

Multi-Objective Reference-Aligned Machine Unlearning

Rasa Khosrowshahli^{†,*}, Stephen Asobiela[†], Beatrice Ombuki-Berman[†], Shahryar Rahnamayan[‡]

[†] Department of Computer Science, Brock University, St. Catharines, ON, Canada

[‡] Yousef Haj-Ahmad Department of Engineering, Brock University, St. Catharines, ON, Canada

Abstract

Machine unlearning aims to remove the influence of specific training samples while preserving the model’s utility. Existing single-objective approaches, such as gradient ascent or random relabeling, often induce catastrophic forgetting due to conflicting optimization dynamics and unbounded forgetting objectives that cause the model to drift from its pre-trained knowledge. We propose Reference-Aligned UnLearning (RAUL), a multi-objective framework that jointly optimizes forgetting and retention by replacing unbounded loss maximization with a bounded KL alignment of predictions on forgotten samples toward a reference distribution representing unseen data, instantiated either as a uniform distribution or an empirical distribution from a held-out reference set, which constrains the forgetting objective and reduces gradient conflict with retention. The resulting multi-objective optimization (MOO) problem is solved via Jacobian descent, which aggregates multiple gradients into a direction that does not conflict. Our results demonstrate that RAUL achieves the closest gap compared to full retraining.

Keywords: Machine Unlearning, Multi-objective Machine Unlearning, Multi-objective Optimization, Multi-task Learning.

1. Introduction

Machine unlearning (MU) removes the influence of designated training points from a deployed model, motivated by privacy regulations [1] and the prohibitive cost of retraining from scratch. Approximate methods usually optimize a single forgetting signal, which often conflicts with retention and yields catastrophic forgetting [2]. The bottleneck is explicit conflict between forgetting and retaining [3].

We address this unlearning challenge by proposing a multi-objective RAUL¹ formulation over forgetting and retention [4] that replaces loss maximization or random labeling on forgotten samples with a reference-based objective in which predictions on the forget set align with a reference distribution to simulate “unseen” data behavior. We propose two references, namely uniform and held-out data distribution. Together, they yield controllable forgetting by aligning the model’s confidence on forget data with that of unseen data [5].

2. Background and Related Work

We first formalize the machine unlearning problem, then review gradient-based methods, and finally survey recent multi-objective (MO) formulations that motivate our approach.

2.1. Problem Formulation: Machine Unlearning

Let $\mathcal{D} = \mathcal{R} \cup \mathcal{F}$ denote the training dataset, where \mathcal{F} represents the forget set containing samples requested for removal, and \mathcal{R} denotes the retain set consisting of all remaining data. Given a pretrained model f_θ with parameters θ trained on \mathcal{D} , the goal of MU is to obtain an updated model f_{θ_u} such that the influence of samples in \mathcal{F} is effectively removed from the model, and performance on \mathcal{R} is preserved or minimally degraded.

¹Code available at <https://github.com/rkhosrowshahi/MO-RAUL>.

*rkhosrowshahli@brocku.ca (corresponding)

2.2. Gradient-Based Unlearning

Gradient ascent (GA) is widely used as a baseline, and it maximizes loss on the forget set so that the model is pushed toward misclassifying those inputs [6]. A related strategy is to fine-tune while replacing supervision on \mathcal{F} with random labels [7]. Together, these approaches cast forgotten samples as adversarial targets to be repelled from correct predictions, which frequently conflicts with the retention objective and can produce catastrophic forgetting and sharp utility loss on \mathcal{R} [2, 8]. Subsequent work seeks to ease this tension by projecting forgetting gradients so they conflict less with retention gradients. Such refinements improve stability but still treat unlearning on \mathcal{F} as untraining.

Widely used benchmarks mix several instantiations of this tension. GA [9] and RL [10] foreground forgetting by increasing loss or randomizing labels on \mathcal{F} . FT instead omits \mathcal{F} from continued training on \mathcal{R} [11], which stabilizes updates but may leave residual influence from the pretrained snapshot. IU and ℓ_1 -sparse pursue approximate removal through influence-based and sparsity-constrained parameter updates [12, 13], while SalUn [14] couples saliency with forget-set adaptation under a shared training recipe. Despite their diversity, these pipelines often still tie forgetting to aggressive error shaping on forgotten data or to localized edits that interfere with retained representations.

2.3. Multi-Objective Unlearning

Recent work treats forgetting and retention as competing objectives; Wu and Harandi [15] formulate MU as Nash bargaining with Pareto guarantees, but careful utility design and computational cost limit scalability to large pretrained models.

3. Reference-Aligned UnLearning

We propose multi-objective **Reference-Aligned UnLearning** (RAUL), a framework that formulates machine unlearning as a two objective optimization problem. Traditional methods that maximize error on forgotten samples push the model to strongly reject those inputs, damaging its internal structure and causing it to forget things it should have kept [2]. Rather than adversarially repelling forgotten samples, we align their predictions with a reference distribution representing unseen data, while simultaneously minimizing retention loss on the remaining data, so that a truly forgotten sample yields predictions similar to what the model would produce for data it has never encountered. We formally define the two objectives, detail the two reference distribution strategies, and present the MOO formulation solved via Jacobian descent.

3.1. Retention Objective

For the retention objective, we minimize the standard cross-entropy loss on \mathcal{R} as:

$$\min_{\theta} \mathcal{L}_{\text{retain}}(f_{\theta}, \mathcal{R}) = -\mathbb{E}_{(x,y) \sim \mathcal{R}} [\log p_{\theta}(y | x)], \quad (3.1)$$

where $p_{\theta}(y | x)$ is the model’s predicted probability for the true label y given input x . This objective maintains predictive performance on \mathcal{R} during unlearning.

3.2. Reference-based Forgetting Objective

We propose a Reference-based forgetting objective that aligns the model’s predictive distribution on forgotten samples with a suitable reference distribution. Formally, we minimize the KL divergence

$$\min_{\theta} \mathcal{L}_{\text{forget}}(f_{\theta}, \mathcal{F}) = \mathbb{E}_{x_u \sim \mathcal{F}} [\text{KL}(p_{\theta}(y | x_u) || p_{\text{ref}}(y))], \quad (3.2)$$

where $p_\theta(y | x_u)$ denotes the model’s predictive distribution on forget sample x_u , and $p_{\text{ref}}(y)$ is a reference distribution representing unseen data.

The choice of reference distribution is crucial. We propose two strategies. We either use a uniform distribution as a principled baseline or leverage an unseen reference dataset, \mathcal{D}_{ref} , to improve stability. When a target dataset is used as reference, we compute the reference distribution as the expected predictive distribution over the samples as $p_{\text{ref}}(y) = \mathbb{E}_{x' \sim \mathcal{D}_{\text{ref}}} [p_\theta(y | x')]$. This approach aligns the model’s predictions on forgotten samples to exhibit uncertainty similar to truly unseen data.

3.3. Reference Distributions

Uniform Distribution (UniDist). The uniform distribution serves as a principled baseline requiring no additional data. Setting $p_{\text{ref}}(y) = 1/C$, where C is the number of outputs, encourages the model to treat forgotten samples with maximum uncertainty, as if it has no prior knowledge about them. Notably, minimizing the KL divergence to a uniform distribution is equivalent to maximizing the entropy of the predicted distribution, since $\text{KL}(p_\theta(y | x_u) \| U) = \log C - H(p_\theta(y | x_u))$, where $H(\cdot)$ denotes entropy. This objective is bounded between 0 and $\log C$, providing stable optimization dynamics.

Held-out Distribution (HeldDist). We compute an empirical reference distribution from an unseen held-out set \mathcal{D}_{ref} that was not part of the original training. This set should be disjoint from \mathcal{R} , \mathcal{F} , and any test or validation sets to represent a truly unseen distribution and avoid information leakage during evaluation. Using such a reference provides a more stable anchor that better preserves learned representations on \mathcal{R} .

3.4. Multi-Objective Alignment in Unlearning

We recall the multi-objective notions on which our analysis relies. For a vector objective $F(\theta) = (f_1(\theta), \dots, f_m(\theta))$ over parameter space Θ , we say that θ_a dominates θ_b when $f_i(\theta_a) \leq f_i(\theta_b)$ for all i and $f_j(\theta_a) < f_j(\theta_b)$ for at least one j . A point θ^* is Pareto-optimal when no other point in Θ dominates it, and the image of all Pareto-optimal points under F is the Pareto front. For smooth non-convex problems, the relevant first-order condition is Pareto stationarity. A point θ is Pareto-stationary when there exist non-negative weights $\alpha_1, \dots, \alpha_m$ with $\sum_i \alpha_i = 1$ that satisfy $\sum_{i=1}^m \alpha_i \nabla_\theta f_i(\theta) = 0$. Pareto stationarity is a necessary condition for Pareto optimality and is the strongest guarantee typically available for gradient-based MO methods such as Multiple Gradient Descent Algorithm (MGDA) [16, 17] and Jacobian descent [18] on deep networks.

We transform RAUL into the bi-objective problem

$$\theta_u = \arg \min_{\theta} (\mathcal{L}_{\text{retain}}(f_\theta, \mathcal{R}), \mathcal{L}_{\text{forget}}(f_\theta, \mathcal{F})), \quad (3.3)$$

in which both objectives are optimized jointly rather than collapsed into a single scalar. In this setting $m = 2$, so a Pareto-stationary parameter satisfies $\alpha_r g_r + \alpha_f g_f = 0$ for some $\alpha_r, \alpha_f \geq 0$ with $\alpha_r + \alpha_f = 1$, where $g_r = \nabla_\theta \mathcal{L}_{\text{retain}}$ and $g_f = \nabla_\theta \mathcal{L}_{\text{forget}}$. The two objectives are designed to be in conflict, so the inner product $\langle g_r, g_f \rangle$ is negative throughout training, and any unprojected combination of the two gradients can ascend one objective while descending the other. To solve Eq. 3.3 we employ Jacobian descent [18], which operates directly on the per-objective gradients without requiring a manual trade-off parameter. We stack the per-objective gradients into a Jacobian matrix and compute an aggregated update direction as

$$J = \begin{bmatrix} g_r^T \\ g_f^T \end{bmatrix}, \quad G = \mathcal{A}(J), \quad (3.4)$$

where \mathcal{A} is an aggregator function that aggregates gradients into a single gradient G to update the parameters at step t with learning rate η as

$$\theta_u^{t+1} = \theta_u^t - \eta G^t, \quad (3.5)$$

By contrast, the classic linear scalarization (LS) with a fixed weight λ drives the parameters toward a stationary point of the weighted sum $\lambda\mathcal{L}_{\text{retain}} + (1 - \lambda)\mathcal{L}_{\text{forget}}$. The resulting update depends on the fixed weight λ ; it may still carry a conflicting component when g_r and g_f oppose each other, and generally cannot reach the non-convex portion of the Pareto front. We use the Unconflicting Projection of Gradients (UPGrad) aggregator $\mathcal{A}_{\text{UPGrad}}$ [18], which projects each objective gradient onto the dual cone of all objective gradients and then averages the projected vectors. The fixed points of $\mathcal{A}_{\text{UPGrad}}$ therefore coincide with Pareto-stationary points of Eq. 3.3. Utilizing multiple points on the Pareto front is outside the scope of this work and motivates the use of multi-objective evolutionary algorithms [19] for unlearning as future work.

4. Experiments

We evaluate the effectiveness of our proposed RAUL framework through extensive experiments on image classification, comparing against representative unlearning baselines under varying forgetting ratios. Full details of the experimental setup are provided in Appendix A.

Table 1. Comparison of unlearning methods on CIFAR-10 with ResNet-18 under 10% and 50% random data forgetting. Results show $mean_{\pm std}$ over all runs in each forgetting setting. Values in parentheses indicate the performance gap to Retrain (lower is better).

Random 10%					
Methods	UA \uparrow	RA \uparrow	TA \uparrow	MIA \uparrow	Avg. Gap \downarrow
Retrain	5.68 \pm 0.27 (0.00)	100.00 \pm 0.00 (0.00)	94.32 \pm 0.03 (0.00)	13.13 \pm 1.16 (0.00)	0.00
FT	3.93 \pm 5.53 (5.06)	97.56 \pm 3.48 (2.44)	92.22 \pm 3.32 (2.47)	8.15 \pm 9.98 (23.79)	8.44
GA	11.29 \pm 31.73 (15.56)	88.73 \pm 31.75 (11.27)	84.02 \pm 29.86 (10.84)	1.93 \pm 3.33 (29.93)	16.90
IU	0.74 \pm 1.17 (5.13)	99.15 \pm 1.28 (0.85)	93.32 \pm 1.69 (1.26)	1.95 \pm 2.12 (17.00)	6.06
ℓ_1 -sparse	56.12 \pm 42.94 (53.00)	44.04 \pm 42.94 (55.96)	41.98 \pm 40.09 (52.42)	21.43 \pm 21.25 (15.22)	44.15
RL	18.75 \pm 24.47 (18.32)	82.38 \pm 24.85 (17.62)	78.75 \pm 22.22 (15.69)	32.28 \pm 30.04 (21.58)	18.30
SalUn	7.68 \pm 15.91 (10.28)	93.07 \pm 15.45 (6.93)	88.23 \pm 13.59 (6.21)	14.75 \pm 15.89 (11.48)	8.72
UniDist-RAUL-LS	0.97 \pm 0.99 (4.52)	99.20 \pm 1.02 (0.80)	91.56 \pm 2.97 (2.78)	7.44 \pm 0.49 (4.87)	3.24
UniDist-RAUL-UPGrad	1.43 \pm 1.41 (4.06)	99.11 \pm 1.38 (0.89)	91.95 \pm 2.77 (2.39)	9.52 \pm 0.72 (2.79)	2.53
HeldDist-RAUL-LS	7.30 \pm 2.71 (2.97)	99.24 \pm 0.16 (0.76)	92.09 \pm 0.48 (2.25)	17.21 \pm 2.68 (4.90)	2.72
HeldDist-RAUL-UPGrad	6.47 \pm 2.46 (2.34)	99.38 \pm 0.21 (0.62)	92.38 \pm 0.63 (1.96)	15.31 \pm 3.52 (4.00)	2.23
Random 50%					
Methods	UA \uparrow	RA \uparrow	TA \uparrow	MIA \uparrow	Avg. Gap \downarrow
Retrain	19.54 \pm 0.00 (0.00)	100.00 \pm 0.00 (0.00)	79.74 \pm 0.00 (0.00)	20.09 \pm 0.00 (0.00)	0.00
FT	3.89 \pm 5.65 (15.65)	97.77 \pm 3.53 (2.23)	91.71 \pm 4.21 (11.97)	6.87 \pm 8.48 (13.22)	10.77
GA	55.82 \pm 51.09 (51.91)	44.28 \pm 51.00 (55.72)	42.19 \pm 47.97 (49.44)	36.42 \pm 48.98 (39.66)	49.19
IU	20.49 \pm 26.35 (20.85)	79.75 \pm 26.11 (20.25)	74.97 \pm 24.05 (18.05)	19.81 \pm 22.20 (17.45)	19.15
ℓ_1 -sparse	45.12 \pm 43.20 (41.40)	54.96 \pm 43.06 (45.04)	52.32 \pm 40.17 (38.79)	42.01 \pm 38.21 (32.36)	39.40
RL	28.29 \pm 36.72 (30.28)	72.15 \pm 37.09 (27.85)	68.53 \pm 33.88 (27.28)	15.00 \pm 16.35 (14.48)	24.97
SalUn	10.36 \pm 22.06 (21.28)	89.92 \pm 22.09 (10.08)	84.84 \pm 19.25 (16.83)	10.99 \pm 10.82 (13.08)	15.32
UniDist-RAUL-LS	1.59 \pm 1.73 (17.95)	99.05 \pm 1.11 (0.95)	92.49 \pm 1.54 (12.75)	8.93 \pm 2.81 (11.16)	10.70
UniDist-RAUL-UPGrad	11.23 \pm 6.69 (8.31)	90.63 \pm 5.51 (9.37)	84.78 \pm 5.23 (5.04)	17.23 \pm 6.35 (4.28)	6.75
HeldDist-RAUL-LS	3.92 \pm 2.06 (15.61)	98.21 \pm 0.79 (1.79)	90.72 \pm 1.19 (10.98)	10.37 \pm 4.52 (9.72)	9.52
HeldDist-RAUL-UPGrad	1.97 \pm 2.19 (17.57)	98.89 \pm 1.29 (1.11)	92.12 \pm 1.95 (12.38)	14.41 \pm 6.04 (5.83)	9.22

4.1. Results and Analysis

Table 1 reports the performance of all methods under 10% and 50% random forgetting on CIFAR-10 with ResNet-18, measured by Unlearning Accuracy (UA), Retain Accuracy (RA), Test Accuracy (TA), and Membership Inference Attack accuracy (MIA). Our four RAUL

variants achieve the smallest average gaps to Retrain in both settings and outperform every baseline we consider.

At 10% forgetting, HeldDist-RAUL with UPGrad aggregator reaches the best average gap of 2.23, followed by UniDist-RAUL with UPGrad at 2.53, HeldDist-RAUL with LS at 2.72, and UniDist-RAUL with LS at 3.24. The strongest baselines are IU at 6.06 and FT at 8.44, while GA reaches 16.90 and ℓ_1 -sparse collapses to 44.15. These results indicate that error-maximization baselines tend to damage retention or fail to forget reliably. Aligning forget predictions with a reference distribution produces a better balance.

The privacy metrics further support our approach. The MIA scores of HeldDist-RAUL with UPGrad and HeldDist-RAUL with LS are 15.31 and 17.21, which are close to the Retrain value of 13.13. In contrast, GA drops to 1.93 and ℓ_1 -sparse inflates MIA to 21.43. Keeping MIA close to the retraining target is a strong signal that RAUL removes sample influence without the instability of adversarial forgetting.

The 50% forgetting scenario exposes the fragility of the baselines. GA collapses with an RA of 44.28 and a TA of 42.19, producing a gap of 49.19. ℓ_1 -sparse and RL also deteriorate with gaps of 39.40 and 24.97. UniDist-RAUL with UPGrad achieves the best gap of 6.75 and outperforms every baseline by a wide margin. HeldDist-RAUL with UPGrad and HeldDist-RAUL with LS follow with 9.22 and 9.52, while FT sits at 10.77.

The two RAUL families reveal a complementary trade-off. HeldDist-RAUL with UPGrad preserves high retention with an RA of 98.89 and a TA of 92.12 while using a conservative UA of 1.97, which makes it suitable when protecting retain accuracy matters most. UniDist-RAUL with UPGrad offers a more aggressive forgetting profile with UA 11.23 and RA 90.63, and it attains the lowest average gap. The standard deviations of all RAUL variants are also markedly lower than those of GA, ℓ_1 -sparse, and RL. This shows that reference-aligned forgetting produces stable updates even when half of the training data must be removed.

Effect of the gradient aggregator. A direct comparison of UPGrad against LS within each RAUL configuration isolates the effect of the aggregator. At 10% forgetting, UPGrad reduces the average gap from 3.24 to 2.53 for UniDist-RAUL and from 2.72 to 2.23 for HeldDist-RAUL. At 50% forgetting, UPGrad reduces the gap from 10.70 to 6.75 for UniDist-RAUL and from 9.52 to 9.22 for HeldDist-RAUL. The improvement is consistent across both reference strategies and both forgetting ratios. The numerical gains together with the lower variance of UPGrad runs suggest that the Pareto-stationary solutions reached by UPGrad correspond to better approximate unlearning behavior with smaller gaps to Retrain.

5. Conclusion

We presented RAUL, a framework that formulates machine unlearning as a bi-objective optimization problem and replaces adversarial loss maximization with alignment to a reference distribution representing unseen data. This reformulation reduces the gradient conflict between forgetting and retention and enables stable updates under Jacobian descent with the UPGrad aggregator. The two reference strategies offer a complementary advantage. UniDist-RAUL needs no extra data and delivers stronger forgetting, while HeldDist-RAUL delivers tighter retention when a held-out set is available. Since HeldDist-RAUL may not be available in all practical scenarios, UniDist-RAUL serves as a fallback. Constructing reference distributions from limited or synthetic data is an important direction for broader applicability.

In future work, we plan to extend multi-objective RAUL to generative models and LLMs. We also plan to study MOEAs [19] as alternatives to Jacobian descent, which produce more diverse Pareto-optimal solutions for unlearning.

Acknowledgement

This work is supported by Natural Sciences and Engineering Research Council of Canada.

References

- [1] European Parliament and Council of the European Union. *Regulation (EU) 2016/679 of the European Parliament and of the Council. Official Journal of the European Union, L119*, 1–88. 2016.
- [2] C. Shao and Y. Feng. *Overcoming catastrophic forgetting beyond continual learning: Balanced training for neural machine translation*. 2022.
- [3] K. Zhao, M. Kurmanji, G.-O. Bărbulescu, E. Triantafillou, and P. Triantafillou. “What makes unlearning hard and what to do about it”. In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 12293–12333.
- [4] Z. Pan, S. Zhang, Y. Zheng, C. Li, Y. Cheng, and J. Zhao. *Multi-Objective Large Language Model Unlearning*. IEEE, 2025.
- [5] X. Cheng, Z. Huang, W. Zhou, Z. He, R. Yang, Y. Wu, and X. Huang. *Remaining-data-free machine unlearning by suppressing sample contribution*. 2024.
- [6] J. Yao, E. Chien, M. Du, X. Niu, T. Wang, Z. Cheng, and X. Yue. “Machine Unlearning of Pre-trained Large Language Models”. In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by L.-W. Ku, A. Martins, and V. Srikumar. Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 8403–8419. DOI: [10.18653/v1/2024.acl-long.457](https://doi.org/10.18653/v1/2024.acl-long.457).
- [7] J. Li and S. Ghosh. “Random Relabeling for Efficient Machine Unlearning”. In: *arXiv preprint arXiv:2305.12320* (2023).
- [8] H. Zhao, B. Ni, J. Fan, Y. Wang, Y. Chen, G. Meng, and Z. Zhang. *Continual forgetting for pre-trained vision models*. 2024.
- [9] A. Thudi, G. Deza, V. Chandrasekaran, and N. Papernot. “Unrolling sgd: Understanding factors influencing machine unlearning”. In: *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2022, pp. 303–319.
- [10] A. Gohatkar, A. Achille, and S. 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*. 2020, pp. 9304–9312.
- [11] A. Warnecke, L. Pirch, C. Wressnegger, and K. Rieck. “Machine Unlearning of Features and Labels”. In: *arXiv preprint arXiv:2108.11577* (2021).
- [12] Z. Izzo, M. A. Smart, K. Chaudhuri, and J. Zou. “Approximate data deletion from machine learning models”. In: *International Conference on Artificial Intelligence and Statistics*. PMLR, 2021, pp. 2008–2016.
- [13] J. Jia, J. Liu, P. Ram, Y. Yao, G. Liu, Y. Liu, P. Sharma, and S. Liu. “Model sparsification can simplify machine unlearning”. In: *arXiv preprint arXiv:2304.04934* (2023).
- [14] C. Fan, J. Liu, Y. Zhang, E. Wong, D. Wei, and S. Liu. “Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation”. In: *arXiv preprint arXiv:2310.12508* (2023).
- [15] J. Wu and M. Harandi. “Munba: Machine unlearning via nash bargaining”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2025, pp. 4754–4765.
- [16] J.-A. Désidéri. “Multiple-gradient descent algorithm (MGDA) for multiobjective optimization”. In: *Comptes Rendus Mathématique* 350.5-6 (2012), pp. 313–318.
- [17] O. Sener and V. Koltun. “Multi-task learning as multi-objective optimization”. In: *Advances in Neural Information Processing Systems*. Vol. 31. 2018.
- [18] P. Quinton and V. Rey. “Jacobian Descent For Multi-Objective Optimization”. In: *arXiv preprint arXiv:2406.16232* (2024).
- [19] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. “A fast and elitist multiobjective genetic algorithm: NSGA-II”. In: *IEEE transactions on evolutionary computation* 6.2 (2002), pp. 182–197.

Appendix A. Experimental Setup

We evaluate RAUL on CIFAR-10 using ResNet-18 under two random-data forgetting scenarios with 10% and 50% of the training set removed (4,500 and 22,500 samples, respectively), starting from the same pretrained checkpoint in all runs. The reference Retrain model is trained from scratch for 182 epochs with an initial learning rate of 10^{-1} and a multi-step learning rate schedule that multiplies the learning rate by a factor $\gamma = 0.1$ at epochs 91 and 136. We compare against six unlearning baseline methods: Fine-Tuning (FT) [11], Gradient Ascent (GA) [9], Influence Unlearning (IU) [12, 13], ℓ_1 -sparse [13], Random Labeling (RL) [10], and SalUn [14] (random-label unlearning with a fixed saliency mask). All methods use SGD as the optimizer with fixed learning rate (no learning rate scheduler). Except for GA, which uses 2 epochs, baselines use 10 unlearning epochs. Learning rates are fixed per run according to the grids as follows. For FT, the five runs use learning rates $\{10^{-1}, 10^{-2}, 5 \times 10^{-2}, 5 \times 10^{-3}, 5 \times 10^{-3}\}$. For RL, they are $\{10^{-1}, 5 \times 10^{-2}, 10^{-2}, 5 \times 10^{-3}, 10^{-3}\}$. For GA, LRs are $\{10^{-4}, 5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}, 7 \times 10^{-3}, 9 \times 10^{-3}, 10^{-2}, 2 \times 10^{-2}\}$ at 10% forgetting and $\{5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}, 10^{-2}, 2 \times 10^{-2}\}$ at 50% forgetting. IU uses learning rate 10^{-3} with influence coefficient $\alpha \in \{1, 5, 10, 15, 20\}$. ℓ_1 -sparse sweeps $\{10^{-6}, 5 \times 10^{-6}, 10^{-5}, 2 \times 10^{-5}, 3 \times 10^{-5}, 5 \times 10^{-5}, 10^{-4}\}$. SalUn sweeps $\{10^{-4}, 5 \times 10^{-4}, 10^{-3}, 10^{-2}, 5 \times 10^{-2}\}$.

We configure our multi-objective RAUL framework with two gradient aggregators: fixed linear scalarization (LS) and UPGrad [18]. We use SGD as the optimizer with learning rates sweeping in $\{10^{-3}, 2.5 \times 10^{-3}, 5 \times 10^{-3}, 7.5 \times 10^{-3}, 10^{-2}, 5 \times 10^{-2}\}$ without any learning rate scheduling.

We report UA, RA, TA, and MIA following the benchmarking protocol of [14]. Performance is measured by the gap to Retrain, and a smaller gap indicates better approximate unlearning.