to investigate how long-tailed learning strategies shape subsequent machine unlearning. Since the training and unlearning pipelines differ across domains, we present the vision and language settings separately. Full details on training configurations, benchmark construction, and robustness evaluations are provided in Appendix A.

Notation. Let $\mathcal{S}$ and $\mathcal{D}$ denote the training and test sets, respectively. Let $\mathcal{S}_f$ and $\mathcal{S}_r = \mathcal{S} \setminus \mathcal{S}_f$ denote the forget and retain training sets. We denote the parameters of the corresponding retraining reference by θ_r and those of the approximately unlearned model by θ_u. The retraining reference is defined separately for each setting below. Superscripts or subscripts H and T indicate head and tail groups, respectively. The imbalance factor is defined as $\mathrm{IF} = n_H/n_T$, where the underlying unit is a class in vision and an author or entity in the language setting. Throughout the paper, Δ measures the discrepancy between an unlearned model and its corresponding retraining model.

3.1 IMAGE CLASSIFICATION UNLEARNING

Data and model. We use CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009), and Tiny-ImageNet (Le et al., 2015), with ResNet-18 (He et al., 2016) as the primary backbone. We consider both class-wise and random-sample forgetting, following common image-unlearning settings (Fan et al., 2024; Kim et al., 2026). For class-wise forgetting, we construct single-tail datasets with $\mathrm{IF} = 10$ and delete the entire designated tail class. We evaluate all ten tail-class assignments for CIFAR-10 and three prespecified assignments each for CIFAR-100 and Tiny-ImageNet. For random-sample forgetting, we construct step-imbalanced datasets (Buda et al., 2018) with $\mathrm{IF} \in \{10, 50, 100\}$, evenly split classes into head and tail groups, and delete 10% of samples from each class using forget-set seeds $\{42, 123, 2024\}$. Low-count $\mathrm{IF} = 100$ cases are detailed in Appendix A.1. We additionally conduct targeted robustness experiments with ViT-S/4 (Dosovitskiy, 2020) (Appendix C.1).

Long-tailed learning strategies. We compare seven strategies: Standard, ROS, Rotation, ColorJitter, Mixup, CutMix, and CB Loss. Standard uses random cropping and horizontal flipping (He et al., 2016). ROS (Buda et al., 2018), Rotation (Shorten & Khoshgoftaar, 2019), and ColorJitter (Chen et al., 2020) are applied only to tail examples, with Rotation and ColorJitter matched in tail exposure. Mixup (Zhang et al., 2017) and CutMix (Yun et al., 2019) are applied globally, whereas CB Loss (Cui et al., 2019) reweights the training objective according to class frequency.

Unlearning and evaluation. We evaluate six approximate unlearning methods: FT, NegGrad, RL (Golatkar et al., 2020), SalUn (Fan et al., 2024), SCRUB (Kurmanji et al., 2023), and UAM (Kim et al., 2026). We use strategy-matched Exact Retraining as the reference for approximate unlearning (Golatkar et al., 2020; Fan et al., 2024). Unlearning hyperparameters are selected under Standard and fixed across learning strategies; details are provided in Appendix D. Before unlearning, we first assess the effectiveness of long-tailed learning strategies by evaluating their performance on the test set. We mainly report overall test accuracy (overall TA), tail test accuracy (tail TA), and head test accuracy (head TA), with group accuracy defined as the unweighted mean of per-class accuracies. Let $\mathrm{Acc}(\theta, \mathcal{A})$ denote the accuracy of model θ on an evaluation set $\mathcal{A}$, and let $\mathcal{D}^g \subseteq \mathcal{D}$ denote the test examples from group $g \in \{H, T\}$.

For class-wise forgetting, the forget set consists of all training examples from one designated tail class. We denote $\Delta_{\mathrm{Tail}} := |\mathrm{Acc}(\theta_r, \mathcal{S}_f) - \mathrm{Acc}(\theta_u, \mathcal{S}_f)|$ as the forgetting discrepancy on the forget set, and $\Delta_{\mathrm{Head}} := \mathrm{Acc}(\theta_r, \mathcal{D}^H) - \mathrm{Acc}(\theta_u, \mathcal{D}^H)$ as the retained-utility discrepancy on the head classes. Smaller Δ_{Tail} indicates closer agreement with Exact Retraining, while Δ_{Head} values closer to zero indicate better preservation of head-class utility.

For random-sample forgetting, let $\mathcal{S}_f^g \subseteq \mathcal{S}^g$ denote the samples in the forget set from group $g \in \{H, T\}$, so that $\mathcal{S}_f = \mathcal{S}_f^H \cup \mathcal{S}_f^T$. We evaluate:

$$\Delta^g_{\mathrm{Forget}} := \left|\mathrm{Acc}(\theta_r; \mathcal{S}_f^g) - \mathrm{Acc}(\theta_u; \mathcal{S}_f^g)\right|, \qquad \Delta^g_{\mathrm{Utility}} := \mathrm{Acc}(\theta_r, \mathcal{D}^g) - \mathrm{Acc}(\theta_u, \mathcal{D}^g).$$

All vision Δ values are reported in percentage points.

3.2 LLM UNLEARNING

Data and model. We study TOFU-LT and ELUDe-LT with Llama 3.1 8B (Grattafiori et al., 2024) as the primary backbone, using IF $= 5$ and seeds $\{42, 123, 2024\}$. TOFU-LT is derived from TOFU (Maini et al., 2024), with the ten authors in the official `forget05` cohort forming the tail group and the remaining 190 authors forming the head group. Each head author contributes 20 QA pairs to supervised fine-tuning (SFT), whereas each tail author contributes four, with the remaining 16 tail QA pairs reserved for held-out evaluation. ELUDe-LT is derived from the target-entity QA pool of ELUDe (Choi et al., 2025). Each assignment contains one tail entity and 19 head entities, with 16 and 80 training QA pairs per entity, respectively, and 100 held-out tail QA pairs. Separate tail entities are used for development and final evaluation. Benchmark construction and the IF $= 10$ extension are detailed in Appendix A.2, with Qwen2.5-7B Base robustness results (Hui et al., 2024) in Appendix C.2.

Long-tailed learning strategies. We compare four strategies: Standard, ROS, Loss Weighting, and Paraphrase. Standard applies no additional intervention. ROS follows oversampling approaches (Buda et al., 2018), Loss Weighting follows class-reweighting approaches (Kang et al., 2019), and Paraphrase uses human-reviewed `GPT-5.6 Sol`-generated variants following paraphrasing-based augmentation (Feng et al., 2021). ROS and Paraphrase are exposure-matched. If a source QA is selected for unlearning, all derived instances are included in the same forget set. Detailed SFT and strategy settings are provided in Appendix A.2.

Unlearning and evaluation. We evaluate five approximate unlearning methods: GradAscent and GradDiff (Maini et al., 2024), NPO (Zhang et al., 2024), SimNPO (Fan et al., 2026), and DPO (Rafailov et al., 2023). Hyperparameters are selected under Standard and fixed across learning strategies; for ELUDe-LT, selection uses the development tail assignment before transfer to the final assignment. Details are provided in Appendix D. The forget set contains the SFT-observed tail source QA. Matched retraining repeats SFT from the same base model after removing all training instances derived from $\mathcal{S}_f$ (Maini et al., 2024; Choi et al., 2025). This removes all SFT exposure to the fictitious TOFU-LT tail authors, whereas pretrained knowledge of the real-world ELUDe-LT entities may remain.

Before unlearning, we report Seen Tail Probability and Held-out Tail Probability. Following prior LLM unlearning benchmarks (Maini et al., 2024; Choi et al., 2025; Lin, 2004), we evaluate forgetting using answer Probability and ROUGE-L. For $m \in \{\text{Probability}, \text{ROUGE-L}\}$, let $m_T(\theta)$ denote metric m evaluated over tail QA and define $\Delta_{\text{Tail},m} := |m_T(\theta_u) - m_T(\theta_r)|$. These discrepancies are Tail Probability Gap and Tail ROUGE-L Gap, computed jointly over seen and held-out tail QA unless otherwise specified; split-specific analyses report Seen Tail Probability Gap and Held-out Tail Probability Gap. For retained head utility, let $m_H(\theta)$ denote metric m on the benchmark-specific head evaluation set and define $\Delta_{\text{Head},m} := m_H(\theta_r) - m_H(\theta_u)$. We additionally report Model Utility and MIA Separation, where $\text{MIA Separation} := |\text{AUC} - 0.5|$ using negative per-token answer NLL as the attack score (Yeom et al., 2018); lower values indicate weaker membership distinguishability. Benchmark-specific Model Utility components and MIA member/non-member sets are detailed in Appendix A.2.

4 EXPERIMENTAL RESULTS

As tail performance improves, subsequent unlearning tends to become harder. Table 1 summarizes pre-unlearning tail performance and the corresponding unlearning discrepancies across vision and LLMs. Most long-tailed learning strategies improve tail performance before unlearning, but these gains are often accompanied by larger deviations from matched retraining.

In image classification unlearning, ColorJitter, CB Loss, and CutMix improve Tail TA over Standard across all three datasets, yet do not consistently reduce Δ_{Tail}. ColorJitter and CB Loss yield larger Δ_{Tail} on all three datasets, while CutMix remains comparable to Standard on CIFAR-10 and Tiny-ImageNet but is larger on CIFAR-100. ROS shows a similar pattern on CIFAR-10 and CIFAR-100, and matches Standard in Tail TA on Tiny-ImageNet while still yielding a larger Δ_{Tail}. By contrast, Rotation lowers Tail TA across all three datasets and generally remains close to Standard in Δ_{Tail}.

Table 1: Pre-unlearning tail performance and unlearning performance. Higher Tail TA and Seen Tail Probability indicate better pre-unlearning tail performance, while lower Δ_{Tail} and Tail Probability Gap indicate closer agreement with the corresponding retraining reference after unlearning.

(a) Vision: class-wise forgetting

Strategy	CIFAR-10		CIFAR-100		Tiny-ImageNet	
	Tail TA	Δ_{Tail}	Tail TA	Δ_{Tail}	Tail TA	Δ_{Tail}
Standard	72.52 ± 12.79	4.62 ± 3.31	27.67 ± 17.47	7.44 ± 6.95	31.33 ± 9.02	0.11 ± 0.19
ROS	73.24 ± 13.48	9.05 ± 7.31	32.33 ± 18.82	22.22 ± 3.02	31.33 ± 8.08	5.11 ± 2.83
Rotation	70.95 ± 11.58	3.02 ± 2.09	23.33 ± 10.97	8.89 ± 3.56	22.67 ± 8.08	0.67 ± 1.15
ColorJitter	75.93 ± 11.75	4.72 ± 3.75	31.33 ± 15.28	9.56 ± 2.22	32.67 ± 8.08	3.11 ± 3.72
Mixup	74.05 ± 12.65	4.46 ± 3.55	32.33 ± 15.31	5.44 ± 1.35	11.33 ± 4.62	0.56 ± 0.69
CutMix	76.57 ± 12.49	4.64 ± 5.18	36.67 ± 18.72	9.44 ± 2.91	32.00 ± 9.17	0.11 ± 0.19
CB Loss	76.40 ± 11.96	10.26 ± 6.48	35.00 ± 20.81	18.00 ± 3.18	42.67 ± 17.93	2.56 ± 2.55

(b) LLMs

Strategy	TOFU-LT		ELUDe-LT	
	Seen Tail Probability	Tail Probability Gap	Seen Tail Probability	Tail Probability Gap
Standard	0.452 ± 0.008	0.015 ± 0.004	0.962 ± 0.005	0.016 ± 0.006
ROS	0.958 ± 0.001	0.104 ± 0.006	0.998 ± 0.002	0.079 ± 0.006
Loss Weighting	0.825 ± 0.011	0.084 ± 0.001	0.986 ± 0.008	0.039 ± 0.009
Paraphrase	0.679 ± 0.003	0.068 ± 0.028	0.998 ± 0.000	0.121 ± 0.022

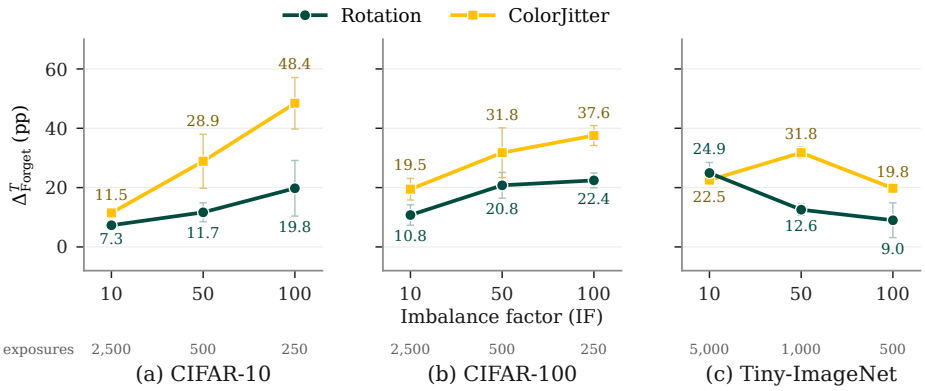


Figure 2: Exposure-matched tail augmentations exhibit different unlearning behavior in vision. Points report mean Δ^T_{Forget} across three forget-set seeds after averaging the six unlearning methods within each seed; error bars denote sample standard deviation across the three random seeds. Lower values indicate closer agreement with retrained model.

In LLM unlearning, ROS, Loss Weighting, and Paraphrase achieve higher Seen Tail Probability than Standard while also producing larger Tail Probability Gaps on both benchmarks at IF $= 5$. The same pattern persists in the targeted IF $= 10$ extension (Appendix Table 18). Overall, improvements in tail performance tend to come with larger unlearning discrepancies relative to matched retraining.

Same exposure does not guarantee the same unlearning behavior. Even when tail data receive the same number of training exposures, subsequent unlearning can differ substantially across learning strategies. In image classification unlearning, Rotation and ColorJitter use the same tail examples with the same exposure, yet exhibit markedly different unlearning behavior. As shown in Figure 2, at IF $\in \{50, 100\}$, Rotation yields a smaller Δ^T_{Forget} than ColorJitter in all six dataset and IF settings across the three vision datasets. The same pattern extends to retained tail utility. In Appendix Tables 9, 10, and 11, we show that Rotation achieves $\Delta^T_{\text{Utility}}$ closer to zero in all six settings.

In LLM unlearning, ROS and Paraphrase likewise receive the same tail exposure but exhibit benchmark-dependent unlearning behavior. Figure 3 and Appendix Table 16 show that, on TOFU-

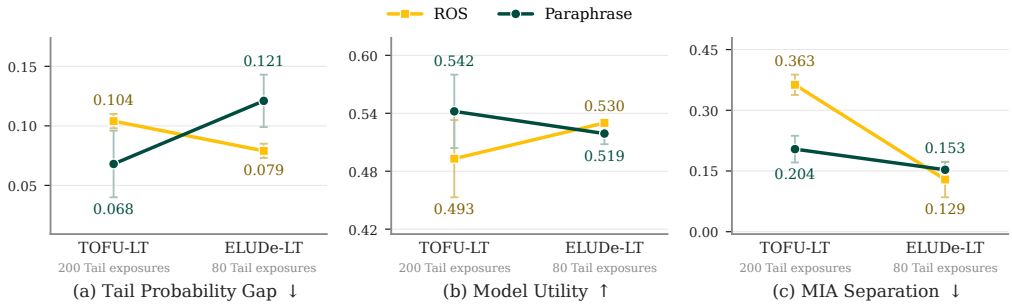


Figure 3: Comparison between ROS and Paraphrase. Points report means across seeds after averaging the five unlearning methods within each seed; error bars denote sample standard deviation. Lower Tail Probability Gap and MIA Separation, and higher Model Utility, are better.

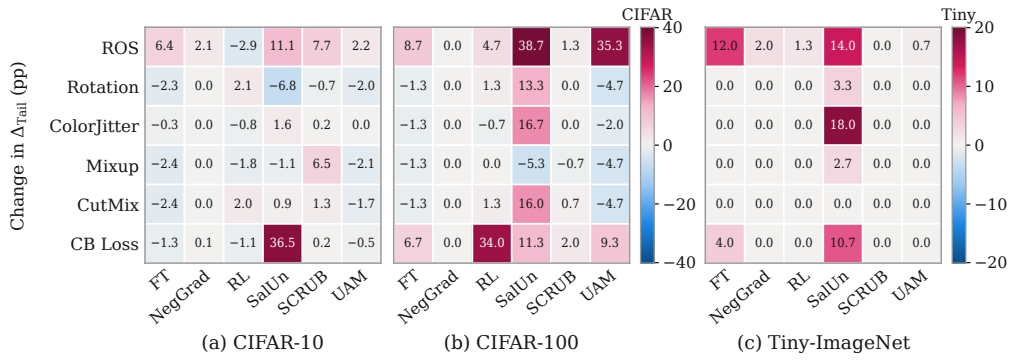


Figure 4: Strategy–method interaction in class-wise forgetting. Each cell reports the paired change in Δ_{Tail} relative to Standard, averaged over the evaluated tail classes. Positive values indicate larger Δ_{Tail} than Standard.

LT, Paraphrase yields a smaller Tail Probability Gap, higher model utility, and lower MIA separation than ROS, whereas this ordering reverses on ELUDe-LT across all three outcomes. In Appendix Table 22, we further show that Qwen2.5-7B Base reproduces the same benchmark-dependent reversal. These results show that the number of training exposures alone is insufficient to determine how a long-tailed learning strategy affects subsequent unlearning.

4.1 Unlearning in Image Classification

In this subsection, we examine in detail how long-tailed learning strategies affect unlearning difficulty in image classification.

Forgetting difficulty under a long-tail learning strategy is class-dependent. As shown in Figure 4, under class-wise forgetting, strategy–method averages initially suggest that SalUn is particularly sensitive to the preceding learning strategy. Per-class results demonstrate that these larger average gaps are not uniform across tail classes. On CIFAR-10, class 3 is forgotten close to Exact Retraining under Standard but becomes severely under-forgotten under ROS and CB Loss. Class 9 shows the reversed pattern, with substantial under-forgetting under Standard but near-retraining forgetting under Mixup and CutMix. Figure 5 tells us that five of the ten CIFAR-10 tail classes vary by more than 50 pp in Δ_{Tail} across learning strategies. On CIFAR-100, SalUn hyperparameters selected on tail 28 do not transfer reliably to other tails (Appendix Table 26). In Appendix C.1, we show that ViT-S/4 exhibits the same class-dependent behavior, with SalUn remaining sensitive to the deleted tail class.

Severe imbalance amplifies strategy-dependent differences in tail forgetting. Under random-sample forgetting, the effect of the learning strategy on $\Delta^{T}_{\text{Forget}}$ becomes increasingly pronounced

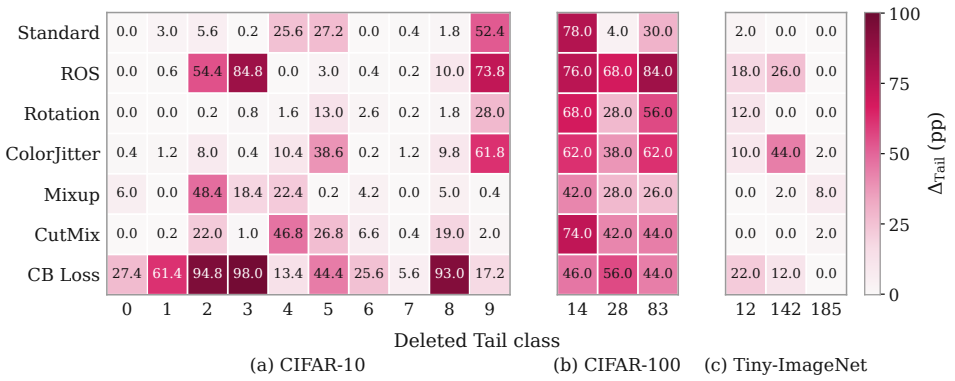


Figure 5: Strategy-dependent tail class under SalUn. Rows denote strategies and columns denote the deleted tail class. Each cell reports Δ_{Tail} in percentage points relative to strategy-matched Exact Retraining. Lower values indicate closer agreement with Exact Retraining.

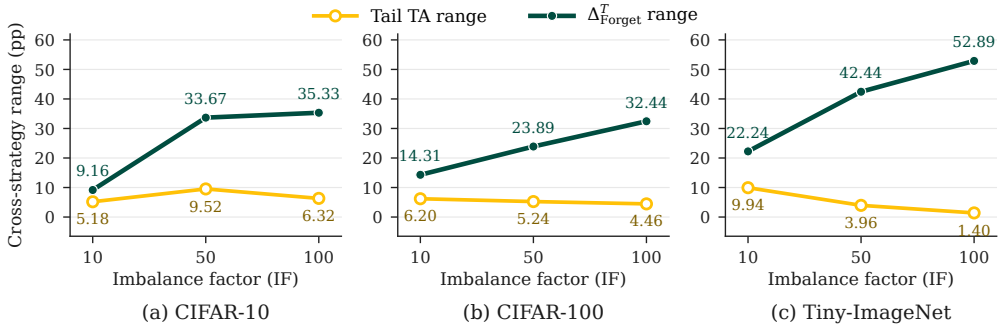


Figure 6: Long-tailed learning strategies under increasing imbalance and random-sample forgetting. It shows the cross-strategy range in pre-unlearning Tail TA and post-unlearning Δ^T_{Forget}. Lower Δ^T_{Forget} indicates closer agreement with the Exact Retraining reference.

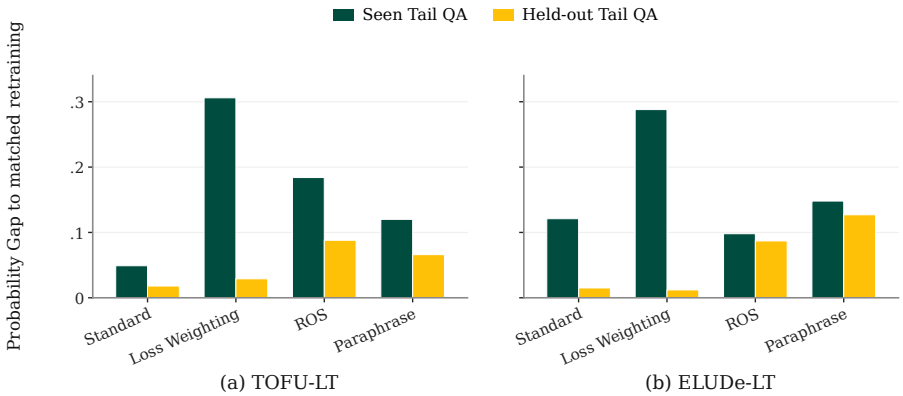
as imbalance grows, even though pre-unlearning Tail TA remains relatively similar across strategies (Figure 6; Appendix Table 13). In other words, severe imbalance amplifies differences in subsequent unlearning difficulty without producing comparably large differences in tail performance before unlearning. This trend is consistent across CIFAR-10, CIFAR-100, and Tiny-ImageNet, and is also reproduced with ViT-S/4 in Appendix C.1.

In Appendix Table 12, we further observe that several long-tailed learning strategies mitigate tail forgetting difficulty relative to Standard under stronger imbalance. Rotation, Mixup, CutMix, and CB Loss generally reduce Δ^T_{Forget}, whereas ROS behaves less consistently across datasets. In particular, ROS becomes substantially harder to unlearn on Tiny-ImageNet under severe imbalance, in contrast to its behavior on the CIFAR datasets. Together, these results show that increasing imbalance does not affect all learning strategies uniformly and can substantially change their relative unlearning behavior.

CB Loss shows a consistent Tail–Head forgetting trade-off. Under random-sample forgetting, the improved tail forgetting performance of CB Loss does not extend uniformly across groups or retained utility (Table 2). Across all nine dataset–IF settings, CB Loss yields a smaller or equal Δ^T_{Forget} than Standard, while Δ^H_{Forget} is larger in every setting. The magnitude of $\Delta^T_{\text{Utility}}$ is also larger in eight of the nine settings. This trade-off is especially pronounced on CIFAR-100 at IF $= 50$. Δ^T_{Forget} decreases from 33.1 to 9.2 pp, whereas Δ^H_{Forget} increases from 7.2 to 39.1 pp and $\Delta^T_{\text{Utility}}$ from 3.0 to 7.1 pp. Tiny-ImageNet shows the same qualitative pattern, including settings where closer Tail forgetting coincides with substantially larger Head forgetting gaps. At IF $= 100$, CB Loss only matches Standard in tail forgetting while still showing a larger Head forgetting gap. Thus,

Table 2: Standard and CB Loss under random-sample forgetting. Results report mean ± sample standard deviation across three forget requests after averaging the six unlearning methods within each request. Lower $\Delta^{T}_{\text{Forget}}$ and $\Delta^{H}_{\text{Forget}}$ indicate closer agreement with the Exact Retraining reference, while $\Delta^{T}_{\text{Utility}}$ values closer to zero indicate closer retained Tail utility.

		CIFAR-10			CIFAR-100			Tiny-ImageNet		
Metric	**Strategy**	IF=10	IF=50	IF=100	IF=10	IF=50	IF=100	IF=10	IF=50	IF=100
$\Delta^{T}_{\text{Forget}}$	Standard	16.5 ± 2.2	45.3 ± 5.3	55.1 ± 4.5	14.9 ± 1.2	33.1 ± 2.3	33.6 ± 4.0	19.4 ± 2.4	37.2 ± 1.0	26.0 ± 2.0
	CB Loss	8.2 ± 1.1	12.7 ± 6.9	20.4 ± 3.7	14.6 ± 2.0	9.2 ± 1.4	10.9 ± 5.6	13.6 ± 5.6	18.9 ± 2.6	26.0 ± 0.3
$\Delta^{H}_{\text{Forget}}$	Standard	2.3 ± 0.2	2.6 ± 0.3	2.7 ± 0.2	10.8 ± 0.5	7.2 ± 2.8	5.8 ± 0.7	17.7 ± 4.5	17.6 ± 0.8	16.5 ± 1.4
	CB Loss	7.6 ± 0.8	8.2 ± 0.4	4.7 ± 0.5	26.8 ± 2.4	39.1 ± 1.8	23.6 ± 4.7	18.9 ± 1.9	34.3 ± 0.8	21.4 ± 1.7
$\Delta^{T}_{\text{Utility}}$	Standard	2.2 ± 1.8	2.5 ± 2.7	0.9 ± 1.5	9.4 ± 1.1	3.0 ± 0.4	1.5 ± 0.3	4.8 ± 1.4	0.3 ± 0.2	0.5 ± 0.1
	CB Loss	7.9 ± 1.3	16.1 ± 1.4	17.3 ± 1.6	15.8 ± 0.3	7.1 ± 0.1	2.3 ± 0.4	5.9 ± 1.8	2.5 ± 0.1	0.4 ± 0.2



(a) TOFU-LT (b) ELUDe-LT

Figure 7: Probability Gap to matched retraining on Seen Tail QA and Held-out Tail QA for TOFU-LT and ELUDe-LT. For each seed, gaps are first averaged over the five unlearning methods; bar heights report the mean across three random seeds. Lower values indicate closer agreement with matched retraining.

CB Loss does not uniformly improve unlearning performance: gains in Tail forgetting consistently coincide with worse Head forgetting performance and, in most settings, larger deviations in retained tail utility.

4.2 Unlearning in LLMs

In this subsection, we analyze how long-tailed learning strategies shape behavior in LLM unlearning.

Learning strategies redistribute the unlearning gap across Seen and Held-out Tail QA. Across both benchmarks, learning strategies differ in whether deviations from matched retraining remain concentrated on Seen Tail QA or extend to Held-out Tail QA from the same tail unit. As shown in Figure 7, this distinction is most evident in the Probability Gap. At IF $= 5$, Loss Weighting exhibits a large Seen Tail gap but a relatively small Held-out Tail gap, whereas ROS and Paraphrase produce substantially larger Held-out gaps. In Appendix Table 17, we observe the same pattern at the method level.

The strategy-level contrast persists at IF $= 10$ and with the Qwen2.5-7B Base model. In Appendix Tables 18 and 22, we observe that Loss Weighting maintains a comparatively small Held-out Tail Probability Gap, whereas ROS and Paraphrase yield larger Held-out gaps on both benchmarks. In Appendix Tables 6 and 7, we further observe that these differences are not explained by pre-unlearning Held-out Tail QA performance. Thus, long-tailed learning strategies affect not only the magnitude of the gap from matched retraining, but also how that gap is distributed across related tail examples.

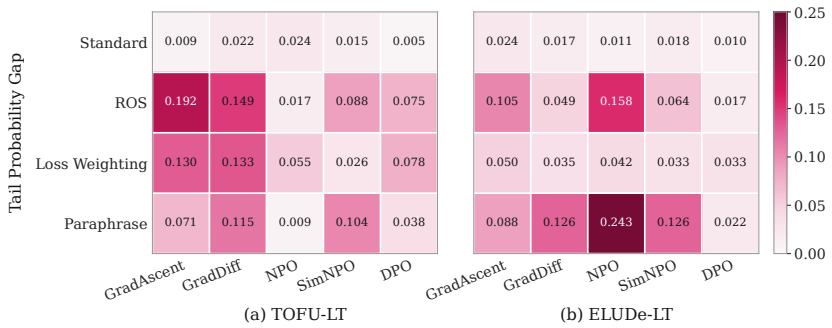


Figure 8: Method-level Tail Probability Gap across long-tailed learning strategies. Rows denote learning strategies and columns denote unlearning methods. Each cell reports the mean Tail Probability Gap over all Tail QA (Seen and Held-out) across three random seeds. Lower values indicate closer agreement with matched retraining.

The long-tailed learning strategy used during training matters for subsequent unlearning method selection. Under Standard training, the five unlearning methods exhibit relatively small differences in Tail Probability Gap, with max–min spreads of 0.019 on TOFU-LT and 0.014 on ELUDe-LT, as shown in Figure 8. In Appendix Tables 14 and 15, we observe that these differences widen substantially under long-tailed learning strategies. Across ROS, Loss Weighting, and Paraphrase, five of the six strategy–benchmark settings yield method spreads 5.6–$15.8\times$ larger than under Standard, with Loss Weighting on ELUDe-LT as the only exception.

The relative behavior of individual unlearning methods also changes across training strategies. On TOFU-LT, NPO has the largest Tail Probability Gap under Standard but the smallest under ROS and Paraphrase, whereas GradAscent has the largest gap under ROS. On ELUDe-LT, NPO shifts from one of the smallest gaps under Standard to the largest under both ROS and Paraphrase. In Appendix Table 23, we observe the same interaction with Qwen2.5-7B Base: every non-Standard setting increases the method spread and alters the relative ordering of methods. These results show that evaluations under Standard training alone can miss substantial interactions between the training strategy and the subsequent unlearning method.

5 Limitations and Future Work

Our controlled design isolates the effect of learning strategy on subsequent unlearning, but does not capture the full complexity of real-world data distributions and deletion requests. Our evaluation also covers a limited range of architectures and tail configurations: additional backbones are examined only in targeted robustness studies, and ELUDe-LT uses a single final tail entity. Future work should extend the analysis to naturally occurring long-tailed data, broader model families, and more diverse tail settings. Finally, our metrics measure empirical agreement with exact retraining but do not provide formal guarantees of complete removal or privacy, motivating stronger privacy-oriented evaluations and certified unlearning analyses.

6 Conclusion

We study how long-tailed learning strategies shape subsequent machine unlearning across vision and language. Across five datasets and benchmarks, we find that improvements in tail performance tend to come with larger deviations from matched retraining, while the same amount of tail exposure does not guarantee similar unlearning behavior. We further observe domain-specific effects: in vision, strategy-dependent tail forgetting becomes more pronounced as imbalance increases, whereas in LLMs, learning strategies affect whether unlearning discrepancies remain on seen examples or extend to held-out examples about the same subject. Together, these findings show that unlearning behavior depends not only on the unlearning method but also on how the deleted data were learned. We therefore argue that long-tailed learning strategies and training-data distributions should be explicitly considered when evaluating and comparing unlearning methods.

AI USE STATEMENT

In this work, we used generative AI tools for generating synthetic data sets, assisting with translation, designing or providing feedback on research methodology or experiments. For the Paraphrase conditions in TOFU-LT and ELUDe-LT, we used `GPT-5.6 Sol` to generate paraphrased training Q–A variants. The generated variants were manually reviewed for consistency with their source Q–A pairs before use, and the corresponding prompts are provided in Appendix E. We have not used generative AI tools for developing theoretical models or conceptual frameworks, proposing hypotheses, implementing methods, interpreting results, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, cleaning and reformatting datasets, or supporting qualitative and thematic data analysis.

Additionally, we used generative AI tools for creating or editing software code, drafting parts of the paper, and editing the paper to improve readability. We have reviewed all AI-assisted work. LLM-generated code and writing was verified and tested for correctness by all authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our experiments do not collect new personal data. TOFU-LT is derived from a benchmark of fictitious authors, while ELUDe-LT is constructed from the existing ELUDe benchmark, whose real-world target entities and associated knowledge are derived from publicly available Wikipedia pages. We use these data only as a controlled testbed for studying entity-level machine unlearning and do not introduce additional private information about the individuals. Our experiments should therefore be interpreted as a methodological evaluation of unlearning rather than as a claim about the privacy status of the underlying individuals.

REPRODUCIBILITY STATEMENT

The experimental design, including dataset construction, long-tailed learning strategies, unlearning settings, retraining references, evaluation metrics, and aggregation procedures, is described in Section 3, with additional LLM construction details in Appendix A.2. Final unlearning hyperparameters for the primary and robustness experiments are reported in Appendix D, and the prompts used to construct the reviewed paraphrase augmentation are provided in Appendix E. We also provide an anonymized supplementary code archive containing the training, retraining, unlearning, and evaluation implementations, frozen experiment inputs and forget-set indices, environment information, and documentation mapping the reported experiments to their corresponding code and execution procedures.

REFERENCES

Sumyeong Ahn, Jongwoo Ko, and Se-Young Yun. Cuda: Curriculum of data augmentation for long-tailed recognition. *arXiv preprint arXiv:2302.05499*, 2023.

Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural networks*, 106:249–259, 2018.

Pengfei Cao, Chenhao Wang, Zhitao He, Hongbang Yuan, Jiachun Li, Yubo Chen, Kang Liu, Jun Zhao, et al. Rwku: Benchmarking real-world knowledge unlearning for large language models. *Advances in Neural Information Processing Systems*, 37:98213–98263, 2024.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE symposium on security and privacy*, pp. 463–480. IEEE, 2015.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.

Minseok Choi, Daniel Rim, Dohyun Lee, and Jaegul Choo. Opt-out: Investigating entity-level unlearning for large language models via optimal transport. *arXiv preprint arXiv:2406.12329*, 2025.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 9260–9269. IEEE, 2019.

Andrea Dal Pozzolo, Olivier Caelen, Yann-Ael Le Borgne, Serge Waterschoot, and Gianluca Bontempi. Learned lessons in credit card fraud detection from a practitioner perspective. *Expert systems with applications*, 41(10):4915–4928, 2014.

Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Andreza MC Falcao and Filipe R Cordeiro. Data augmentation improves machine unlearning. In *2025 38th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pp. 1–2. IEEE, 2025.

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*, volume 2024, pp. 53643–53673, 2024.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for llm unlearning. *Advances in Neural Information Processing Systems*, 38:1540–1567, 2026.

Steven Y Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. A survey of data augmentation approaches for nlp. In *Findings of the association for computational linguistics: ACL-IJCNLP 2021*, pp. 968–988, 2021.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9304–9312, 2020.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Sophie Henning, William Beluch, Alexander Fraser, and Annemarie Friedrich. A survey of methods for addressing class imbalance in deep-learning based natural language processing. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 523–540, 2023.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.

Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019.

Yigitcan Kaya and Tudor Dumitras. When does data augmentation help with membership inference attacks? In *International conference on machine learning*, pp. 5345–5355. PMLR, 2021.

Hoki Kim, Keonwoo Kim, Sungwon Chae, and Sangwon Yoon. Unlearning-aware minimization. *Advances in Neural Information Processing Systems*, 38:93806–93829, 2026.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. *Advances in neural information processing systems*, 36:1957–1987, 2023.

Yann Le, Xuan Yang, et al. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, et al. The wmdp benchmark: Measuring and reducing malicious use with unlearning. *arXiv preprint arXiv:2403.03218*, 2024.

Shuang Li, Kaixiong Gong, Chi Harold Liu, Yulin Wang, Feng Qiao, and Xinjing Cheng. Metasaug: Meta semantic augmentation for long-tailed visual recognition. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5208–5217. IEEE, 2021.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 2532–2541. IEEE, 2019.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C Lipton, and J Zico Kolter. Tofu: A task of fictitious unlearning for llms. *arXiv preprint arXiv:2401.06121*, 2024.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah Smith, and Chiyuan Zhang. Muse: Machine unlearning six-way evaluation for language models. In *International Conference on Learning Representations*, volume 2025, pp. 27797–27818, 2025.

Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):60, 2019.

Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, 5(1): 180161, 2018.

Ruihan Wu, Konstantin Garov, and Kamalika Chaudhuri. Learning-time encoding shapes unlearning in llms. In *International Conference on Learning Representations*, volume 2026, pp. 87040–87067, 2026.

Zhengzhuo Xu, Ruikang Liu, Shuo Yang, Zenghao Chai, and Chun Yuan. Learning imbalanced data with vision transformers. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15793–15803. IEEE, 2023.

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 (CSF)*, pp. 268–282. IEEE, 2018.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6023–6032, 2019.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2404.05868*, 2024.

A Experimental Details

A.1 Vision Experimental Details

For the primary ResNet-18 experiments, both the original models and the Exact Retraining references are trained from scratch for 100 epochs using SGD with an initial learning rate of 0.1, momentum 0.9, weight decay 5×10^{-4}, and batch size 128. The learning rate follows a cosine annealing schedule over the training period. For each learning strategy and unlearning request, Exact Retraining uses the corresponding learning strategy and training protocol on the retained training data. Dataset-construction and model-training seeds are fixed to 42; random-sample forgetting separately uses forget-set seeds $\{42, 123, 2024\}$. Table 3 summarizes the strategy-specific settings. Approximate-unlearning hyperparameters are reported in Appendix D, and the ViT-S/4 configuration is described separately in Appendix C.1.

For class-wise forgetting, CIFAR-10 evaluates all ten class assignments, while CIFAR-100 uses tail classes $\{14, 28, 83\}$ and Tiny-ImageNet uses tail classes $\{12, 142, 185\}$. For random-sample forgetting, the nominal forget-set fraction is 10% per class. At IF $= 100$, each CIFAR-100 tail class contains only five training examples, so selecting one example for the forget set corresponds to 20% of that class. For Tiny-ImageNet at IF $= 100$, we instead select one example from 50 of the 100 tail classes, preserving an aggregate 10% forget-set fraction over the tail group.

Table 3: Long-tailed learning strategies in the ResNet-18 experiments. Tail-only augmentations are applied to tail examples, while head examples receive the Standard pipeline. CB Loss weights class c by $(1-\beta)/(1-\beta^{n_c})$, where n_c is its number of training examples, normalized to mean one over the classes present.

Strategy	**Setting**
Standard	Random crop with 4-pixel padding (32 px on CIFAR, 64 px on Tiny-ImageNet) and horizontal flip
ROS	Each tail class is repeated cyclically up to the size of the largest head class
Rotation	Tail only; random rotation within $\pm 10^{\circ}$; no oversampling
ColorJitter	Tail only; brightness 0.2, contrast 0.2, saturation 0.2, hue 0.1; no oversampling
Mixup	Every batch; $\lambda \sim \mathrm{Beta}(1, 1)$
CutMix	Each batch with probability 0.5; $\lambda \sim \mathrm{Beta}(1, 1)$
CB Loss	$\beta = 0.9999$ for class-wise forgetting and for CIFAR-10 random-sample forgetting; $\beta = 0.99$ at IF $\in \{10, 50\}$ and $\beta = 0.9$ at IF $= 100$ for CIFAR-100 and Tiny-ImageNet random-sample forgetting

A.2 LLM Experimental Details

For TOFU-LT, authors 0–189 form the head group, while authors 190–199, corresponding to the official `forget05` cohort, form the tail group. Each head author contributes 20 QA pairs to supervised fine-tuning (SFT), whereas each tail author contributes four. The remaining 16 QA pairs per tail author are reserved for held-out evaluation, yielding 40 Seen Tail and 160 Held-out Tail QA. The head evaluation set contains 300 training-observed QA pairs from 15 head authors.

The primary Llama 3.1 8B experiments use full-parameter SFT for five epochs with a learning rate of 10^{-5} and an effective batch size of 32. We use seeds $\{42, 123, 2024\}$ while fixing benchmark construction, head–tail assignments, and paraphrase variants across seeds. ROS gives each tail source QA five total exposures, Loss Weighting uses a 5:1 tail-to-head loss-weight ratio, and Paraphrase adds four human-reviewed variants per tail source QA. ROS and Paraphrase are therefore exposure-matched, increasing the number of tail training instances from 40 to 200 in TOFU-LT and from 16 to 80 in ELUDe-LT.

For ELUDe-LT, target entities and QA examples are selected using a deterministic SHA-256 ranking with seed 42. Justin Bieber is used as the development tail entity and Michael Jordan as the final tail entity. For each assignment, the remaining 19 target entities form the head group. Each head entity contributes 80 training and 20 held-out QA pairs, whereas the tail entity contributes 16 training and 100 held-out QA pairs. For the final assignment, the head held-out set contains 380 QA pairs, and

the official target-conditioned neighbor validation and test sets contain 770 QA pairs each. The official retain splits are reserved for collateral evaluation rather than head training.

For TOFU-LT, Model Utility is the harmonic mean of nine set-level scores: Probability, ROUGE-L, and Truth Ratio computed separately on the head evaluation set, Real Authors, and World Facts. For ELUDe-LT, we use the same aggregation over the head held-out set and the official neighbor validation and test sets. For membership inference, TOFU-LT uses Seen Tail QA as members and the official `holdout05` split as non-members, whereas ELUDe-LT uses Seen Tail QA as members and Held-out Tail QA from the same tail entity as non-members.

For the primary IF $= 5$ experiments, strategy-level summaries first average the five unlearning methods within each seed and then report the mean and sample standard deviation across seeds $\{42, 123, 2024\}$. Method-resolved results report the mean and sample standard deviation across the same seeds.

For the targeted Llama IF $= 10$ extension, we retain the same head–tail assignments, head training data, and Held-out Tail and head evaluation sets as in the primary experiments, while reducing the number of tail source QA pairs used for SFT. We select a fixed subset of the IF $= 5$ tail training QA so that the IF $= 10$ source set is nested within the primary source set: two QA pairs per tail author in TOFU-LT, yielding 20 Seen Tail QA in total, and eight QA pairs from the final tail entity in ELUDe-LT. The 160 TOFU-LT and 100 ELUDe-LT Held-out Tail QA remain unchanged, and excluded IF $= 5$ training QA are not added to the held-out evaluation sets.

We retain the same intervention strength as at IF $= 5$. ROS gives each selected tail source QA five total exposures, Loss Weighting retains the 5:1 tail-to-head loss-weight ratio, and Paraphrase reuses the four existing human-reviewed variants of each selected source QA together with the original. ROS and Paraphrase therefore remain exposure-matched, with 100 tail training instances in TOFU-LT and 40 in ELUDe-LT; no additional paraphrases are generated. The extension uses seed 42, all four learning strategies, and GradDiff, NPO, and DPO. We transfer the corresponding benchmark-specific IF $= 5$ unlearning hyperparameters without retuning. Strategy-level IF $= 10$ summaries average these three methods.

A.2.1 Rationale for LLM Benchmark Selection

Our LLM experiments are designed to isolate how learning-time data imbalance and long-tailed learning strategies affect subsequent unlearning. This objective imposes requirements that differ from those of general-purpose unlearning benchmarks. A suitable dataset should provide multiple knowledge units that can be partitioned into head and tail groups, permit direct control over the exposure frequency and representational diversity of those units, and support construction of a strategy-matched retraining reference without the requested data.

MUSE evaluates language-model unlearning along six dimensions, including verbatim memorization, knowledge memorization, privacy leakage, utility preservation, scalability, and sequential unlearning, using book and news corpora (Shi et al., 2025). Its News domain is attractive as a potential real-corpus validation setting. However, our inspection of the MUSE-News data indicated that the available records did not provide sufficiently reliable article-level identifiers or metadata to define units for Head–Tail construction. Moreover, Evaluation QA instances could not be consistently linked to a unique source article, and we observed duplicate or highly overlapping records across the released data.

WMDP evaluates hazardous knowledge in biosecurity, cybersecurity, and chemical security using multiple-choice questions, with unlearning as the mechanism for reducing such knowledge (Li et al., 2024). The benchmark is organized around broad knowledge domains rather than multiple comparable entity-level units. Moreover, the base model may already contain substantial and unknown exposure to these domains from pretraining, making it difficult to construct a controlled Head–Tail imbalance or determine the total exposure of the model to the target knowledge.

RWKU studies the removal of knowledge about 200 real-world individuals without access to the original forget or retain corpora (Cao et al., 2024). Because the base model may already contain unknown and uneven exposure to these individuals, a controlled Head–Tail allocation is difficult to define. The unavailable training corpus additionally prevents construction of a strategy-matched retraining reference.

Table 4: Pre-unlearning performance under single-tail imbalance (IF=10). Results are averaged over ten tail class assignments for CIFAR-10 and three assignments for CIFAR-100 and Tiny-ImageNet.

Strategy	Overall TA	Tail TA	Head TA
CIFAR-10			
Standard	92.80	72.52	95.05
ROS	93.01	73.24	95.21
Rotation	92.70	70.95	95.11
ColorJitter	93.13	75.93	95.04
Mixup	93.20	74.05	95.33
CutMix	**93.92**	**76.57**	**95.85**
CB Loss	92.85	76.40	94.68
CIFAR-100			
Standard	76.65	27.67	77.14
ROS	76.67	32.33	77.12
Rotation	76.51	23.33	77.04
ColorJitter	76.34	31.33	76.79
Mixup	77.68	32.33	78.13
CutMix	**79.12**	**36.67**	**79.55**
CB Loss	76.41	35.00	76.83
Tiny-ImageNet			
Standard	65.75	31.33	65.92
ROS	65.25	31.33	65.42
Rotation	65.36	22.67	65.57
ColorJitter	65.29	32.67	65.45
Mixup	66.37	11.33	66.65
CutMix	**67.60**	32.00	**67.78**
CB Loss	65.58	**42.67**	65.70

B Additional Experimental Results

B.1 Pre-Unlearning Performance

For the class-wise single-tail setting, Table 4 reports the full pre-unlearning Overall TA, Tail TA, and Head TA under the single-tail setting, while Figure 9 reports the corresponding Tail-class variation relative to Standard. For the step-imbalanced random-sample setting, Table 5 reports the full pre-unlearning performance across IFs, and Figure 10 visualizes the corresponding Tail TA.

Tables 6 and 7 report the full three-seed pre-unlearning SFT results for TOFU-LT and ELUDe-LT, respectively.

B.2 Detailed Class-Wise Unlearning Results

Figure 11 reports, for each strategy–method pair, the change in Δ_{Head} relative to Standard on all three datasets. Table 8 reports the full method–strategy results under CIFAR-100 class-wise forgetting, with variation across the three tail classes, and Figures 13 and 14 break these results down by tail class.

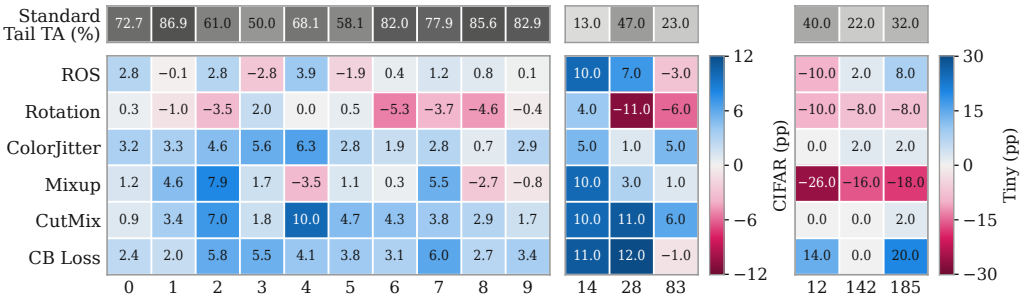
Table 8: Detailed CIFAR-100 class-wise forgetting results. Values are mean $\pm$ sample standard deviation across the three tail classes $\{14, 28, 83\}$. Worst Retained TA is the minimum test accuracy among the 99 retained classes for each tail assignment; the table reports its mean and sample standard deviation across the three tail assignments.

Strategy	Method	Δ_{Tail}	Δ_{Head}	Worst Retained TA
Standard	FT	1.33 ± 1.15	2.50 ± 5.20	45.33 ± 10.97

Strategy	Method	Δ_{Tail}	Δ_{Head}	Worst Retained TA
Standard	NegGrad	0.00 ± 0.00	0.09 ± 0.66	51.67 ± 3.06
Standard	RL	0.67 ± 1.15	0.06 ± 0.46	52.33 ± 2.08
Standard	SalUn	37.33 ± 37.54	6.78 ± 0.81	28.67 ± 13.05
Standard	SCRUB	0.67 ± 1.15	3.80 ± 1.90	36.33 ± 5.13
Standard	UAM	4.67 ± 4.62	0.05 ± 0.52	49.67 ± 4.04
ROS	FT	10.00 ± 11.14	1.66 ± 2.97	49.00 ± 6.00
ROS	NegGrad	0.00 ± 0.00	0.49 ± 0.34	53.00 ± 0.00
ROS	RL	5.33 ± 4.16	0.23 ± 0.36	52.67 ± 2.31
ROS	SalUn	76.00 ± 8.00	5.95 ± 1.09	34.33 ± 3.79
ROS	SCRUB	2.00 ± 0.00	3.89 ± 1.43	38.00 ± 5.29
ROS	UAM	40.00 ± 14.00	0.09 ± 0.35	50.00 ± 2.65
Rotation	FT	0.00 ± 0.00	7.54 ± 7.16	32.00 ± 18.08
Rotation	NegGrad	0.00 ± 0.00	0.32 ± 0.73	52.67 ± 1.15
Rotation	RL	2.00 ± 2.00	0.20 ± 0.70	49.33 ± 3.21
Rotation	SalUn	50.67 ± 20.53	6.77 ± 1.28	31.33 ± 3.79
Rotation	SCRUB	0.67 ± 1.15	3.71 ± 1.76	37.33 ± 5.86
Rotation	UAM	0.00 ± 0.00	-0.02 ± 0.59	49.67 ± 3.21
ColorJitter	FT	0.00 ± 0.00	2.69 ± 4.67	46.67 ± 7.51
ColorJitter	NegGrad	0.00 ± 0.00	0.51 ± 0.13	48.33 ± 1.53
ColorJitter	RL	0.00 ± 0.00	0.42 ± 0.44	50.67 ± 3.51
ColorJitter	SalUn	54.00 ± 13.86	6.60 ± 1.37	32.00 ± 9.17
ColorJitter	SCRUB	0.67 ± 1.15	4.39 ± 1.55	37.00 ± 9.54
ColorJitter	UAM	2.67 ± 1.15	0.17 ± 0.22	46.00 ± 4.58
Mixup	FT	0.00 ± 0.00	2.51 ± 3.21	39.33 ± 11.37
Mixup	NegGrad	0.00 ± 0.00	-0.85 ± 1.01	51.67 ± 2.89
Mixup	RL	0.67 ± 1.15	-1.60 ± 0.51	50.33 ± 4.04
Mixup	SalUn	32.00 ± 8.72	3.43 ± 0.61	37.00 ± 7.94
Mixup	SCRUB	0.00 ± 0.00	2.04 ± 0.68	35.67 ± 5.69
Mixup	UAM	0.00 ± 0.00	-0.96 ± 0.50	48.67 ± 5.69
CutMix	FT	0.00 ± 0.00	5.87 ± 8.59	38.00 ± 19.92
CutMix	NegGrad	0.00 ± 0.00	1.02 ± 1.45	53.00 ± 6.93
CutMix	RL	2.00 ± 3.46	-0.12 ± 0.43	53.33 ± 2.08
CutMix	SalUn	53.33 ± 17.93	4.99 ± 1.39	42.33 ± 4.62
CutMix	SCRUB	1.33 ± 1.15	4.11 ± 2.10	39.67 ± 9.29
CutMix	UAM	0.00 ± 0.00	1.04 ± 0.53	49.67 ± 4.93
CB Loss	FT	8.00 ± 8.00	6.74 ± 12.08	39.67 ± 21.36
CB Loss	NegGrad	0.00 ± 0.00	0.33 ± 0.17	51.33 ± 3.21
CB Loss	RL	34.67 ± 23.18	0.55 ± 0.46	50.33 ± 2.08
CB Loss	SalUn	48.67 ± 6.43	6.56 ± 0.99	32.33 ± 3.06
CB Loss	SCRUB	2.67 ± 3.06	4.35 ± 1.70	35.67 ± 5.86
CB Loss	UAM	14.00 ± 3.46	0.19 ± 0.11	48.33 ± 3.51

B.3 Detailed Random-Sample Unlearning Results

Tables 9, 10, and 11 report all four random-sample metrics for every learning strategy and IF. Each entry averages the six unlearning methods within a forget request and reports the mean and sample standard deviation across the three forget requests. Table 12 collects Δ^T_{Forget} from these tables across the three datasets, and Table 13 summarizes the cross-strategy range in pre-unlearning Tail TA and in Δ^T_{Forget}.



(a) CIFAR-10 (b) CIFAR-100 (c) Tiny-ImageNet

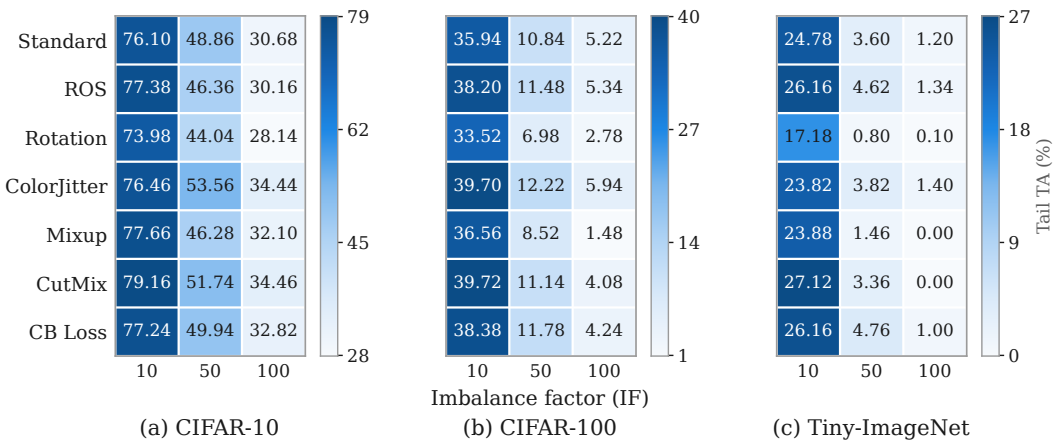
Figure 9: Pre-unlearning tail performance by tail class. The top row reports Tail TA under Standard training, and the bottom row reports the paired change from Standard for each strategy. CIFAR-10 and CIFAR-100 share a symmetric $[-12, 12]$ pp scale, while Tiny-ImageNet uses $[-30, 30]$ pp to preserve within-dataset contrast.

Table 5: Pre-unlearning performance under step imbalance. Bold indicates the highest value within each dataset, IF, and metric.

		CIFAR-10			CIFAR-100			Tiny-ImageNet		
IF	**Strategy**	**Overall TA**	**Tail TA**	**Head TA**	**Overall TA**	**Tail TA**	**Head TA**	**Overall TA**	**Tail TA**	**Head TA**
10	Standard	85.90	76.10	95.70	57.46	35.94	78.98	46.98	24.78	69.18
	ROS	86.52	77.38	95.66	58.68	38.20	79.16	47.53	26.16	68.90
	Rotation	84.81	73.98	95.64	56.86	33.52	80.20	43.48	17.18	69.78
	ColorJitter	85.90	76.46	95.34	59.88	39.70	80.06	46.84	23.82	69.86
	Mixup	86.95	77.66	96.24	58.65	36.56	80.74	46.12	23.88	68.36
	CutMix	**87.76**	**79.16**	**96.36**	**60.57**	**39.72**	**81.42**	**48.70**	**27.12**	**70.28**
	CB Loss	85.65	77.24	94.06	59.24	38.38	80.10	47.32	26.16	68.48
50	Standard	72.15	48.86	95.44	45.62	10.84	80.40	36.74	3.60	69.88
	ROS	71.41	46.36	96.46	45.92	11.48	80.36	36.65	4.62	68.68
	Rotation	69.80	44.04	95.56	44.03	6.98	81.08	35.61	0.80	70.42
	ColorJitter	**74.56**	**53.56**	95.56	46.05	**12.22**	79.88	36.52	3.82	69.22
	Mixup	71.27	46.28	96.26	44.91	8.52	81.30	35.06	1.46	68.66
	CutMix	74.15	51.74	**96.56**	**47.22**	11.14	**83.30**	**36.98**	3.36	**70.60**
	CB Loss	70.40	49.94	90.86	46.16	11.78	80.54	36.35	**4.76**	67.94
100	Standard	63.17	30.68	95.66	43.08	5.22	80.94	35.27	1.20	69.34
	ROS	63.15	30.16	96.14	43.05	5.34	80.76	35.51	1.34	69.68
	Rotation	61.98	28.14	95.82	42.16	2.78	81.54	35.16	0.10	70.22
	ColorJitter	65.01	34.44	95.58	43.09	**5.94**	80.24	35.58	**1.40**	69.76
	Mixup	64.26	32.10	96.42	41.55	1.48	81.62	34.78	0.00	69.56
	CutMix	**65.61**	**34.46**	**96.76**	**43.65**	4.08	**83.22**	35.36	0.00	70.72
	CB Loss	58.97	32.82	85.12	42.70	4.24	81.16	**35.93**	1.00	**70.86**

B.4 Detailed LLM Unlearning Results

Figure 15 shows the method-resolved Tail Probability Gap and Tail ROUGE-L Gap across long-tailed learning strategies. Each heatmap cell reports the three-seed mean for the corresponding strategy–method combination over all Tail QA. Tables 14 and 15 provide the complete method-resolved forgetting, head-retention, utility, and membership-inference results. The standard deviations in the tables quantify variation across the same three transferred seeds used in the main LLM results. Table 16 compares ROS and Paraphrase under matched tail exposure, and Table 17 separates the Tail Probability Gap into Seen and Held-out Tail QA. Table 18 reports the extended experiments at IF $= 10$. Despite matched tail exposure, ROS and Paraphrase also yield different Tail Probability Gaps at IF $= 10$ on both benchmarks.



(a) CIFAR-10 (b) CIFAR-100 (c) Tiny-ImageNet

Figure 10: Absolute pre-unlearning Tail TA under step imbalance. Each panel reports the seven strategies at IF $\in \{10, 50, 100\}$. Tail TA is the unweighted mean class-wise accuracy over classes 5–9 for CIFAR-10, 50–99 for CIFAR-100, and 100–199 for Tiny-ImageNet.

Table 6: Pre-unlearning SFT performance on TOFU-LT. We report Probability (P) / ROUGE-L (R). Entries report mean $\pm$ sample standard deviation across seeds $\{42, 123, 2024\}$.

Strategy	Seen Tail P/R	Held-out Tail P/R	Head P/R	Model Utility	MIA Separation
Matched Retraining	0.180 ± 0.001 / 0.377 ± 0.008	0.212 ± 0.000 / 0.377 ± 0.002	0.524 ± 0.002 / 0.500 ± 0.005	0.610 ± 0.008	0.224 ± 0.001
Standard	0.452 ± 0.008 / 0.438 ± 0.004	0.229 ± 0.001 / 0.383 ± 0.002	0.521 ± 0.003 / 0.494 ± 0.007	0.606 ± 0.003	0.343 ± 0.017
ROS	0.958 ± 0.001 / 0.957 ± 0.009	0.216 ± 0.002 / 0.386 ± 0.004	0.532 ± 0.003 / 0.509 ± 0.004	0.608 ± 0.003	0.500 ± 0.000
Loss Weighting	0.825 ± 0.011 / 0.792 ± 0.019	0.234 ± 0.002 / 0.386 ± 0.002	0.503 ± 0.003 / 0.488 ± 0.002	0.602 ± 0.006	0.499 ± 0.001
Paraphrase	0.679 ± 0.003 / 0.603 ± 0.013	0.234 ± 0.001 / 0.396 ± 0.009	0.521 ± 0.003 / 0.499 ± 0.002	0.613 ± 0.003	0.486 ± 0.000

C Backbone Robustness

C.1 ViT-S/4

We conduct a robustness evaluation with ViT-S/4. We evaluate CIFAR-10 under random-sample forgetting at IF $\in \{10, 100\}$ and CIFAR-100 under class-wise forgetting with tail classes $\{14, 28, 83\}$. We use five learning strategies—Standard, ROS, Rotation, CutMix, and CB Loss—and the same six unlearning methods as in the primary ResNet-18 experiments. All ViT-S/4 experiments use seed 42. Because directly transferring the ResNet-18 hyperparameters to ViT-S/4 caused severe under-forgetting or utility degradation for some methods, we tune ViT-specific unlearning hyperparameters.

All models are trained from scratch for 300 epochs with AdamW, a learning rate of 10^{-3}, weight decay of 0.1, and batch size 128. We use a 10-epoch linear warmup followed by cosine decay, label smoothing of 0.1, and model dropout of 0.1. Prior work has shown that vanilla ViTs trained from scratch can degrade substantially under long-tailed supervision (Xu et al., 2023). Our setting is particularly data-scarce: in the most severe CIFAR-10 condition and in the CIFAR-100 single-tail setting, each tail class contains only 50 training examples. The resulting low pre-unlearning Tail TA limits one-to-one comparison of all ResNet-18 patterns, so we interpret these experiments as targeted architecture-level stress tests.

On CIFAR-10, ViT-S/4 under random-sample forgetting reproduces the main severe-imbalance pattern observed with ResNet-18 (Table 20). As IF increases from 10 to 100, the cross-strategy range in pre-unlearning Tail TA decreases from 20.50 to 7.90 pp, while the range in $\Delta^{T}_{\text{Forget}}$ increases from 29.40 to 51.33 pp. On CIFAR-100, aggregate results show strong strategy–method dependence (Table 21), while the per-class heatmap (Figure 16) shows that SalUn also remains sensitive to the deleted tail class. The exact failing classes differ from ResNet-18, but the broader class-dependent behavior persists. NegGrad under ROS yields $\Delta_{\text{Tail}} = 100$ pp for all three tail classes, indicating a strategy-specific forgetting failure rather than class dependence.

Table 7: Pre-unlearning SFT performance on ELUDe-LT. We report Probability (P) / ROUGE-L (R). Entries report mean ± sample standard deviation across seeds $\{42, 123, 2024\}$.

Strategy	Seen Tail P/R	Held-out Tail P/R	Head P/R	Model Utility	MIA Separation
Matched Retraining	0.620 ± 0.001 / 0.814 ± 0.014	0.586 ± 0.002 / 0.782 ± 0.003	0.485 ± 0.003 / 0.683 ± 0.002	0.561 ± 0.003	0.033 ± 0.002
Standard	0.962 ± 0.005 / 0.935 ± 0.001	0.580 ± 0.002 / 0.782 ± 0.005	0.478 ± 0.003 / 0.679 ± 0.007	0.558 ± 0.001	0.436 ± 0.008
ROS	0.998 ± 0.002 / 0.993 ± 0.012	0.552 ± 0.002 / 0.784 ± 0.005	0.473 ± 0.002 / 0.680 ± 0.002	0.554 ± 0.001	0.478 ± 0.004
Loss Weighting	0.986 ± 0.008 / 0.975 ± 0.004	0.575 ± 0.011 / 0.784 ± 0.009	0.481 ± 0.005 / 0.677 ± 0.007	0.558 ± 0.002	0.469 ± 0.011
Paraphrase	0.998 ± 0.000 / 1.000 ± 0.000	0.554 ± 0.001 / 0.771 ± 0.010	0.472 ± 0.002 / 0.681 ± 0.004	0.555 ± 0.001	0.479 ± 0.004

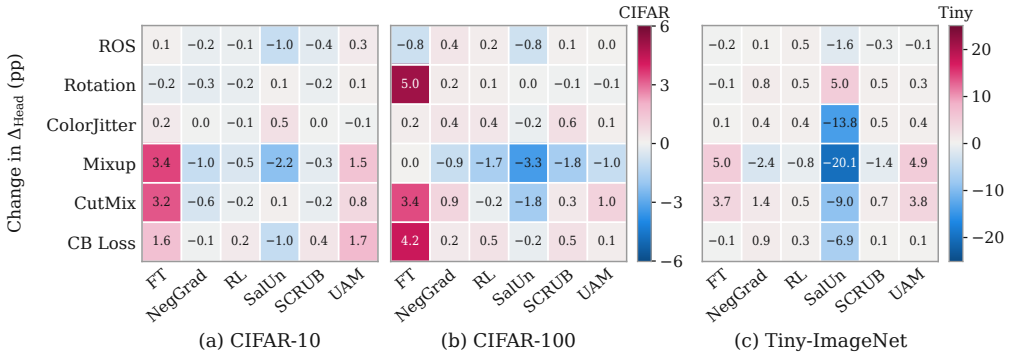


Figure 11: Strategy–method interaction in class-wise head differences. Each cell reports the paired change in Δ_{Head} relative to Standard for the same unlearning method and tail class, averaged over the evaluated tail classes. Positive and negative values indicate larger and smaller Δ_{Head} than Standard, respectively. The two CIFAR benchmarks share a symmetric scale of $[-6, 6]$ pp, while Tiny-ImageNet uses $[-25, 25]$ pp.

C.2 Qwen2.5-7B Base

We repeat the full IF $= 5$ LLM experimental grid with Qwen2.5-7B Base, using the same four learning strategies, five unlearning methods, and seeds $\{42, 123, 2024\}$. Qwen-specific unlearning hyperparameters are selected separately on Standard and then fixed across learning strategies. The unlearning strategy-level patterns emphasized in the main text persist under this additional backbone (Table 22). Loss Weighting shows a large Seen Tail Probability Gap but a comparatively small Held-out Tail Probability Gap, whereas ROS and Paraphrase show larger Held-out gaps on both benchmarks. The exposure-matched comparison also reproduces the benchmark-dependent reversal observed with Llama 3.1 8B: Paraphrase has a smaller Tail Probability Gap, higher Model Utility, and lower MIA Separation than ROS on TOFU-LT, while ROS has the corresponding advantage on ELUDe-LT. Method-resolved results further show larger method spreads under every non-Standard strategy than under Standard on both benchmarks, together with changes in the relative ordering of methods (Table 23).

D Unlearning Hyperparameters

Unless otherwise noted, method-specific unlearning hyperparameters are selected on a prespecified Standard condition and then fixed across long-tailed learning strategies within the corresponding experimental setting. The configurations below are the final settings used in the reported experiments: Tables 24, 25, 27, 28, and 29 for ResNet-18; Tables 31 and 32 for ViT-S/4; and Tables 33 and 34 for the LLMs. For SCRUB, m denotes the maximization-step setting. For UAM, the reported cosine schedule applies to ρ, not to the learning rate.

Hyperparameter selection considers both forgetting performance relative to the corresponding retraining reference and retained utility.

Vision unlearning configurations The primary ResNet-18 unlearning experiments use SGD with batch size 128, momentum 0.9, weight decay 5×10^{-4}, and a constant learning rate unless otherwise specified by the method-specific entries below.

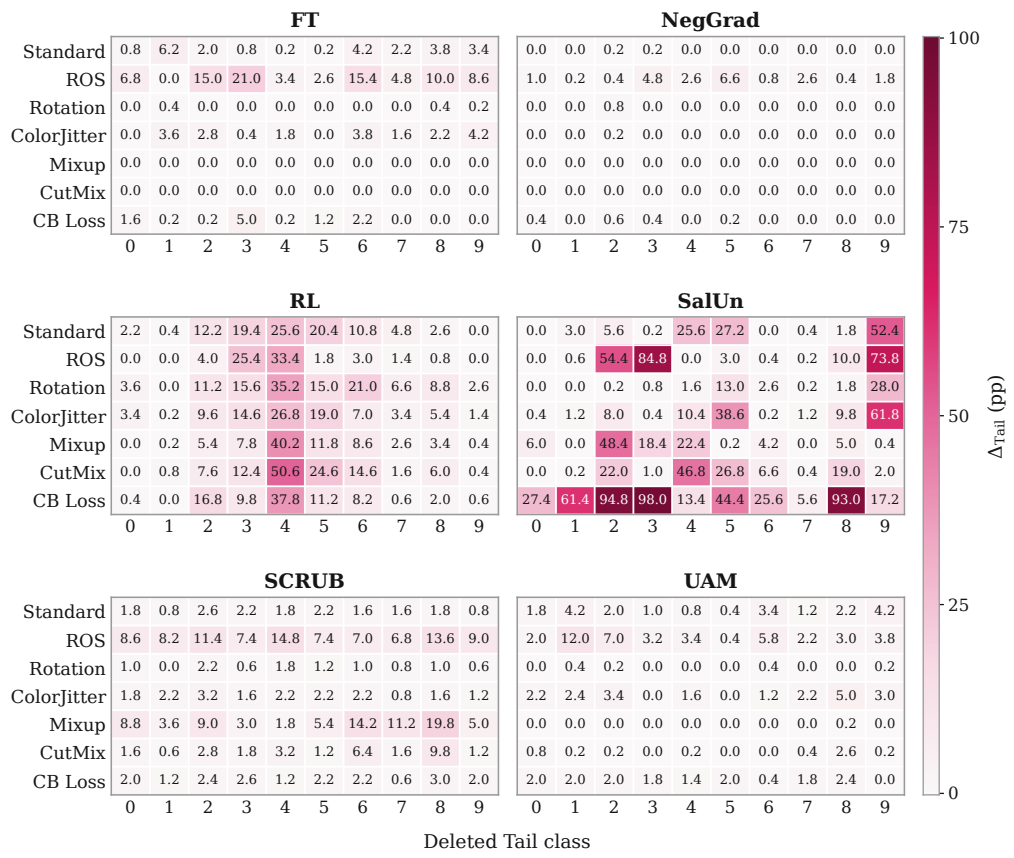


Figure 12: Method-specific Δ_{Tail} under CIFAR-10 class-wise forgetting. Each panel corresponds to one unlearning method, with rows denoting long-tailed learning strategies and columns denoting the deleted tail class. Because strategy-matched Exact Retraining attains zero forget accuracy in this setting, Δ_{Tail} equals the residual forget accuracy on the deleted tail class.

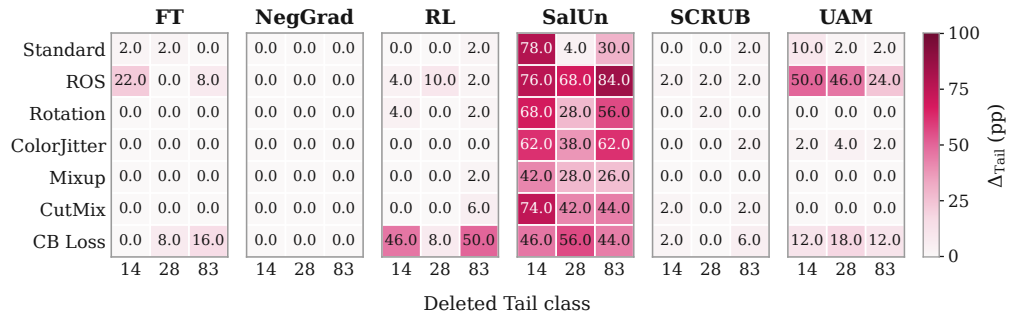


Figure 13: Method-specific Δ_{Tail} under CIFAR-100 class-wise tail forgetting. Each panel corresponds to one unlearning method, with rows denoting long-tailed learning strategies and columns denoting the forgotten tail class. Because strategy-matched Exact Retraining attains zero forget accuracy in these conditions, Δ_{Tail} equals the residual forget accuracy on the deleted tail class. Lower values indicate closer agreement with Exact Retraining.

Because the ViT-S/4 robustness study uses a reduced set of learning strategies, Table 30 records the strategy-specific training settings used in the CIFAR-10 robustness grid. These are learning-time settings rather than unlearning hyperparameters. Details are provided later.

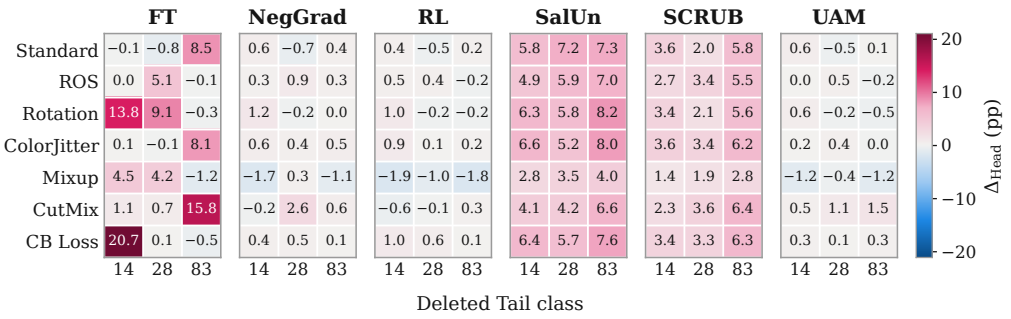


Figure 14: Method-specific Δ_{Head} across the CIFAR-100 class-wise forgetting conditions. Rows denote long-tailed learning strategies and columns denote the forgotten tail class. Positive values indicate reduced retained utility, whereas negative values indicate higher retained macro test accuracy than strategy-matched Exact Retraining.

Table 9: Random-sample unlearning results on CIFAR-10 across all strategies. Each entry reports mean ± sample standard deviation across forget-set seeds $\{42, 123, 2024\}$ after averaging the six unlearning methods within each forget request. Lower $\Delta^{T}_{\text{Forget}}$ and $\Delta^{H}_{\text{Forget}}$ indicate closer agreement with strategy- and request-matched Exact Retraining. $\Delta^{T}_{\text{Utility}}$ and $\Delta^{H}_{\text{Utility}}$ are signed; values closer to zero indicate closer retained utility to Exact Retraining.

Strategy	IF=10				IF=50				IF=100			
	$\Delta^{T}_{\text{Forget}}$	$\Delta^{H}_{\text{Forget}}$	$\Delta^{T}_{\text{Utility}}$	$\Delta^{H}_{\text{Utility}}$	$\Delta^{T}_{\text{Forget}}$	$\Delta^{H}_{\text{Forget}}$	$\Delta^{T}_{\text{Utility}}$	$\Delta^{H}_{\text{Utility}}$	$\Delta^{T}_{\text{Forget}}$	$\Delta^{H}_{\text{Forget}}$	$\Delta^{T}_{\text{Utility}}$	$\Delta^{H}_{\text{Utility}}$
Standard	16.5 ± 2.2	2.3 ± 0.2	2.2 ± 1.8	1.2 ± 0.1	45.3 ± 5.3	2.6 ± 0.3	2.5 ± 2.7	0.9 ± 0.2	55.1 ± 4.5	2.7 ± 0.2	0.9 ± 1.5	0.8 ± 0.2
ROS	13.9 ± 1.2	4.4 ± 0.2	5.2 ± 0.6	1.7 ± 0.4	33.2 ± 9.2	4.8 ± 0.0	5.4 ± 4.8	0.9 ± 0.3	43.1 ± 13.5	4.9 ± 1.4	5.7 ± 0.9	1.1 ± 0.4
Rotation	7.3 ± 0.7	2.6 ± 0.1	2.8 ± 0.5	1.1 ± 0.5	11.7 ± 3.2	2.6 ± 0.6	7.2 ± 1.5	1.2 ± 0.4	19.8 ± 9.4	2.4 ± 0.3	3.7 ± 2.6	0.6 ± 0.2
ColorJitter	11.5 ± 1.5	2.5 ± 0.4	6.0 ± 0.3	0.9 ± 0.2	28.9 ± 9.1	2.3 ± 0.2	9.0 ± 0.5	0.8 ± 0.1	48.4 ± 8.7	2.8 ± 0.2	6.3 ± 0.7	0.6 ± 0.1
Mixup	15.2 ± 0.4	2.2 ± 0.2	-0.4 ± 0.7	1.3 ± 0.2	20.9 ± 7.5	1.9 ± 0.3	1.7 ± 1.5	1.3 ± 0.5	35.8 ± 3.2	2.0 ± 0.3	0.3 ± 0.6	1.0 ± 0.2
CutMix	12.0 ± 0.5	2.0 ± 0.4	3.6 ± 0.1	1.2 ± 0.5	23.4 ± 8.0	1.9 ± 0.6	8.0 ± 1.1	1.1 ± 0.1	22.2 ± 6.2	1.6 ± 0.5	9.3 ± 3.7	1.0 ± 0.7
CB Loss	8.2 ± 1.1	7.6 ± 0.8	7.9 ± 1.3	2.5 ± 0.1	12.7 ± 6.9	8.2 ± 0.4	16.1 ± 1.4	2.6 ± 0.4	20.4 ± 3.7	4.7 ± 0.5	17.3 ± 1.6	-0.2 ± 0.1

SalUn is more sensitive to the shared hyperparameter configuration than the other methods in this setting. We therefore evaluate a targeted set of alternative SalUn configurations on tail class 28 and transfer the competitive settings to the other evaluated tail classes.

For CIFAR-100 random-sample forgetting, method-specific hyperparameters are selected using Standard and forget-set seed 42 while jointly considering IF $\in \{10, 50, 100\}$, and are then fixed across all seven learning strategies, all three IFs, and forget-set seeds $\{42, 123, 2024\}$.

All CIFAR-100 random-sample unlearning runs use batch size 128, momentum 0.9, weight decay 5×10^{-4}, and unlearning seed 42. The nominal forget-set fraction is 10% per class; at IF $= 100$, each tail class contains five training examples, so deleting one example corresponds to 20% of that class. CB Loss uses $\beta = 0.99$ for IF 10 and 50 and $\beta = 0.9$ for IF 100; these are learning-time strategy parameters rather than unlearning hyperparameters.

For Tiny-ImageNet class-wise forgetting, hyperparameters are selected using Standard pretraining with tail class 12 and then fixed across all long-tailed learning strategies and transfer classes 142 and 185.

For Tiny-ImageNet random-sample forgetting, hyperparameters are selected using Standard with forget-set seed 42 while jointly considering IF $\in \{10, 50, 100\}$ and are transferred unchanged across learning strategies, IFs, and forget-set seeds. The Tiny-ImageNet random-sample unlearning runs use batch size 128, momentum 0.9, weight decay 5×10^{-4}, and fixed non-forget-set random seeds of 42.

For CIFAR-10, ViT-specific unlearning hyperparameters are selected using Standard with forget-set seed 42 while jointly considering IF $\in \{10, 100\}$ and then fixed across all five learning strategies and both IFs.

Table 10: Random-sample unlearning results on CIFAR-100 across all long-tailed learning strategies. Each entry reports mean ± sample standard deviation across forget-set seeds $\{42, 123, 2024\}$ after averaging the six unlearning methods within each forget request. Lower Δ^T_{Forget} and Δ^H_{Forget} indicate closer agreement with strategy- and request-matched Exact Retraining. $\Delta^T_{\text{Utility}}$ and $\Delta^H_{\text{Utility}}$ are signed; values closer to zero indicate closer retained utility to Exact Retraining.

Strategy	IF=10				IF=50				IF=100			
	Δ^T_{Forget}	Δ^H_{Forget}	$\Delta^T_{\text{Utility}}$	$\Delta^H_{\text{Utility}}$	Δ^T_{Forget}	Δ^H_{Forget}	$\Delta^T_{\text{Utility}}$	$\Delta^H_{\text{Utility}}$	Δ^T_{Forget}	Δ^H_{Forget}	$\Delta^T_{\text{Utility}}$	$\Delta^H_{\text{Utility}}$
Standard	14.9 ± 1.2	10.8 ± 0.5	9.4 ± 1.1	9.0 ± 1.2	33.1 ± 2.3	7.2 ± 2.8	3.0 ± 0.4	7.3 ± 1.3	33.6 ± 4.0	5.8 ± 0.7	1.5 ± 0.3	6.6 ± 0.3
ROS	18.7 ± 5.8	29.1 ± 0.3	11.8 ± 2.2	18.9 ± 1.2	18.8 ± 1.0	26.6 ± 3.5	3.6 ± 0.4	19.0 ± 1.4	22.2 ± 2.1	25.0 ± 2.4	1.8 ± 0.4	18.0 ± 1.6
Rotation	10.8 ± 3.4	10.9 ± 1.3	6.8 ± 0.7	9.0 ± 0.9	20.8 ± 4.4	7.1 ± 0.8	1.3 ± 0.7	6.5 ± 1.7	22.4 ± 2.5	7.0 ± 1.3	0.2 ± 0.2	4.6 ± 0.8
ColorJitter	19.5 ± 3.6	10.7 ± 1.3	6.7 ± 1.4	4.6 ± 0.6	31.8 ± 8.4	8.6 ± 1.1	4.3 ± 0.5	6.8 ± 1.5	37.6 ± 3.4	8.8 ± 2.1	2.4 ± 0.3	6.0 ± 1.4
Mixup	25.1 ± 2.7	18.2 ± 0.4	6.9 ± 1.5	11.1 ± 0.5	17.6 ± 2.3	12.2 ± 1.0	2.2 ± 1.2	8.1 ± 2.2	5.1 ± 0.4	14.3 ± 1.7	-0.3 ± 0.2	9.0 ± 1.9
CutMix	21.8 ± 1.7	7.6 ± 1.4	7.2 ± 2.5	6.0 ± 1.6	25.4 ± 6.0	7.2 ± 0.4	3.9 ± 0.5	5.7 ± 1.3	15.1 ± 1.4	8.0 ± 1.0	1.0 ± 0.3	6.1 ± 0.6
CB Loss	14.6 ± 2.0	26.8 ± 2.4	15.8 ± 0.3	19.2 ± 0.9	9.2 ± 1.4	39.1 ± 1.8	7.1 ± 0.1	23.5 ± 1.4	10.9 ± 5.6	23.6 ± 4.7	2.3 ± 0.4	18.0 ± 2.8

Table 11: Random-sample unlearning results on Tiny-ImageNet across all long-tailed learning strategies. Each entry reports mean ± sample standard deviation across forget-set seeds $\{42, 123, 2024\}$ after averaging the six unlearning methods within each forget request. Lower Δ^T_{Forget} and Δ^H_{Forget} indicate closer agreement with strategy- and request-matched Exact Retraining. $\Delta^T_{\text{Utility}}$ and $\Delta^H_{\text{Utility}}$ are signed; values closer to zero indicate closer retained utility to Exact Retraining.

Strategy	IF=10				IF=50				IF=100			
	Δ^T_{Forget}	Δ^H_{Forget}	$\Delta^T_{\text{Utility}}$	$\Delta^H_{\text{Utility}}$	Δ^T_{Forget}	Δ^H_{Forget}	$\Delta^T_{\text{Utility}}$	$\Delta^H_{\text{Utility}}$	Δ^T_{Forget}	Δ^H_{Forget}	$\Delta^T_{\text{Utility}}$	$\Delta^H_{\text{Utility}}$
Standard	19.4 ± 2.4	17.7 ± 4.5	4.8 ± 1.4	10.9 ± 4.6	37.2 ± 1.0	17.6 ± 0.8	0.3 ± 0.2	4.1 ± 0.5	26.0 ± 2.0	16.5 ± 1.4	0.5 ± 0.1	5.7 ± 0.8
ROS	19.9 ± 2.0	25.9 ± 1.1	8.3 ± 0.5	16.5 ± 1.1	49.2 ± 2.0	14.8 ± 1.3	-0.1 ± 0.1	9.3 ± 1.3	52.9 ± 0.5	14.6 ± 0.6	-0.1 ± 0.2	10.5 ± 0.8
Rotation	24.9 ± 3.5	17.5 ± 5.1	-1.3 ± 0.7	10.1 ± 4.9	12.6 ± 1.9	15.8 ± 0.7	-0.6 ± 0.3	6.1 ± 2.7	9.0 ± 5.9	22.3 ± 0.7	-0.2 ± 0.1	16.2 ± 0.6
ColorJitter	22.5 ± 1.8	21.4 ± 4.1	4.7 ± 0.5	7.5 ± 5.8	31.8 ± 1.9	19.4 ± 0.2	0.6 ± 0.2	3.5 ± 0.6	19.8 ± 1.1	16.2 ± 1.0	0.8 ± 0.2	4.7 ± 0.7
Mixup	35.3 ± 0.2	18.3 ± 1.0	-0.1 ± 0.5	0.5 ± 0.2	6.7 ± 1.0	19.4 ± 0.5	-0.5 ± 0.4	0.5 ± 0.1	0.0 ± 0.0	20.4 ± 1.5	0.0 ± 0.0	-0.4 ± 1.2
CutMix	35.8 ± 1.1	18.8 ± 0.1	3.0 ± 0.3	2.3 ± 0.3	16.6 ± 1.8	19.3 ± 0.6	0.1 ± 0.4	1.9 ± 0.3	0.0 ± 0.0	18.3 ± 0.7	0.0 ± 0.0	4.0 ± 0.9
CB Loss	13.6 ± 5.6	18.9 ± 1.9	5.9 ± 1.8	10.7 ± 3.1	18.9 ± 2.6	34.3 ± 0.8	2.5 ± 0.1	19.4 ± 0.2	26.0 ± 0.3	21.4 ± 1.7	0.4 ± 0.2	8.2 ± 0.6

For CIFAR-100, the class-wise configurations are selected using Standard with tail class 28 and then fixed across all five learning strategies and tail classes $\{14, 28, 83\}$. All runs use batch size 128 and seed 42 with constant learning rates. SGD uses momentum 0.9; SCRUB uses AdamW.

LLM unlearning configurations Tables 33 and 34 report the final settings for Llama 3.1 8B and Qwen2.5-7B Base, respectively. Within each backbone and benchmark, hyperparameters are selected on Standard with seed 42 and then fixed across Standard, ROS, Loss Weighting, and Paraphrase and seeds $\{42, 123, 2024\}$. For ELUDe-LT, selection uses the development tail assignment (Justin Bieber), and the settings are transferred without retuning to the final tail assignment (Michael Jordan).

All LLM unlearning runs use full-parameter updates with AdamW, weight decay 0.01, a forget batch size of 4, and a retain batch size of 4 when a retain term is used. Learning rates are constant; gradient-norm clipping is set to 1.0, with gradient checkpointing and bfloat16 model weights. The maximum sequence lengths are 200 for TOFU-LT and 128 for ELUDe-LT. Llama uses SDPA attention, while the final Qwen unlearning runs use eager attention. The retain term in GradDiff, NPO, SimNPO, and DPO is answer-token negative log-likelihood. NPO and DPO use cached reference log-probabilities from the corresponding pre-unlearning SFT model. In the tables, α weights the retain term, γ weights the forget objective, and β is the method-specific preference-loss parameter. SimNPO uses $\delta = 0$ in both benchmarks and backbones. A dash denotes an inapplicable parameter; GradAscent has no retain term.

The targeted Llama IF $= 10$ extension reuses exactly the GradDiff, NPO, and DPO configurations in Table 33 for the corresponding benchmark, including the epoch counts, learning rates, objective coefficients, and shared optimizer settings. No IF $= 10$ retuning is performed. The extension uses seed 42 and all four learning strategies on both benchmarks; GradAscent and SimNPO are not included.

Table 12: Tail forgetting differences under random-sample forgetting. Each entry reports mean $\pm$ sample standard deviation of Δ^T_{Forget} across three forget requests, after averaging the six unlearning methods within each request. Lower values indicate closer agreement with the strategy- and request-matched Exact Retraining reference.

Dataset	IF	Standard	ROS	Rotation	ColorJitter	Mixup	CutMix	CB Loss
CIFAR-10	10	16.5 ± 2.2	13.9 ± 1.2	7.3 ± 0.7	11.5 ± 1.5	15.2 ± 0.4	12.0 ± 0.5	8.2 ± 1.1
	50	45.3 ± 5.3	33.2 ± 9.2	11.7 ± 3.2	28.9 ± 9.1	20.9 ± 7.5	23.4 ± 8.0	12.7 ± 6.9
	100	55.1 ± 4.5	43.1 ± 13.5	19.8 ± 9.4	48.4 ± 8.7	35.8 ± 3.2	22.2 ± 6.2	20.4 ± 3.7
CIFAR-100	10	14.9 ± 1.2	18.7 ± 5.8	10.8 ± 3.4	19.5 ± 3.6	25.1 ± 2.7	21.8 ± 1.7	14.6 ± 2.0
	50	33.1 ± 2.3	18.8 ± 1.0	20.8 ± 4.4	31.8 ± 8.4	17.6 ± 2.3	25.4 ± 6.0	9.2 ± 1.4
	100	33.6 ± 4.0	22.2 ± 2.1	22.4 ± 2.5	37.6 ± 3.4	5.1 ± 0.4	15.1 ± 1.4	10.9 ± 5.6
Tiny-ImageNet	10	19.4 ± 2.4	19.9 ± 2.0	24.9 ± 3.5	22.5 ± 1.8	35.3 ± 0.2	35.8 ± 1.1	13.6 ± 5.6
	50	37.2 ± 1.0	49.2 ± 2.0	12.6 ± 1.9	31.8 ± 1.9	6.7 ± 1.0	16.6 ± 1.8	18.9 ± 2.6
	100	26.0 ± 2.0	52.9 ± 0.5	9.0 ± 5.9	19.8 ± 1.1	0.0 ± 0.0	0.0 ± 0.0	26.0 ± 0.3

Table 13: Cross-strategy variation before and after random-sample forgetting. Ranges denote the difference between the largest and smallest strategy-level mean across the seven long-tailed learning strategies. Tail TA is measured before unlearning, while Δ^T_{Forget} is measured after unlearning. All values are in percentage points.

Dataset	Quantity	IF=10	IF=50	IF=100
CIFAR-10	Tail TA range	5.18	9.52	6.32
	Δ^T_{Forget} range	9.16	33.67	35.33
CIFAR-100	Tail TA range	6.20	5.24	4.46
	Δ^T_{Forget} range	14.31	23.89	32.44
Tiny-ImageNet	Tail TA range	9.94	3.96	1.40
	Δ^T_{Forget} range	22.24	42.44	52.89



(a) TOFU-LT (b) ELUDe-LT

Figure 15: Method-resolved tail unlearning performance across long-tailed learning strategies. Rows denote learning strategies and columns denote unlearning methods. Cells report the mean Tail Probability Gap (top) and Tail ROUGE-L Gap (bottom) over all Tail QA across seeds $\{42, 123, 2024\}$. Lower values indicate closer agreement with matched retraining.

Table 14: Method-resolved unlearning results on TOFU-LT. Each tail entry reports Tail Probability Gap / Tail ROUGE-L Gap, while each Head entry reports Head Probability Δ / Head ROUGE-L Δ. Entries are mean $\pm$ sample standard deviation across seeds $\{42, 123, 2024\}$. Lower Tail gaps indicate closer agreement with matched retraining. Head differences closer to 0 indicate closer retained head performance to matched retraining. Higher Model Utility and lower MIA Separation are better.

Strategy	Method	Tail	Head	Model Utility	MIA Separation
		Probability Gap / ROUGE-L Gap	Probability Δ / ROUGE-L Δ		
Standard	GradAscent	$0.009 \pm 0.012/0.003 \pm 0.005$	$0.031 \pm 0.009/0.024 \pm 0.004$	0.595 ± 0.008	0.115 ± 0.028
	GradDiff	$0.022 \pm 0.013/0.010 \pm 0.004$	$0.048 \pm 0.018/0.032 \pm 0.004$	0.584 ± 0.011	0.043 ± 0.009
	NPO	$0.024 \pm 0.008/0.008 \pm 0.006$	$0.043 \pm 0.010/0.031 \pm 0.017$	0.586 ± 0.011	0.213 ± 0.021
	SimNPO	$0.015 \pm 0.009/0.016 \pm 0.011$	$0.085 \pm 0.004/0.055 \pm 0.009$	0.579 ± 0.008	0.308 ± 0.008
	DPO	$0.005 \pm 0.002/0.006 \pm 0.005$	$0.094 \pm 0.007/0.059 \pm 0.010$	0.584 ± 0.014	0.138 ± 0.011
ROS	GradAscent	$0.192 \pm 0.014/0.108 \pm 0.111$	$0.479 \pm 0.050/0.162 \pm 0.160$	0.182 ± 0.171	0.348 ± 0.170
	GradDiff	$0.149 \pm 0.010/0.025 \pm 0.017$	$0.178 \pm 0.054/0.021 \pm 0.010$	0.521 ± 0.013	0.430 ± 0.054
	NPO	$0.017 \pm 0.004/0.021 \pm 0.005$	$0.022 \pm 0.025/0.011 \pm 0.008$	0.579 ± 0.030	0.114 ± 0.033
	SimNPO	$0.088 \pm 0.020/0.010 \pm 0.006$	$0.013 \pm 0.004/-0.001 \pm 0.012$	0.598 ± 0.012	0.499 ± 0.001
	DPO	$0.075 \pm 0.008/0.015 \pm 0.006$	$0.060 \pm 0.002/0.034 \pm 0.006$	0.585 ± 0.005	0.425 ± 0.011
Loss Weighting	GradAscent	$0.130 \pm 0.002/0.049 \pm 0.006$	$0.091 \pm 0.007/0.051 \pm 0.004$	0.585 ± 0.002	0.463 ± 0.010
	GradDiff	$0.133 \pm 0.004/0.056 \pm 0.011$	$0.069 \pm 0.010/0.031 \pm 0.003$	0.593 ± 0.003	0.472 ± 0.007
	NPO	$0.055 \pm 0.006/0.036 \pm 0.022$	$0.195 \pm 0.012/0.069 \pm 0.036$	0.526 ± 0.026	0.328 ± 0.021
	SimNPO	$0.026 \pm 0.005/0.017 \pm 0.014$	$0.092 \pm 0.009/0.055 \pm 0.008$	0.578 ± 0.008	0.036 ± 0.030
	DPO	$0.078 \pm 0.002/0.023 \pm 0.010$	$0.110 \pm 0.008/0.064 \pm 0.008$	0.574 ± 0.010	0.370 ± 0.033
Paraphrase	GradAscent	$0.071 \pm 0.095/0.018 \pm 0.021$	$0.285 \pm 0.155/0.121 \pm 0.025$	0.451 ± 0.163	0.118 ± 0.041
	GradDiff	$0.115 \pm 0.076/0.013 \pm 0.007$	$0.163 \pm 0.092/0.045 \pm 0.015$	0.522 ± 0.035	0.193 ± 0.189
	NPO	$0.009 \pm 0.004/0.011 \pm 0.007$	$0.032 \pm 0.002/0.019 \pm 0.013$	0.545 ± 0.011	0.026 ± 0.025
	SimNPO	$0.104 \pm 0.022/0.017 \pm 0.010$	$0.015 \pm 0.008/0.005 \pm 0.010$	0.612 ± 0.001	0.478 ± 0.019
	DPO	$0.038 \pm 0.007/0.008 \pm 0.002$	$0.065 \pm 0.005/0.043 \pm 0.004$	0.580 ± 0.008	0.205 ± 0.025

Table 15: Method-resolved unlearning results on ELUDe-LT. Each tail entry reports Tail Probability Gap / Tail ROUGE-L Gap, while each Head entry reports Head Probability Δ / Head ROUGE-L Δ. Entries are mean $\pm$ sample standard deviation across seeds $\{42, 123, 2024\}$. Lower Tail gaps indicate closer agreement with matched retraining. Head differences closer to 0 indicate closer retained head performance to matched retraining. Higher Model Utility and lower MIA Separation are better.

Strategy	Method	Tail	Head	Model Utility	MIA Separation
		Probability Gap / ROUGE-L Gap	Probability Δ / ROUGE-L Δ		
Standard	GradAscent	$0.024 \pm 0.002/0.008 \pm 0.006$	$0.008 \pm 0.000/0.012 \pm 0.003$	0.555 ± 0.001	0.288 ± 0.020
	GradDiff	$0.017 \pm 0.009/0.006 \pm 0.005$	$0.014 \pm 0.004/0.008 \pm 0.008$	0.554 ± 0.001	0.294 ± 0.022
	NPO	$0.011 \pm 0.009/0.019 \pm 0.010$	$0.014 \pm 0.008/0.006 \pm 0.016$	0.553 ± 0.005	0.165 ± 0.045
	SimNPO	$0.018 \pm 0.018/0.020 \pm 0.010$	$0.027 \pm 0.011/0.004 \pm 0.010$	0.550 ± 0.004	0.135 ± 0.013
	DPO	$0.010 \pm 0.014/0.116 \pm 0.029$	$-0.006 \pm 0.011/0.019 \pm 0.005$	0.547 ± 0.005	0.081 ± 0.070
ROS	GradAscent	$0.105 \pm 0.014/0.014 \pm 0.010$	$0.079 \pm 0.007/0.021 \pm 0.005$	0.517 ± 0.004	0.082 ± 0.059
	GradDiff	$0.049 \pm 0.019/0.017 \pm 0.011$	$0.051 \pm 0.010/0.007 \pm 0.002$	0.534 ± 0.006	0.223 ± 0.061
	NPO	$0.158 \pm 0.004/0.005 \pm 0.004$	$0.104 \pm 0.003/0.022 \pm 0.009$	0.504 ± 0.002	0.053 ± 0.046
	SimNPO	$0.064 \pm 0.012/0.016 \pm 0.012$	$0.041 \pm 0.004/0.007 \pm 0.003$	0.539 ± 0.003	0.110 ± 0.048
	DPO	$0.017 \pm 0.012/0.115 \pm 0.101$	$-0.008 \pm 0.008/0.009 \pm 0.013$	0.554 ± 0.002	0.177 ± 0.089
Loss Weighting	GradAscent	$0.050 \pm 0.012/0.021 \pm 0.006$	$-0.012 \pm 0.012/0.005 \pm 0.008$	0.564 ± 0.003	0.419 ± 0.017
	GradDiff	$0.035 \pm 0.015/0.022 \pm 0.014$	$0.006 \pm 0.009/0.005 \pm 0.008$	0.557 ± 0.003	0.438 ± 0.013
	NPO	$0.042 \pm 0.014/0.015 \pm 0.010$	$-0.012 \pm 0.015/0.008 \pm 0.012$	0.562 ± 0.004	0.345 ± 0.023
	SimNPO	$0.033 \pm 0.025/0.025 \pm 0.006$	$0.005 \pm 0.017/0.000 \pm 0.002$	0.558 ± 0.005	0.419 ± 0.024
	DPO	$0.033 \pm 0.011/0.070 \pm 0.029$	$-0.007 \pm 0.017/0.012 \pm 0.005$	0.545 ± 0.008	0.280 ± 0.019
Paraphrase	GradAscent	$0.088 \pm 0.024/0.013 \pm 0.012$	$0.061 \pm 0.011/0.013 \pm 0.007$	0.524 ± 0.010	0.205 ± 0.039
	GradDiff	$0.126 \pm 0.058/0.018 \pm 0.005$	$0.088 \pm 0.032/0.010 \pm 0.005$	0.506 ± 0.024	0.208 ± 0.048
	NPO	$0.243 \pm 0.038/0.071 \pm 0.022$	$0.125 \pm 0.018/0.042 \pm 0.016$	0.473 ± 0.018	0.106 ± 0.076
	SimNPO	$0.126 \pm 0.025/0.006 \pm 0.004$	$0.032 \pm 0.023/0.002 \pm 0.003$	0.534 ± 0.009	0.076 ± 0.085
	DPO	$0.022 \pm 0.016/0.173 \pm 0.098$	$-0.019 \pm 0.018/0.007 \pm 0.008$	0.557 ± 0.007	0.170 ± 0.121

Table 16: Detailed comparison of exposure-matched ROS and Paraphrase. Both strategies use the same number of tail training exposures within each benchmark, but differ in whether the additional exposure repeats source QA or introduces paraphrased questions. For each seed, unlearning metrics are first averaged over the five unlearning methods; entries report mean $\pm$ sample standard deviation across seeds $\{42, 123, 2024\}$. Lower Tail Probability Gap and MIA Separation, and higher Model Utility, are better.

Dataset	Strategy	Tail exposures	Tail Probability Gap ↓	Model Utility ↑	MIA Separation ↓
TOFU-LT	ROS	200	0.104 ± 0.006	0.493 ± 0.040	0.363 ± 0.025
	Paraphrase	200	0.068 ± 0.028	0.542 ± 0.038	0.204 ± 0.033
ELUDe-LT	ROS	80	0.079 ± 0.006	0.530 ± 0.003	0.129 ± 0.044
	Paraphrase	80	0.121 ± 0.022	0.519 ± 0.011	0.153 ± 0.019

Table 17: Method-resolved Tail Probability Gaps on Seen Tail QA and Held-out Tail QA. Each entry reports **Seen / Held-out** gap relative to strategy-matched retraining, as mean $\pm$ sample standard deviation across seeds $\{42, 123, 2024\}$. Lower values indicate closer agreement with matched retraining.

Method	Standard	Loss Weighting	ROS	Paraphrase
		TOFU-LT		
GradAscent	0.117 ± 0.026 / 0.018 ± 0.008	0.440 ± 0.005 / 0.052 ± 0.002	0.175 ± 0.008 / 0.196 ± 0.016	0.133 ± 0.031 / 0.067 ± 0.102
GradDiff	0.045 ± 0.047 / 0.030 ± 0.019	0.468 ± 0.014 / 0.050 ± 0.003	0.148 ± 0.017 / 0.149 ± 0.009	0.116 ± 0.037 / 0.133 ± 0.062
NPO	0.028 ± 0.012 / 0.023 ± 0.007	0.198 ± 0.006 / 0.019 ± 0.006	0.091 ± 0.052 / 0.018 ± 0.012	0.053 ± 0.008 / 0.025 ± 0.005
SimNPO	0.030 ± 0.004 / 0.011 ± 0.010	0.101 ± 0.013 / 0.007 ± 0.004	0.165 ± 0.006 / 0.069 ± 0.025	0.139 ± 0.020 / 0.096 ± 0.023
DPO	0.025 ± 0.004 / 0.005 ± 0.006	0.323 ± 0.009 / 0.016 ± 0.001	0.339 ± 0.019 / 0.009 ± 0.006	0.158 ± 0.013 / 0.008 ± 0.006
		ELUDe-LT		
GradAscent	0.208 ± 0.011 / 0.006 ± 0.004	0.324 ± 0.016 / 0.010 ± 0.007	0.053 ± 0.072 / 0.115 ± 0.004	0.083 ± 0.036 / 0.116 ± 0.026
GradDiff	0.206 ± 0.026 / 0.013 ± 0.008	0.341 ± 0.010 / 0.014 ± 0.016	0.123 ± 0.061 / 0.077 ± 0.012	0.088 ± 0.054 / 0.157 ± 0.057
NPO	0.097 ± 0.039 / 0.011 ± 0.012	0.260 ± 0.019 / 0.008 ± 0.013	0.167 ± 0.042 / 0.157 ± 0.011	0.305 ± 0.041 / 0.233 ± 0.040
SimNPO	0.060 ± 0.021 / 0.030 ± 0.018	0.320 ± 0.024 / 0.020 ± 0.019	0.031 ± 0.038 / 0.077 ± 0.008	0.152 ± 0.119 / 0.121 ± 0.015
DPO	0.035 ± 0.035 / 0.015 ± 0.012	0.193 ± 0.022 / 0.008 ± 0.010	0.114 ± 0.071 / 0.008 ± 0.003	0.110 ± 0.094 / 0.008 ± 0.007

Table 18: Strategy-level summary of the targeted IF $= 10$ LLM extension. Unlearning quantities are averaged over GradDiff, NPO, and DPO. Tail Probability Gap is computed over the full tail evaluation set, while Seen and Held-out Tail Probability Gaps are reported separately. Lower gap values indicate closer agreement with matched retraining.

Benchmark	Strategy	Tail exposures	Seen Tail Probability	Tail Probability Gap	Seen Tail Probability Gap	Held-out Tail Probability Gap
TOFU-LT	Standard	20	0.450	0.012	0.043	0.015
	ROS	100	0.918	0.052	0.163	0.039
	Loss Weighting	20	0.801	0.056	0.340	0.022
	Paraphrase	100	0.663	0.037	0.067	0.036
ELUDe-LT	Standard	8	0.963	0.007	0.104	0.011
	ROS	40	0.999	0.037	0.142	0.039
	Loss Weighting	8	0.971	0.013	0.129	0.008
	Paraphrase	40	0.995	0.042	0.113	0.046

Table 19: Training configuration for the CIFAR-10 ViT-S/4 robustness experiment. The same configuration is used for the original model and its strategy-matched retraining reference.

Component	**Configuration**
Architecture	Direct-head ViT-S/4
Patch size	4×4
Transformer depth	12
Embedding dimension	384
Attention heads	6
MLP dimension	1,536
Classifier	$\mathrm{Linear}(384, 10)$
Parameters	21,342,346
Initialization	From scratch
Epochs	300
Optimizer	AdamW
Initial learning rate	1×10^{-3}
Weight decay	0.1
Batch size	128
Scheduler	10-epoch linear warmup followed by cosine decay
Minimum learning rate	1×10^{-6}
Label smoothing	0.1
Model dropout	0.1
Training/data seed	42
IFs	$\{10, 100\}$
Head/Tail partition	Head: classes 0–4; Tail: classes 5–9
Forgetting	Random 10% per class
Forget-set seed	42

Table 20: CIFAR-10 ViT-S/4 robustness results under random-sample forgetting. Tail TA and Head TA are measured on the corresponding pre-unlearning checkpoints. The unlearning Δ metrics are averaged over the six unlearning methods within each strategy and IF. Lower $\Delta^{T}_{\mathrm{Forget}}$ and $\Delta^{H}_{\mathrm{Forget}}$ indicate closer agreement with Exact Retraining, while $\Delta^{T}_{\mathrm{Utility}}$ values closer to zero indicate closer retained tail utility.

IF	**Strategy**	**Tail TA**	**Head TA**	$\boldsymbol{\Delta^{T}_{\mathrm{Forget}}}$	$\boldsymbol{\Delta^{H}_{\mathrm{Forget}}}$	$\boldsymbol{\Delta^{T}_{\mathrm{Utility}}}$
10	Standard	49.76	86.54	28.67	7.67	2.52
	ROS	47.16	85.24	39.27	9.06	4.92
	Rotation	47.78	86.66	20.33	7.17	1.25
	CutMix	56.18	89.66	15.27	9.75	5.04
	CB Loss	35.68	77.94	44.67	12.97	3.01
100	Standard	8.02	87.44	36.67	6.67	2.04
	ROS	5.90	85.30	78.00	7.50	-0.17
	Rotation	9.04	87.30	30.67	6.81	-1.82
	CutMix	13.80	90.72	26.67	7.05	2.05
	CB Loss	12.74	60.18	30.00	13.43	11.91

Table 21: Aggregate CIFAR-100 class-wise forgetting results with ViT-S/4. The pre-unlearning column reports Tail TA / Head TA. Each unlearning-method entry reports $\Delta_{\mathrm{Tail}}/\Delta_{\mathrm{Head}}$. Values are mean $\pm$ sample standard deviation across tail classes $\{14, 28, 83\}$. Lower Δ_{Tail} indicates closer forgetting behavior to strategy-matched Exact Retraining, while Δ_{Head} values closer to zero indicate closer retained head utility.

Strategy	**Pre-unlearning Tail / Head TA**	**FT**	**NegGrad**	**RL**	**SalUn**	**SCRUB**	**UAM**
Standard	14.00 ± 12.12 / 59.52 ± 0.13	0.00 ± 0.00 / 2.07 ± 0.72	0.00 ± 0.00 / 1.59 ± 0.98	0.00 ± 0.00 / 0.90 ± 0.30	12.67 ± 17.01 / 0.41 ± 0.34	0.00 ± 0.00 / 0.46 ± 0.22	0.67 ± 1.15 / 5.33 ± 0.84
ROS	16.33 ± 12.06 / 59.31 ± 0.37	11.33 ± 1.15 / 2.27 ± 0.72	100.00 ± 0.00 / 0.75 ± 0.74	0.00 ± 0.00 / 1.48 ± 0.79	28.67 ± 25.01 / 0.82 ± 0.86	0.00 ± 0.00 / 0.96 ± 0.92	20.00 ± 8.72 / 5.26 ± 1.05
Rotation	12.33 ± 8.39 / 59.39 ± 0.22	0.00 ± 0.00 / 2.14 ± 0.69	0.00 ± 0.00 / 4.29 ± 3.46	0.00 ± 0.00 / 1.37 ± 0.41	90.67 ± 12.70 / 0.76 ± 0.65	0.00 ± 0.00 / 0.65 ± 0.60	0.00 ± 0.00 / 4.43 ± 0.89
CutMix	18.00 ± 13.11 / 65.30 ± 0.19	0.67 ± 1.15 / 3.24 ± 0.84	0.00 ± 0.00 / 7.68 ± 1.63	0.00 ± 0.00 / 0.72 ± 0.91	76.67 ± 25.32 / 0.13 ± 0.74	0.00 ± 0.00 / 0.69 ± 1.11	2.00 ± 2.00 / 5.97 ± 0.54
CB Loss	12.33 ± 9.07 / 59.27 ± 0.19	5.33 ± 4.16 / 2.50 ± 0.81	0.00 ± 0.00 / 0.57 ± 0.19	0.00 ± 0.00 / 1.48 ± 0.54	0.00 ± 0.00 / 0.60 ± 0.34	0.00 ± 0.00 / 0.82 ± 0.37	18.67 ± 13.32 / 4.48 ± 0.89

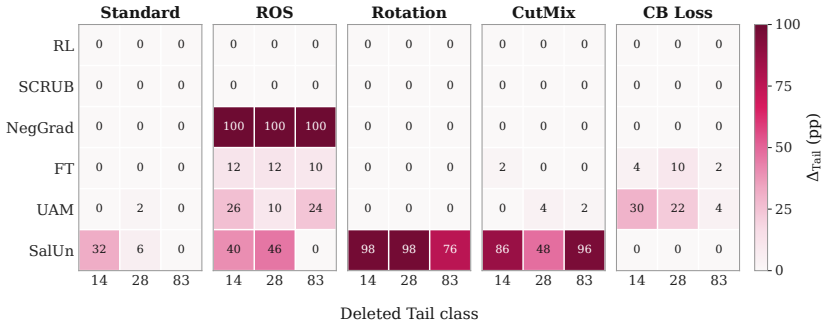


Figure 16: Method- and per-class forgetting gaps for CIFAR-100 with ViT-S/4 under class-wise forgetting. Panels denote long-tailed learning strategies, and columns denote deleted tail classes $\{14, 28, 83\}$. Each cell reports Δ_{Tail} in percentage points for the corresponding unlearning method. Lower values indicate closer agreement with strategy-matched Exact Retraining.

Table 22: Strategy-level Qwen2.5-7B Base results at IF $= 5$. Within each seed, unlearning quantities are averaged over GradAscent, GradDiff, NPO, SimNPO, and DPO; entries report mean $\pm$ sample standard deviation across seeds $\{42, 123, 2024\}$. Lower probability gaps indicate closer agreement with matched retraining, lower MIA Separation indicates weaker membership distinguishability, and higher Model Utility is better.

Benchmark	Strategy	Tail Probability Gap	Seen Tail Probability Gap	Held-out Tail Probability Gap	Model Utility	MIA Separation
TOFU-LT	Standard	0.0066 ± 0.0017	0.0309 ± 0.0018	0.0026 ± 0.0006	0.4330 ± 0.0014	0.1116 ± 0.0017
	ROS	0.0614 ± 0.0167	0.1430 ± 0.0111	0.0415 ± 0.0202	0.3888 ± 0.0148	0.2480 ± 0.0169
	Loss Weighting	0.0397 ± 0.0016	0.1541 ± 0.0045	0.0111 ± 0.0012	0.4279 ± 0.0003	0.1974 ± 0.0077
	Paraphrase	0.0311 ± 0.0127	0.0905 ± 0.0065	0.0184 ± 0.0126	0.4193 ± 0.0105	0.1438 ± 0.0123
ELUDe-LT	Standard	0.0066 ± 0.0013	0.0371 ± 0.0024	0.0043 ± 0.0018	0.4489 ± 0.0017	0.1124 ± 0.0086
	ROS	0.0342 ± 0.0015	0.1066 ± 0.0085	0.0249 ± 0.0016	0.4424 ± 0.0002	0.1528 ± 0.0294
	Loss Weighting	0.0169 ± 0.0006	0.1173 ± 0.0048	0.0064 ± 0.0033	0.4476 ± 0.0014	0.2746 ± 0.0097
	Paraphrase	0.0446 ± 0.0052	0.0986 ± 0.0069	0.0395 ± 0.0067	0.4354 ± 0.0030	0.2126 ± 0.0026

Table 23: Method-resolved Tail Probability Gap for Qwen2.5-7B Base. Entries report the mean across seeds $\{42, 123, 2024\}$. The final column reports the max–min spread across the five unlearning methods within each learning strategy. Lower Tail Probability Gap indicates closer agreement with matched retraining.

Benchmark	Strategy	GradAscent	GradDiff	NPO	SimNPO	DPO	Spread
TOFU-LT	Standard	0.0060	0.0106	0.0068	0.0011	0.0085	0.0095
	ROS	0.1225	0.0332	0.0459	0.0466	0.0588	0.0893
	Loss Weighting	0.0404	0.0444	0.0383	0.0333	0.0418	0.0110
	Paraphrase	0.0431	0.0321	0.0246	0.0198	0.0358	0.0233
ELUDe-LT	Standard	0.0119	0.0058	0.0039	0.0062	0.0051	0.0080
	ROS	0.0053	0.0230	0.0522	0.0608	0.0296	0.0555
	Loss Weighting	0.0155	0.0178	0.0091	0.0169	0.0249	0.0158
	Paraphrase	0.0120	0.0705	0.0229	0.0863	0.0315	0.0742

Table 24: Final unlearning hyperparameters for CIFAR-10 with ResNet-18. Each unlearning setting uses its own fixed method-specific configuration.

Unlearning setting	Method	Epochs	LR	Retain coeff.	Forget coeff.	Other
Class-wise	FT	25	1×10^{-2}	–	–	–
	NegGrad	1	3×10^{-4}	0.95	–	–
	RL	1	6×10^{-3}	0.5	–	–
	SalUn	3	3×10^{-3}	–	–	Threshold $= 0.5$
	SCRUB	5	5×10^{-4}	1	0.1	$m = 2$
	UAM	5	1.5×10^{-2}	–	–	$\rho = 0.05$, $\gamma = 2$, cosine ρ decay
Random-sample	FT	50	1×10^{-2}	–	–	–
	NegGrad	1	1×10^{-2}	0.85	–	–
	RL	1	6×10^{-3}	0.5	–	–
	SalUn	2	3×10^{-4}	–	–	Threshold $= 0.5$
	SCRUB	5	5×10^{-4}	1	1	$m = 2$
	UAM	5	1.75×10^{-2}	–	–	$\rho = 0.05$, $\gamma = 2$, cosine ρ decay

Table 25: Unlearning hyperparameters for CIFAR-100 class-wise forgetting. Hyperparameters are selected using Standard pretraining with tail class 28 and then fixed across all long-tailed learning strategies and tail-class assignments.

Method	Epochs	LR	Retain coeff.	Forget coeff.	Threshold	Other
FT	25	1×10^{-2}	–	–	–	–
NegGrad	1	3×10^{-3}	0.97	–	–	–
RL	1	3×10^{-3}	0.90	–	–	–
SalUn	1	9×10^{-4}	–	–	0.10	–
SCRUB	1	1×10^{-3}	1.0	0.01	–	$m_{\text{steps}} = 1$
UAM	10	1.5×10^{-2}	–	–	–	$\rho = 0.05,\ \gamma = 2.0$, cosine ρ decay

Table 26: SalUn hyperparameter sensitivity for CIFAR-100 class-wise forgetting. Each entry reports $\Delta_{\text{Tail}}/\Delta_{\text{Head}}$. The main configuration is selected on tail class 28 and transferred unchanged to tail classes 14 and 83. The competitive 2-epoch configuration is also evaluated on all three tail classes, whereas 3-epoch configurations that fail the predefined selection criteria on class 28 are not transferred. A dash denotes an unevaluated condition.

Epochs	Learning rate	Threshold	Class 14	Class 28	Class 83	Status on class 28
1	$\mathbf{9\times10^{-4}}$	**0.10**	**78.00 / 5.84**	**4.00 / 7.20**	**30.00 / 7.29**	**Selected main HP**
2	9×10^{-4}	0.10	66.00 / 5.91	2.00 / 8.64	34.00 / 7.11	Transferred sensitivity point
3	3×10^{-4}	0.10	–	42.00 / 6.67	–	FA criterion failed
3	9×10^{-4}	0.10	–	18.00 / 9.45	–	FA criterion failed
3	1.2×10^{-3}	0.10	–	12.00 / 10.04	–	Both criteria narrowly failed
3	1.5×10^{-3}	0.10	–	12.00 / 11.08	–	Both criteria failed
3	2×10^{-3}	0.10	–	12.00 / 11.72	–	Both criteria failed
3	3×10^{-3}	0.10	–	8.00 / 12.64	–	Utility criterion failed
3	3×10^{-3}	0.50	–	0.00 / 23.02	–	Utility criterion failed

Table 27: Fixed unlearning hyperparameters for CIFAR-100 random-sample forgetting. Hyperparameters were selected using only the Standard long-tailed learning strategy and forget-set seed 42, jointly considering IF $\in \{10, 50, 100\}$. Each selected setting is then fixed across all IFs, all seven long-tailed learning strategies, and forget-set seeds $\{42, 123, 2024\}$.

Method	Epochs	Learning rate	Method-specific hyperparameters
FT	10	5×10^{-2}	–
NegGrad	5	4×10^{-5}	$\lambda_{\text{retain}} = 0.1$
RL	9	1×10^{-3}	$\lambda_{\text{retain}} = 0.5$
SalUn	4	1×10^{-3}	saliency threshold $= 0.6$
SCRUB	5	5×10^{-4}	$\lambda_{\text{retain}} = 1.0$, $\lambda_{\text{forget}} = 0.875$, $m = 4$
UAM	5	5×10^{-2}	$\rho = 1.0$, $\gamma = 2.0$, cosine-ρ schedule

Table 28: Unlearning hyperparameters for Tiny-ImageNet class-wise forgetting. Hyperparameters are selected using Standard pretraining with tail class 12 and then fixed across all long-tailed learning strategies and transfer classes 142 and 185.

Method	Epochs	LR	Retain coeff.	Forget coeff.	Method-specific parameters
FT	5	1×10^{-2}	–	–	–
NegGrad	5	1×10^{-6}	0.5	–	–
RL	5	1×10^{-3}	0.5	–	–
SCRUB	5	1×10^{-4}	1.0	0.5	$m = 2$
SalUn	5	1×10^{-3}	–	–	saliency threshold $= 0.6$
UAM	5	1×10^{-2}	–	–	$\rho = 0.1,\ \gamma = 0$, cosine ρ decay

Table 29: Final unlearning hyperparameters for Tiny-ImageNet random-sample forgetting with ResNet-18. Selection uses Standard with forget-set seed 42 across IF $\in \{10, 50, 100\}$; the selected settings are transferred unchanged across strategies, IFs, and forget-set seeds.

Method	Epochs	LR	Retain coeff.	Forget coeff.	Other
FT	10	3×10^{-2}	–	–	–
NegGrad	9	7×10^{-6}	0.2	–	–
RL	9	1×10^{-3}	0.5	–	–
SalUn	3	4.5×10^{-4}	–	–	Threshold $= 0.8$
SCRUB	5	5×10^{-4}	1	0.7	$m = 4$
UAM	5	3×10^{-2}	–	–	$\rho = 1$, $\gamma = 2$, cosine ρ decay

Table 30: Long-tailed learning strategies used in the CIFAR-10 ViT-S/4 robustness experiment. All strategies retain the common training configuration in Table 19.

Strategy	Strategy-specific configuration
Standard	Random crop and horizontal flip
ROS	Class-balanced random oversampling with source repetition
Rotation	Tail-only rotation within $\pm10^\circ$; no oversampling
CutMix	$\alpha = 1.0$, application probability $p = 0.5$
CB Loss	Effective-number weighting with $\beta = 0.9999$

Table 31: Fixed unlearning hyperparameters for the CIFAR-10 ViT-S/4 experiment.

Method	Optimizer	Epochs	LR	Weight decay	Momentum	Method-specific configuration
FT	SGD	10	1×10^{-1}	5×10^{-4}	0.9	–
NegGrad	AdamW	2	5×10^{-5}	1×10^{-4}	–	$\lambda_{\text{retain}} = 0.5$
RL	AdamW	5	1×10^{-5}	1×10^{-4}	–	$\lambda_{\text{retain}} = 0.25$
SalUn	AdamW	5	5×10^{-5}	1×10^{-4}	–	Saliency threshold $= 0.6$
SCRUB	AdamW	5	3×10^{-5}	1×10^{-4}	–	$\lambda_{\text{retain}} = 1.0$, $\lambda_{\text{forget}} = 1.0$, $m = 4$
UAM	AdamW	10	1×10^{-3}	1×10^{-4}	–	$\rho = 0.05$, $\gamma = 2.0$, cosine-ρ schedule

All unlearning runs use batch size 128 and seed 42. The UAM configuration is the utility-aware knee selected from the ViT-specific tuning results rather than the more aggressive scalar-ranking winner. Exact Retraining uses the same architecture, learning strategy, and training hyperparameters as the corresponding original model, with the matched forget set removed.

Table 32: Final unlearning hyperparameters for CIFAR-100 ViT-S/4 under class-wise forgetting.

Method	**Optimizer**	**Epochs**	**LR**	**Weight decay**	**Method-specific configuration**
FT	SGD	40	1×10^{-2}	5×10^{-4}	–
NegGrad	SGD	1	3×10^{-3}	5×10^{-4}	$\lambda_{\text{retain}} = 0.97$
RL	SGD	5	1×10^{-2}	5×10^{-4}	$\lambda_{\text{retain}} = 0.5$
SalUn	SGD	1	9×10^{-4}	5×10^{-4}	Threshold $= 0.1$
SCRUB	AdamW	5	3×10^{-5}	1×10^{-4}	$\lambda_{\text{retain}} = 1$, $\lambda_{\text{forget}} = 1$, $m = 3$
UAM	SGD	10	3×10^{-2}	5×10^{-4}	$\rho = 0.1$, $\gamma = 2$, cosine ρ decay

Table 33: Final Llama 3.1 8B unlearning hyperparameters for the primary IF $= 5$ setting.

Benchmark	Method	Epochs	LR	α	β	γ
TOFU-LT	GradAscent	2	2×10^{-6}	–	–	1
	GradDiff	2	3×10^{-6}	1	–	1
	NPO	3	3×10^{-6}	1	0.2	1
	SimNPO	10	1×10^{-5}	1	1.5	0.125
	DPO	5	1×10^{-5}	1	0.2	1
ELUDe-LT	GradAscent	2	2×10^{-6}	–	–	1
	GradDiff	2	2.5×10^{-6}	1	–	1
	NPO	2	3×10^{-6}	1	0.1	1
	SimNPO	5	5×10^{-6}	1	1.5	0.125
	DPO	5	1×10^{-5}	1	0.2	1

Table 34: Final Qwen2.5-7B Base unlearning hyperparameters for the IF $= 5$ backbone robustness experiment.

Benchmark	Method	Epochs	LR	α	β	γ
TOFU-LT	GradAscent	2	2.3×10^{-6}	–	–	1
	GradDiff	2	2×10^{-6}	0.5	–	1
	NPO	2	2.5×10^{-6}	1	0.5	1
	SimNPO	7	7.5×10^{-6}	1	0.5	0.125
	DPO	3	6.5×10^{-6}	1	0.4	1
ELUDe-LT	GradAscent	2	1.7×10^{-6}	–	–	1
	GradDiff	2	4.15×10^{-6}	1	–	1
	NPO	2	3×10^{-6}	1.5	0.2	1
	SimNPO	6	7.5×10^{-6}	1	0.5	0.125
	DPO	4	1×10^{-5}	1	0.35	1

E ANNOTATION PROMPTS

We used `GPT-5.6 Sol` to generate the paraphrased Q–A sets evaluated in Figure 3. The prompts below summarize the generation instructions used in our experiments. Fields enclosed in angle brackets were filled with the corresponding source information before inference.

Prompt E.1: TOFU-LT paraphrasing prompt

```
You are given a question-answer pair about a fictitious author.
    Generate four paraphrased question-answer pairs that express the
    same underlying fact.

Requirements:

1. Preserve all factual information in the source pair, including
    names, titles, dates, numbers, events, and relationships.
2. Preserve the intent and question type of the source question.
3. Paraphrase both the question and the answer using natural and
    meaningfully different wording and sentence structure.
4. Include the author's full name explicitly in every generated answer
    .
5. Do not add unsupported information, omit essential information,
    make new inferences, or generalize beyond the source.
6. Do not copy the source wording too closely.
7. Make the four variants distinct from both the source and one
    another.
8. Avoid unnecessary elaboration, stylistic padding, and answer
    leakage in the question.
9. Keep each answer reasonably close to the source answer in length
    and level of detail.
10. Return only the four generated question-answer pairs.

Source question:
<SOURCE_QUESTION>

Source answer:
<SOURCE_ANSWER>
```

Prompt E.2: ELUDe-LT paraphrasing prompt

```
You are given a question-answer pair about a target entity. Generate
    four paraphrased versions of the question. The original gold
    answer will be retained unchanged.

Requirements:

1. Preserve the exact meaning, factual scope, and answerable content
    of the source question.
2. Every generated question must be answerable by the original gold
    answer without requiring additional information.
3. Include the exact target entity name explicitly in every question.
4. Use natural but lexically and syntactically diverse wording.
5. Do not introduce new facts, omit essential constraints, change the
    requested relation, or make unsupported inferences.
6. Do not reveal or strongly suggest the gold answer in the question.
7. Do not reproduce the source question verbatim or produce duplicate
    variants.
8. Use only the supplied source question and answer. Do not rely on
    evaluation-only questions, paraphrases, or perturbed answers.
9. Return only the four paraphrased questions.
```

Prompt E.2: ELUDe-LT paraphrasing prompt (continued)

```
Target entity:
<TARGET_ENTITY>

Source question:
<SOURCE_QUESTION>

Gold answer:
<SOURCE_ANSWER>
```