Robust Continual Unlearning against Knowledge Erosion and Forgetting Reversal

Supplementary Material

8. Theoretical Analysis of Unlearning Margin Suppression

Proposition 1 (Effect of Negative Unlearning Margin)**.** *For a non-retain sample* $\mathbf{x}$ *with original label* y*, optimizing the KL divergence loss with the random retain-class target distribution* $\mathbf{q}$ *in Eq. 9 drives its unlearning margin toward negative values:*

$$UM(\mathbf{x}) < 0. \tag{15}$$

Thus, forgotten samples remain suboptimal within the decision space.

Proof Sketch. Let $\mathbf{p}(\mathbf{x}) = \mathrm{softmax}(\boldsymbol{\ell}(\mathbf{x}))$ denote the model's predictive distribution over logits $\boldsymbol{\ell}$. Then the KL objective is written as:

$$\mathcal{L}_{\mathrm{KL}}(\mathbf{x}) = \mathrm{KL}(\mathbf{q} \,\|\, \mathbf{p}(\mathbf{x})) = \sum_i q_i \log \frac{q_i}{p_i(\mathbf{x})}, \tag{16}$$

where i indexes over all classes.
Since $q_y = 0$ and $q_i > 0$ only for $i \in \mathcal{D}^{(t)}_{\text{retain}}$, the gradient w.r.t. logits becomes:

$$\frac{\partial \mathcal{L}_{\mathrm{KL}}}{\partial \ell_i} = \begin{cases} p_i(\mathbf{x}) - q_i, & i \in \mathcal{D}^{(t)}_{\text{retain}}, \\ p_i(\mathbf{x}), & i \notin \mathcal{D}^{(t)}_{\text{retain}}. \end{cases} \tag{17}$$

In particular, for the ground-truth class y, if $p_y(\mathbf{x}) > 0$, gradient descent decreases ℓ_y. Meanwhile, logits of retain classes $p_i(\mathbf{x})$ are pushed toward matching q_i. At a stationary point, $p_y(\mathbf{x})$ approaches 0, leading to $\ell_y \ll \max_{j \in \mathcal{D}^{(t)}_{\text{retain}}} \ell_j$. Because $\max_{k \neq y} \ell_k = \max_{j \in \mathcal{D}^{(t)}_{\text{retain}}} \ell_j$, we obtain $UM(\mathbf{x}) = \ell_y - \max_{k \neq y} \ell_k < 0$.
Therefore, the forgotten class is consistently discouraged from re-entering its decision region. Since this update is repeatedly applied across unlearning phases, negative unlearning margins consistently prevent forgetting reversal. □

9. Experiment Details

9.1. Implementation

Original Models. We trained ResNet-18 on CIFAR-100, ResNet-50 on VGGFace2, and ViT-B/16 on MUFAC from scratch. For optimization, we used SGD with a learning rate of 0.1 for CIFAR-100 and VGGFace2, and 0.001 for MUFAC, along with weight decay of 0.0005 and momentum of 0.9. We set the batch size to 128 for CIFAR-100 and 64 for VGGFace2 and MUFAC.
SAFER and Baselines. Retrain models were trained in the same manner as original models, using only the remaining data $\mathcal{D}^{(t)}_{\text{retain}}$. For the other baseline methods, we utilized publicly available source code.
We perform 10 unlearning epochs for all experiments, and the optimal hyperparameters are selected using Optuna [2]. All methods are tuned only in Phase 1, and the selected hyperparameters are fixed and reused for all subsequent phases to ensure a fair comparison without phase-wise re-optimization. This optimization protocol is applied to all experimental results in this paper unless otherwise noted.
Hardware Specifications. All experiments are conducted on Ubuntu 18.04 using NVIDIA GeForce RTX 3090 GPUs, each equipped with 24,268 MB of memory.

9.2. Unlearning Targets

Tab. 4 describes the unlearning targets used in our experiments. At each phase, we unlearn 3 classes for CIFAR-100 and VGGFace2, and 20 identities for MUFAC. The three-phase unlearning process is considered as a single evaluation unit, and we repeat it three times with different targets. The reported results are averaged over the three runs.

Set	Phase	CIFAR-100	VGGFace2	MUFAC
1	1	31,33,35	31,33,35	250 - 259, 260 - 269
	2	37,39,51	37,39,51	270 - 279, 280 - 289
	3	53,55,57	53,55,57	700 - 709, 710 - 719
2	1	71,63,20	6,7,8	230 - 239, 320 - 329
	2	35,17,81	26,27,28	510 - 519, 600 - 609
	3	9,18,15	51,52,53	610 - 619, 620 - 629
3	1	6,7,8	36,37,38	240 - 249, 340 - 349
	2	26,27,28	2,3,4	540 - 549, 850 - 859
	3	51,52,53	81,82,83	860 - 869, 870 - 879

Table 4. Unlearning target classes for CIFAR-100, VGGFace2, and MUFAC across three phases in our experiments.

Dataset	Phase	IC	IC+CD	UM+IC+CD
CIFAR-100	1	0.8687	0.9857	**0.9910**
	2	0.6052	0.8635	**0.9889**
	3	0.4621	0.3944	**0.9981**
VGGFace2	1	0.6890	0.9806	**0.9824**
	2	0.4655	0.7456	**0.9844**
	3	0.6697	0.8090	**0.9887**
MUFAC	1	0.7372	0.7371	**0.9618**
	2	0.4563	0.4445	**0.8375**
	3	0.4895	0.4783	**0.8086**

Table 5. Ablation study for CIFAR-100, VGGFace2, and MUFAC. The results represent ToW scores, and higher values indicate better unlearning performance. UM corresponds to the optimization in Eq. (9), whereas IC and CD optimize the intra-class compactness and centroid drift suppression terms in Eq. (6), respectively. SAFER includes all three components (UM+IC+CD).

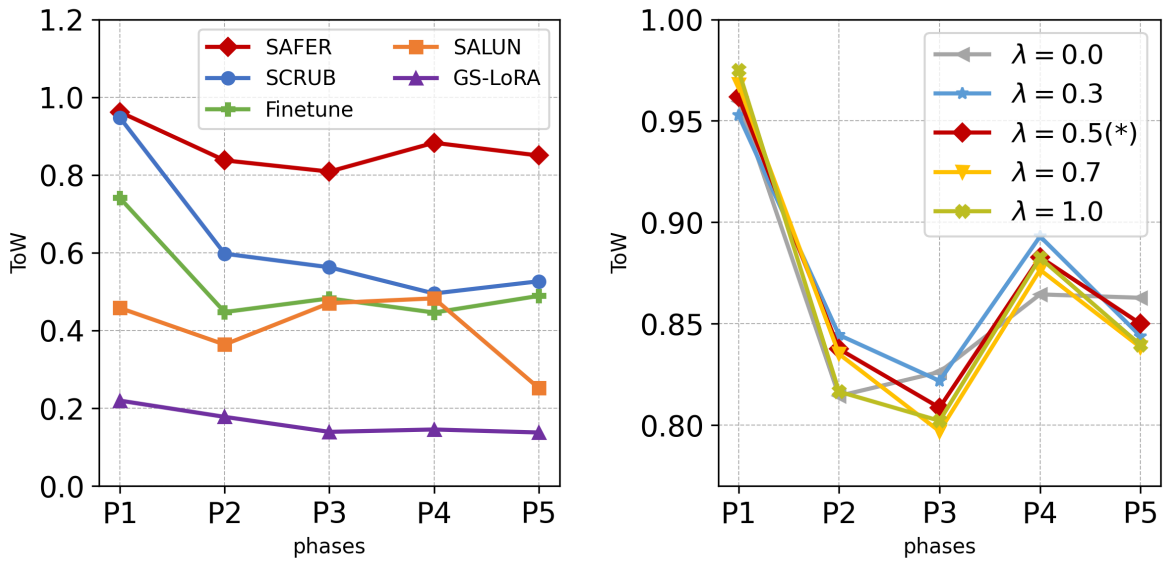
10. Additional Analysis

Ablation Study. As presented in Tab. 5, we conduct ablation studies to evaluate the impact of each component in SAFER. The ablation results are evaluated under the same conditions described in Sec. 5, and for each phase, the highest ToW score is highlighted in bold.

First, when optimizing intra-class compactness (IC), the ToW score gradually decreases after the first phase in both class-aligned and class-misaligned unlearning settings. This suggests that repeated unlearning with only IC optimization causes the model to diverge from the retrained model (the ideal unlearning reference) in both model utility and unlearning efficacy. Second, adding centroid drift suppression to intra-class compactness (IC+CD) yields consistent performance improvements in the class-aligned setting compared to IC alone. However, its effect remains minimal in the class-misaligned setting, suggesting that controlling centroid drift has limited influence under this scenario. Lastly, when all components are incorporated (UM+IC+CD), the ToW scores are significantly higher than those of the other settings across all datasets and phases. This result implies that jointly optimizing unlearning margin and retain-representation clusterability is crucial to preserving both model utility and effective forgetting throughout multiple unlearning phases.

Results in Extended Phases. A downward trend of SAFER is observed for MUFAC in Fig. 2a. Therefore, we conduct additional experiments up to five phases under the same experimental setting. As described above, the phase 5 results are obtained using the same hyperparameters as in the main paper, without any additional tuning. In this experiment, we include GS-Lora as an additional baseline, using the authors' default hyperparameters. As shown in Fig. 7a, SAFER remains competitive with strong baselines and does not exhibit a persistent downward trend beyond phase 3. It consistently maintains a ToW above 0.8, indicating that the discrepancy from the Retrain model does not accumulate over extended phases, even in the class-misaligned setting. These results indicate that SAFER maintains competitive and stable performance beyond three phases.

Hyperparameter Sensitivity. To further examine the robustness of our proposed method, we provide a sensitivity analysis of the regularization weight λ in Eq. (6), which controls cluster separation. As shown in Fig. 7b, varying λ results in highly consistent trends up to phase 5, without significant performance fluctuations. These results indicate that SAFER shows limited sensitivity and remains stable across a reasonable range of hyperparameter values.



(a) Baseline comparison

(b) Hyperparameter sensitivity

Figure 7. ToW performance across phases 1-5.

Representation Similarity. To further evaluate representation stability under continual unlearning settings, we measure the cosine similarity between the feature representations extracted by the model at each unlearning phase and those of the original model before any unlearning is applied.
For a given sample x_i, let the original model and the model after t-th unlearning phase produce feature representations:

$$\mathbf{f}_i^{\text{before}} = f_{\theta_0}(x_i), \qquad \mathbf{f}_i^{\text{after}} = f_{\theta_t}(x_i), \tag{18}$$

where $t \in \{1, 2, 3\}$. Applying $L2$-normalization, we measure the representation similarity using cosine similarity:

$$\text{Sim}(x_i) = \frac{\mathbf{f}_i^{\text{before}} \cdot \mathbf{f}_i^{\text{after}}}{\|\mathbf{f}_i^{\text{before}}\|_2 \, \|\mathbf{f}_i^{\text{after}}\|_2} \tag{19}$$

, where $\text{Sim}(x_i) \in [-1, 1]$. We obtain the similarity distributions for retain data and forget data as:

$$\text{Sim}_{\text{retain}}^{(t)} = \{\text{Sim}(x_i) \mid i \in \mathcal{D}_{\text{retain}}^{(t)}\}, \qquad \text{Sim}_{\text{forget}}^{(t)} = \{\text{Sim}(x_i) \mid i \notin \mathcal{D}_{\text{retain}}^{(t)}\}. \tag{20}$$

The underlying rationale is that an ideal unlearning algorithm should remove information only related to the forgotten samples, while preserving knowledge from the retain samples. Thus, high similarity for retain samples across unlearning phases indicates that the model successfully maintains useful knowledge without feature distortion. Conversely, decreasing similarity for forget samples suggests that their embeddings diverge from their original feature space, reflecting effective forgetting.
Fig. 8, Fig. 9, and Fig. 10 present the representation similarity results between the original model and the unlearned models at each phase for CIFAR-100, VGGFace2, and MUFAC. For the retrained model in the class-aligned unlearning setting, the retain and forget samples form distinguishable distributions in terms of their representation similarity to the original model. The retain-sample distribution is shifted toward the right with a narrower spread, indicating consistently high similarity. In contrast, the forget-sample distributions across all unlearning phases shift further left with a broader spread, indicating that their similarity to the original model remains uniformly low. In the class-misaligned unlearning setting, the retrained model exhibits retain and forget similarity distributions with nearly identical shapes, although the forget distribution tends to be slightly more dispersed. Among all approximate unlearning methods, SAFER most closely preserves the representation similarity characteristics of Retrain. This indicates that SAFER achieves desirable and consistent unlearning behavior, not only in accuracy-based evaluation but also in this representational analysis.

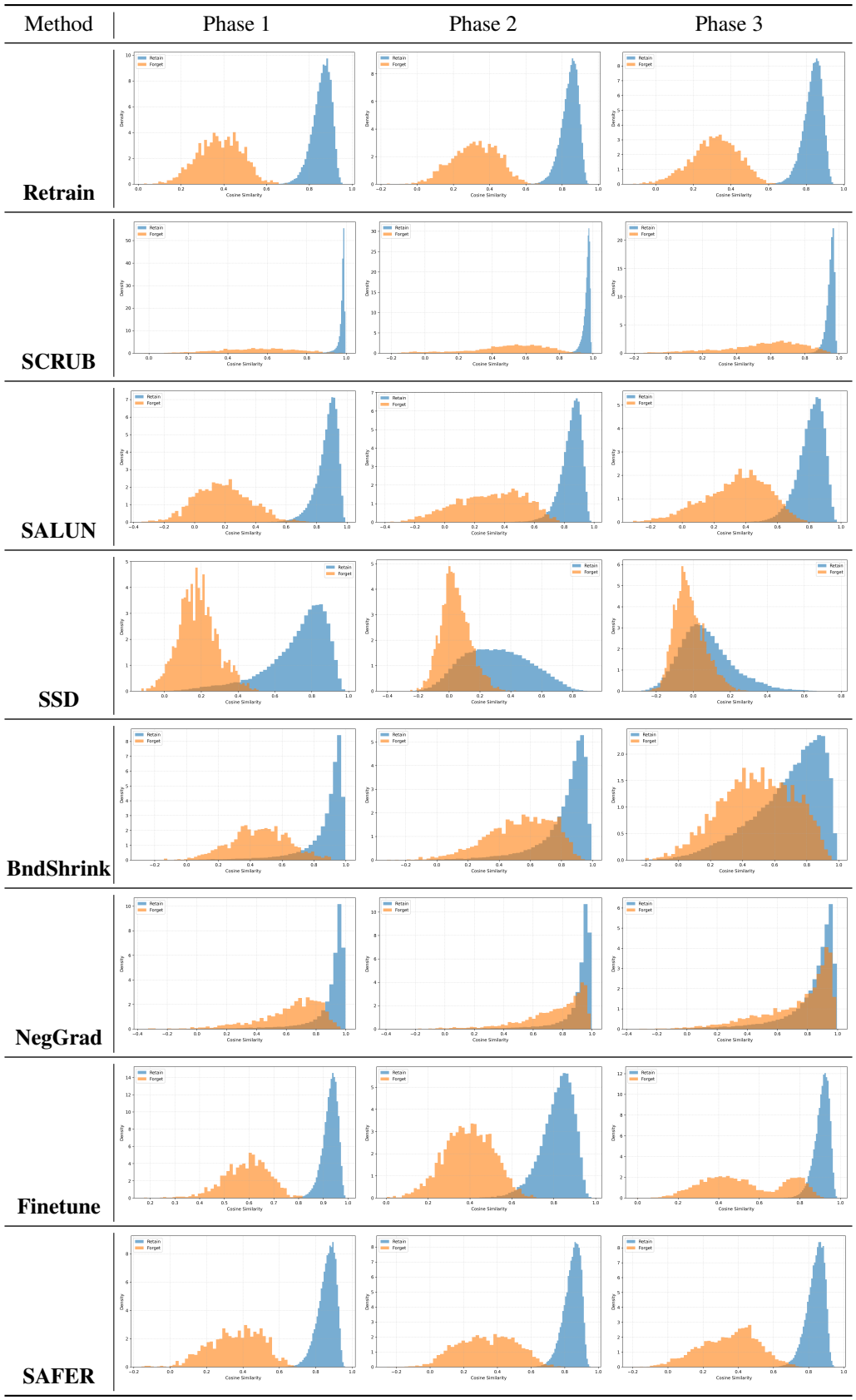


Figure 8. Representation similarity over the three-phase unlearning process on CIFAR-100. The orange shows the retain data distribution, while the blue shows the forget data distribution.

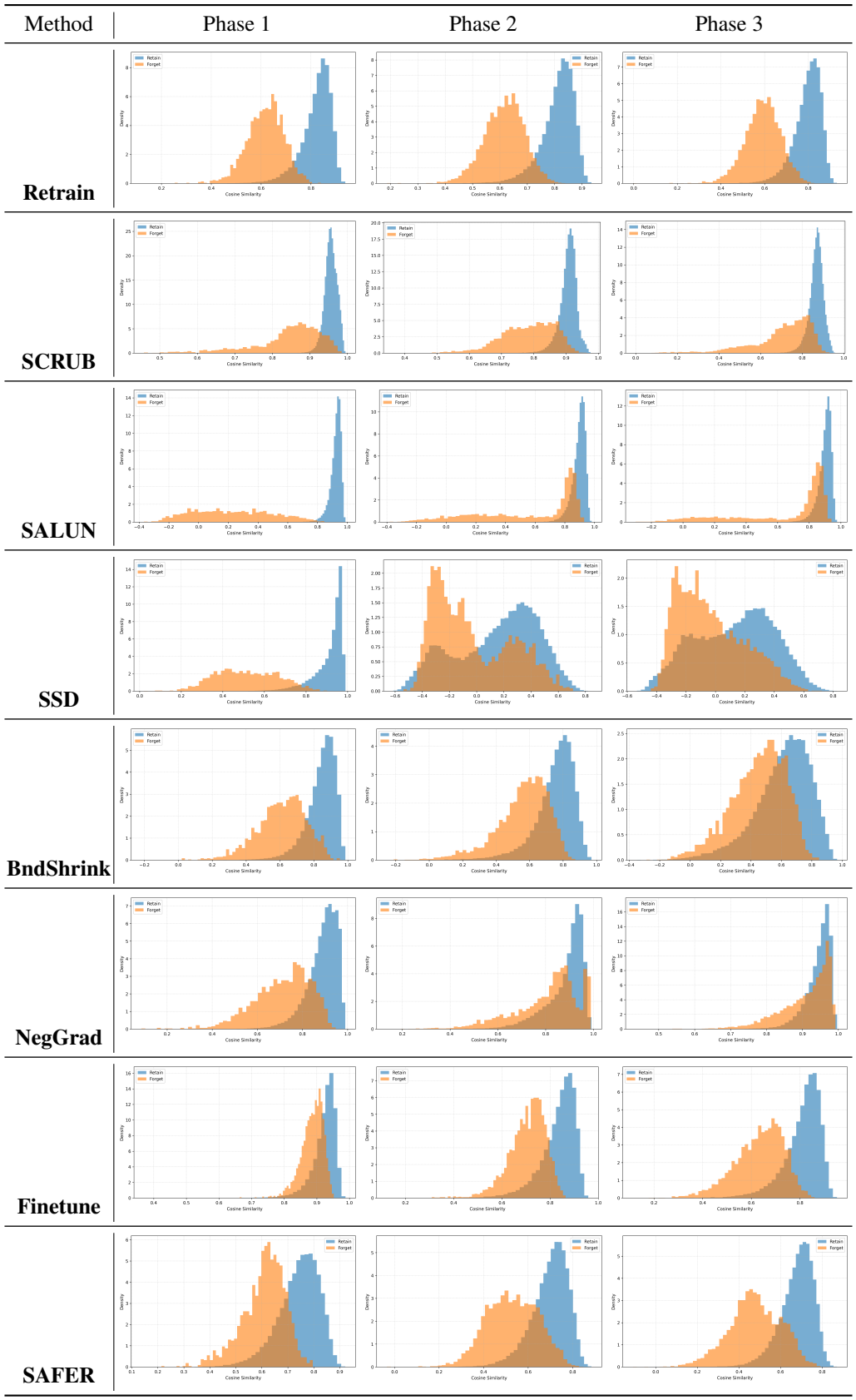


Figure 9. Representation similarity over the three-phase unlearning process on VGGFace2. The orange shows the retain data distribution, while the blue shows the forget data distribution.

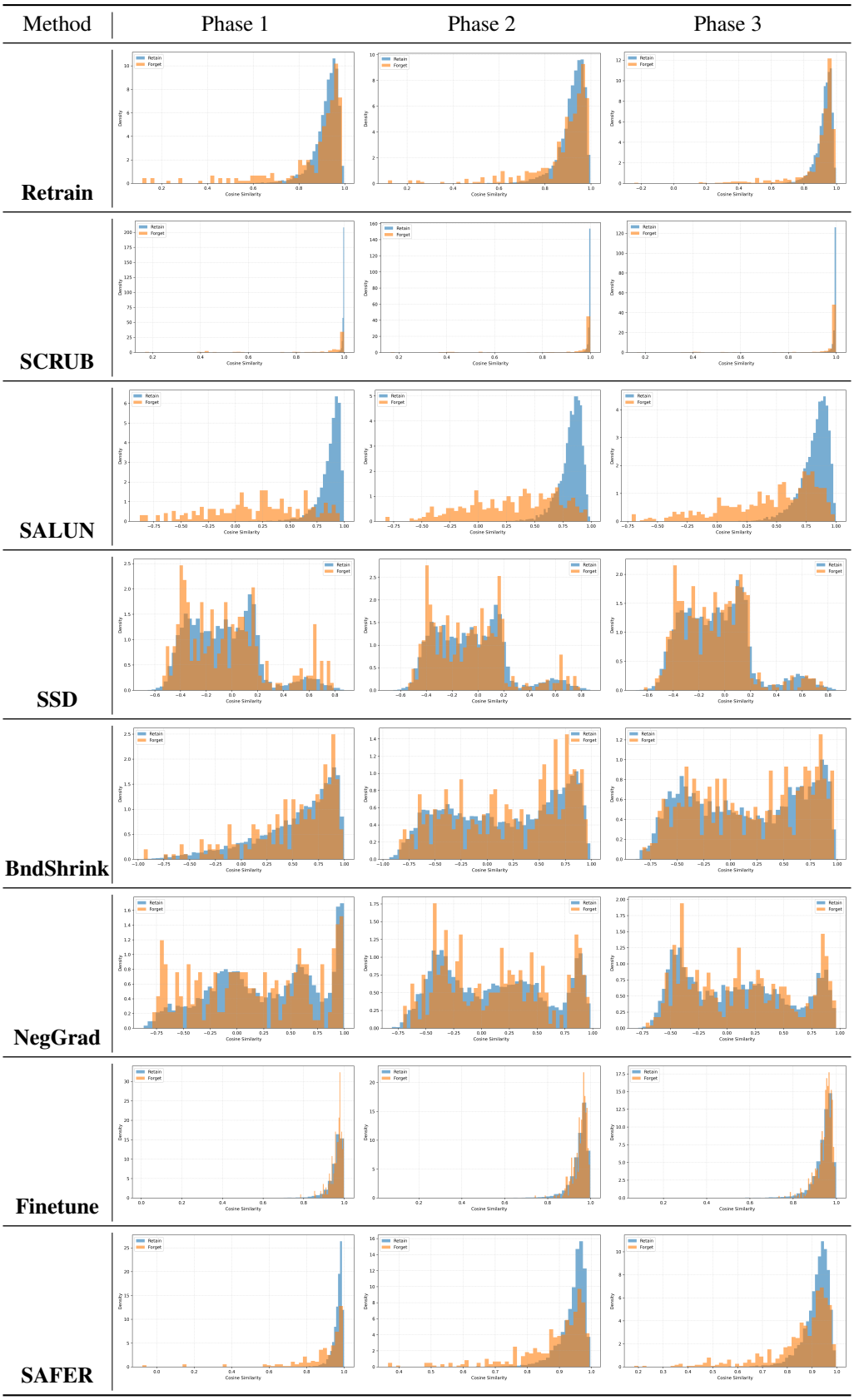


Figure 10. Representation similarity over the three-phase unlearning process on MUFAC. The orange shows the retain data distribution, while the blue shows the forget data distribution.