DECAF: De-Clustering for Adaptive Representational Unlearning

Anjie Le [1] **Can Peng** [1] **Hongcheng Guo** [2] **J. Alison Noble** [1]

Abstract

Machine unlearning, which aims to remove the influence of specific training data from a trained model, is a key requirement for privacy, accountability, and adaptive deployment. We argue that many unlearning methods are vulnerable to a simple *clustering attack*, which can recover class structure in an unsupervised manner, limiting their suitability for continual deployment where removal requests must be handled reliably on demand. To address this, we propose **DECAF** (DE-Clustering for Adaptive Forgetting), a post-hoc method that operates only on the forget set and is designed to break the cluster. DECAF combines input noise, confidence suppression, and entropy-based output diversification to disrupt the residual feature-space structure associated with forgotten data. On CIFAR-10 with ResNet-18, DECAF attains 0.10% forget-class accuracy, 79.4% retain accuracy, and an AUS of 0.88 surpassing all other baselines. In cluster-based analysis, it attains performance comparable to that of unlearning methods that use the full training set, while being significantly more efficient. Code will be released.

1. Introduction

Foundation models are increasingly deployed in dynamic environments where their behavior must be updated after training. In such settings, adaptation is not limited to adding new capabilities: models must also remove outdated, harmful, or private information in a targeted and computationally sustainable manner. This requirement naturally arises in adaptive deployment, where data distributions and usage constraints evolve over time. As a result, machine unlearning becomes a core component of continual adaptation, rather than merely a post-deployment compliance tool.

A key instance of this problem is deliberate forgetting: given a trained model and a target forget set, the goal is to remove the influence of those examples while preserving utility. Retraining from scratch without the forget set (Cao & Yang, 2015) is the standard solution but is impractical at scale, motivating approximate unlearning methods that update a trained model while preserving performance on the remaining retain set (Bourtoule et al., 2021a; Lee et al., 2025); however, many such methods assume access to the original data or pre-training corpora, which is often unrealistic (Voigt & Von dem Bussche, 2017; PIPL, 2021; CCPA, 2018; Bommasani et al., 2021; Carlini et al., 2020). This motivates the post-hoc, forget-only setting, where adaptation must be performed using only the data to be removed.

Beyond this practical limitation, we identify a more fundamental issue in existing unlearning methods. Unlearning is typically evaluated by reduced classification accuracy on the forget set, but this metric alone does not guarantee that the underlying information has been removed from the model (Golatkar et al., 2020). Consistent with prior evidence that class signal can live on in features (Izzo et al., 2021; Ginart et al., 2019), we find that common unlearning methods still leave the forget set tightly clustered in the penultimate layer, indicating that class-discriminative information is preserved. This enables a simple clustering attack, in which an adversary can recover class structure from latent features with unsupervised clustering, revealing that many methods suppress outputs without erasing representations.

To address this, we propose **DECAF** (*DE-Clustering for Adaptive Forgetting*), a lightweight post-hoc method that operates solely on the forget set. DECAF combines input noise, confidence suppression, and entropy-based output diversification, with each component acting on a complementary aspect of cluster geometry: they increase intra-class dispersion, weaken class-aligned representations, and encourage redistribution across the remaining classes, thereby explicitly disrupting residual feature-space structure and preventing re-clustering.

Our contributions are threefold. *(i)* We characterize a clustering-based failure mode: many post-hoc unlearning methods lower forget-class accuracy but leave the forget set class-structured in the penultimate layer, so a simple unsupervised clustering procedure can still recover class information, motivating evaluation beyond standard clas-

[1]Institute of Biomedical Engineering, University of Oxford, UK [2]Fudan University. Correspondence to: Anjie Le <anjie.le@eng.ox.ac.uk>, J. Alison Noble <alison.noble@eng.ox.ac.uk>.

Presented at the ICML 2026 Workshop "Continual Adaptation at Scale: Towards Sustainable AI".

sification tests. *(ii)* We use clustering quality metrics to assess whether latent structure of the forget set has been disrupted, complementing standard unlearning measures. *(iii)* We propose **DECAF** as a lightweight forget-only alternative to retraining that directly targets declustering in feature space. On CIFAR-10 with ResNet-18, it yields a strong aggregate forgetting–utility trade-off and declustering competitive with retrain-from-scratch, using only the forget set and far less compute than retraining and typical retain-set fine-tuning.

2. Clustering Attack

Most existing unlearning methods assess information removal by measuring classification degradation on the forget set D_f. However, poor classification performance on D_f does not necessarily imply that its internal representation has been erased. In practice, we observe that even when the model misclassifies samples in D_f, their penultimate-layer features $z = \phi_{\theta'}(x)$ often remain tightly clustered and class-discriminative. Prior work (Izzo et al., 2021; Golatkar et al., 2020; Ginart et al., 2019) has shown that such residual structure can often be recovered by training a linear probe on Z_f. While effective, linear probing requires data labels and an additional training stage, which is costly and sometimes impractical due to limited access to complete data. Our goal is instead to diagnose residual memorization directly from feature geometry, without additional fine-tuning.

2.1. Representation-Level Diagnosis

Let $\mathcal{C}$ denote the set of classes and $Z_f^{(c)} = \{\phi_{\theta'}(x) \mid x \in D_f, y(x) = c\}$ denote the latent features of the forget set after unlearning. We observe that for many baselines, these features remain compact within class and well separated across classes, i.e., for forget class c,

$$\mathbb{E}_{z,z' \in Z_f^{(c)}} \|z - z'\|^2 \ll \mathbb{E}_{z \in Z_f^{(c)}, z' \in Z_f^{(c')}} \|z - z'\|^2, \quad c' \neq c,$$

indicating that substantial class structure persists in feature space. In other words, unlearning may suppress the classifier output without fully removing the underlying geometry of the forgotten data.

2.2. Clustering-Based Evaluation

Accordingly, **we propose a clustering-based analysis** that directly examines the structure of latent representations as a diagnostic for residual memorization. Specifically, we cluster Z_f under two distinct settings in an unsupervised manner, with number of classes $k = |\mathcal{C}|$ and $k = |\mathcal{C}| - 1$ respectively, corresponding to whether the forgotten class remains a separable cluster or has been dispersed into the remaining classes. We apply multiple clustering algorithms (e.g., DBSCAN, K-Means, GMM) and choose the best performing one, then evaluate the resulting cluster assignments

Table 1. Clustering quality of D_f feature representations before and after unlearning. Lower Silhouette and CH scores and higher DB scores after unlearning indicate disrupted clustering and better forgetting. "Safe?" indicates whether a clustering attack can recover class identity.

Method	Silhouette ↑			Calinski–Harabasz ↑			Davies–Bouldin ↓			Safe?
	$k=9$	$k=10$	Δ	$k=9$	$k=10$	Δ	$k=9$	$k=10$	Δ	
Methods requiring the retain set										
Retrain	0.106	0.102	-0.004	1136.6	1048.5	-88.1	2.141	2.232	+0.091	✓
FT	0.178	0.179	+0.001	1326.5	1267.5	-59.0	1.688	1.726	+0.038	✗
FCS	**0.218**	**0.205**	-0.013	**1460.2**	**1414.0**	-46.2	**1.549**	**1.610**	+0.061	✓
MSG	0.085	0.080	-0.005	642.2	589.5	-52.7	2.579	2.568	-0.011	✗
Methods using forget set only										
GA	0.052	0.049	-0.003	310.5	241.7	-68.8	3.021	2.997	-0.024	✗
DECAF	0.185	0.183	-0.002	1340.8	1283.5	-57.3	1.682	1.700	+0.018	✓

$\tilde{y}$ using three metrics:

- **Silhouette Score:**

$$\mathrm{Sil}(Z_f, \tilde{y}) = \frac{1}{n} \sum_{i=1}^{n} \frac{b(i) - a(i)}{\max\{a(i), b(i)\}},$$

  where $a(i)$ is the average distance between points in the same cluster, and $b(i)$ is the minimum average distance to points in any other cluster, measuring how well-separated and coherent each cluster is.

- **Calinski–Harabasz (CH) Index:**

$$\mathrm{CH}(Z_f, \tilde{y}) = \frac{\mathrm{Tr}(B_k)}{\mathrm{Tr}(W_k)} \cdot \frac{n - k}{k - 1},$$

  where $\mathrm{Tr}(B_k)$ is the between-cluster dispersion, $\mathrm{Tr}(W_k)$ is the within-cluster dispersion, quantifing the ratio of between-cluster to within-cluster variance.

- **Davies–Bouldin (DB) Index:**

$$\mathrm{DB}(Z_f, \tilde{y}) = \frac{1}{k} \sum_{i=1}^{k} \max_{j \neq i} \left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)} \right),$$

  where σ_i is the average distance of points in cluster i to its centroid c_i, and $d(c_i, c_j)$ is the distance between centroids, capturing intra-cluster tightness versus inter-cluster separation.

These metrics capture complementary aspects of clustering quality. Higher Silhouette and CH scores and lower DB scores indicate better-defined cluster structure.

2.3. Clustering Analysis

Residual clustering implies incomplete forgetting. As shown in Table 1, clustering-based analysis reveals several findings. Methods including FT, MSG and GA continue to exhibit recoverable structure, indicating that forgotten samples remain clustered in latent space despite reduced

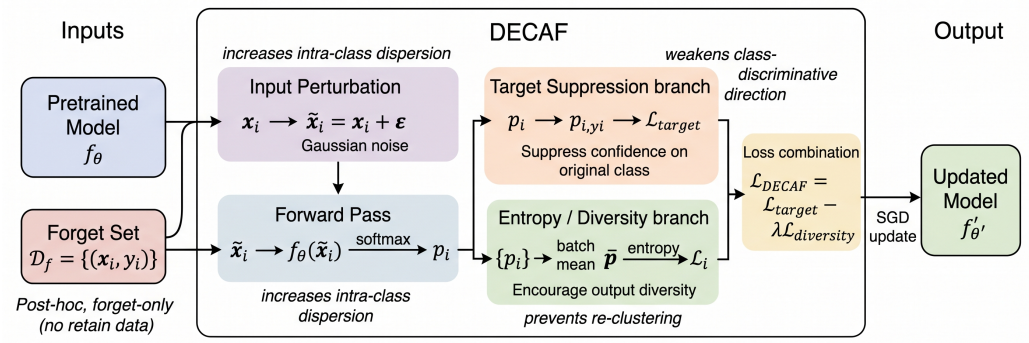

Figure 1. **DECAF overview.** Given a pretrained model and a forget set D_f, DECAF performs post-hoc unlearning using only D_f. Inputs are perturbed with noise, passed through the model, and optimized with target suppression and entropy-based diversity, $\mathcal{L}_{\text{DECAF}} = \mathcal{L}_{\text{target}} - \lambda\mathcal{L}_{\text{diversity}}$.

classification accuracy. This suggests that these approaches primarily suppress the output layer without fully removing underlying feature representations. This analysis complements classification-based evaluation by revealing whether unlearning has truly disrupted the latent structure of D_f, rather than merely weakening the final decision boundary, which motivates our method that focuses on deeper representational forgetting.

3. Method

DECAF is a post-hoc, forget-only unlearning method that operates using only the forget set D_f. It is designed to disrupt the three complementary properties of cluster structure. Input perturbation reduces cluster compactness, target suppression weakens separation from the original class boundary, and entropy regularization prevents forgotten samples from collapsing into a new surrogate cluster (Fig. 1).

Given a forget example $(x_i, y_i) \in D_f$, we first perturb the input with Gaussian noise:

$$\tilde{x}_i = x_i + \epsilon, \qquad \epsilon \sim \mathcal{N}(0, \sigma^2 I), \tag{1}$$

which disrupts low-level cues associated with memorized representations. From a clustering perspective, this increases the within-class dispersion of forgotten features and therefore weakens the compactness of the forget-class cluster.

Next, we suppress the model's confidence in the forget-class label via

$$\mathcal{L}_{\text{target}} = \frac{1}{N}\sum_{i=1}^{N} p_{i,y_i}, \tag{2}$$

where p_{i,y_i} is the softmax probability assigned to the ground-truth forget class y_i. Minimizing this objective discourages the model from preserving the original class-discriminative direction of forgotten samples, thereby destabilizing their shared representation and reducing their compactness and separability in feature space.

However, confidence suppression alone can lead to a degenerate solution in which forgotten samples are reassigned to a small subset of retained classes, forming a new cluster rather than being dispersed. To prevent this, we encourage diversity in the batch-average output distribution. Let

$$\bar{p} = \frac{1}{N}\sum_{i=1}^{N} \text{softmax}(f_\theta(\tilde{x}_i))$$

denote the mean prediction over a minibatch. We define the diversity objective as

$$\mathcal{L}_{\text{diversity}} = -\sum_{c=1}^{C} \bar{p}_c \log \bar{p}_c, \tag{3}$$

which promotes a more diffuse allocation of predictions across the remaining classes. In terms of cluster geometry, this discourages re-clustering and increases the overlap of forgotten samples with the rest of the feature space.

The final DECAF objective combines these two terms:

$$\mathcal{L}_{\text{DECAF}} = \mathcal{L}_{\text{target}} - \lambda\mathcal{L}_{\text{diversity}}, \tag{4}$$

where λ controls the strength of output diversification. We optimize this objective using SGD for several epochs on D_f only. This makes DECAF a lightweight and practical approach to forget-only representation-level unlearning.

4. Experiment

4.1. Experimental Setup

Setup. We evaluate DECAF on CIFAR-10 and randomly select one class as the forget set D_f, with the remaining samples forming the retain set D_r. Model adapts a ResNet-18 backbone. Unlearning methods are applied post-hoc to the trained model.

Baselines. We compare DECAF with several post-hoc unlearning methods: Retrain-from-Scratch (Gold), which retrains using only D_r; Gradient Ascent (GA), which maximizes loss on D_f; Fine-Tuning (FT), which continue training the model for a few more epochs with D_r; Masked Small

Table 2. Unlearning performance on CIFAR-10 with ResNet-18. Lower Forget Acc, MIA, and AIN indicate stronger forgetting, while higher Retain Acc and AUS reflect better utility. Bold entries indicate best results.

Method	Forget Acc ↓	Retain Acc ↑	AUS ↑	MIA ↓	AIN ↑	Time (s)
Retrain	0.00	77.29	0.86	51.70	1.00	1113.72
Methods requiring the retain set						
FT	30.80	82.48	0.72	59.70	0.10	873.81
FCS	9.00	**84.49**	0.77	69.30	0.04	139.78
MSG	15.20	71.74	0.74	55.80	0.02	106.09
Methods using forget set only						
GA	18.50	37.11	0.44	**54.80**	0.10	**2.42**
DECAF	**0.10**	79.42	**0.88**	58.40	**0.38**	9.55

Gradient (MSG) (Cadet et al., 2024), which suppresses updates on salient features; and Forget–Contrast–Strengthen (FCS) (Cadet et al., 2024), a method based on contrastive learning of representations.

Metrics. We evaluate unlearning performance using standard metrics, including forget accuracy (lower is better), retain accuracy (higher is better), membership inference attack (MIA), and runtime. We also report an aggregated unlearning score (AUS) (Cotogni et al., 2024; Li et al., 2025) to summarize the trade-off between forgetting and utility, and anamnesis index (AIN) (Chundawat et al., 2023) which measures residual memory of the forget data.

4.2. Main Results

Table 2 presents the unlearning performance of DECAF alongside several baseline methods.

Forgetting efficacy and utility trade-off. DECAF achieves the lowest forget accuracy (0.10%) among all approximate methods, effectively eliminating class-specific predictive power. At the same time, it retains strong utility with a 79.4% accuracy on the retain set, outperforming GA and MSG, and only slightly trailing FT and FCS. These two trends combine into an AUS of 0.88, higher than all baselines and even surpassing the retrain-from-scratch gold standard (0.86), demonstrating DECAF's superior ability to forget cleanly while preserving generalisation.

Latent memorisation and inference risk. In terms of MIA, DECAF balances between robustness and privacy, achieving stronger protection against MIA than FT and FCS, while preserving higher utility than GA. DECAF also obtains the highest AIN (0.38), indicating minimal residual influence from the forget set in internal representations.

Efficiency. In terms of runtime, DECAF completes unlearning in under 10 seconds, which is over 100× faster than retraining and significantly more efficient than high-performing methods like FT and FCS. While GA is technically faster, its poor retain accuracy and low AUS indicate severe degradation of model utility, rendering it impractical for real-world use.

(a) Original *(b)* Fine-Tuning (FT) *(c)* DECAF (Ours)

Figure 2. t-SNE visualization of penultimate-layer features, where different colors indicate different predicted labels. The original model exhibits well-separated clusters. After fine-tuning, the forget class (dark blue) remains clustered, indicating residual structure. In contrast, DECAF disperses the forget-class representations.

Table 3. Ablation study of DECAF components. Each component contributes meaningfully; removing any harms forgetting quality and increases vulnerability to attacks.

Variant	FAcc ↓	RAcc ↑	AUS ↑	MIA ↓
Full	**0.10**	79.42	0.88	**58.40**
w/o CS	6.70	74.97	0.79	60.90
w/o Entropy	**0.10**	**80.24**	**0.89**	63.10
w/o Noise	0.50	77.67	0.86	62.10

Declustering analysis. As shown in Table 1, DECAF effectively disrupts cluster structure. Compared to baselines, DECAF consistently reduces cluster separability across metrics, indicating that forgotten samples are no longer organized into a distinct group. A t-SNE visualisation is included in Fig. 2 to illustrate this effect.

4.3. Ablation Study

To understand the contribution of each component in DECAF, we perform an ablation study by removing individual mechanisms and measuring the impact on forgetting quality and robustness. As shown in Table 3, each component of DECAF plays a complementary role. Removing confidence suppression leads to the largest drop in AUS, indicating its importance for utility. Removing entropy improves retain accuracy but increases MIA, suggesting weaker robustness. Similarly, removing noise slightly improves retention but worsens MIA, highlighting its role in disrupting feature structure. Overall, the full DECAF configuration best balances forgetting, utility, and robustness.

5. Conclusion

Our experiments show that DECAF effectively removes both predictive and representational traces of the forget set by directly disrupting feature-space structure, rather than merely suppressing outputs. This leads to more diffuse and less entangled representations, which may benefit downstream adaptation under distribution shift or continual learning. Future work could evaluate DECAF on broader benchmarks and further examine these effects.

References

Basaran, U. Y., Ahmed, S. M., Roy-Chowdhury, A., and Guler, B. A certified unlearning approach without access to source data. *arXiv preprint arXiv:2506.06486*, 2025.

Baumhauer, T., Schöttle, P., and Zeppelzauer, M. Machine unlearning: Linear filtration for logit-based classifiers. *Machine Learning*, 2022.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Bourtoule, L., Chandrasekaran, V., Choquette-Choo, C. A., Jia, H., Travers, A., Zhang, B., Lie, D., and Papernot, N. Machine unlearning. In *2021 IEEE symposium on security and privacy (SP)*, pp. 141–159. IEEE, 2021a.

Bourtoule, L., Chandrasekaran, V., Choquette-Choo, C. A., Jia, H., Travers, A., Zhang, B., Lie, D., and Papernot, N. Machine unlearning. In *S&P*, 2021b.

Cadet, X. F., Borovykh, A., Malekzadeh, M., Ahmadi-Abhari, S., and Haddadi, H. Deep unlearn: Benchmarking machine unlearning for image classification. *arXiv preprint arXiv:2410.01276*, 2024. NeurIPS 2024 Datasets and Benchmarks Track.

Cao, Y. and Yang, J. Towards making systems forget with machine unlearning. In *S&P*, 2015.

Carlini, N., Tramer, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, U., et al. Extracting training data from large language models.(dec. *arXiv preprint arXiv:2012.07805*, 2020.

CCPA. California consumer privacy act of 2018, 2018. URL https://leginfo.legislature.ca.gov/faces/billTextClient.xhtml?bill_id=201720180AB375. California Civil Code §1798.100 et seq.

Chen, Y., Xiong, J., Xu, W., and Zuo, J. A novel online incremental and decremental learning algorithm based on variable support vector machine. *Cluster Computing*, 2019.

Chowdhury, S. B. R., Choromanski, K., Sehanobish, A., Dubey, A., and Chaturvedi, S. Towards scalable exact machine unlearning using parameter-efficient fine-tuning. *arXiv preprint arXiv:2406.16257*, 2024.

Chundawat, V. S., Tarun, A. K., Mandal, M., and Kankanhalli, M. Zero-shot machine unlearning. *IEEE Transactions on Information Forensics and Security*, 18:2345–2354, 2023.

Cotogni, M., Bonato, J., Sabetta, L., Pelosin, F., and Nicolosi, A. DUCK: Distance-based Unlearning via Centroid Kinematics, May 2024. URL http://arxiv.org/abs/2312.02052. arXiv:2312.02052 [cs].

Ginart, A., Guan, M., Valiant, G., and Zou, J. Y. Making ai forget you: Data deletion in machine learning. In *NeurIPS*, 2019.

Golatkar, A., Achille, A., and Soatto, S. Eternal Sunshine of the Spotless Net: Selective Forgetting in Deep Networks, March 2020. URL http://arxiv.org/abs/1911.04933. arXiv:1911.04933 [cs].

Graves, L., Nagisetty, V., and Ganesh, V. Amnesiac machine learning. In *AAAI*, 2021.

Guo, C., Goldstein, T., Hannun, A. Y., and van der Maaten, L. Certified data removal from machine learning models. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119, pp. 3832–3842, 2020.

Guo, Q., Liu, X., and Tao, D. Efficient attribute unlearning: Selective removal from feature representations. *arXiv preprint arXiv:2202.13295*, 2022.

Izzo, Z., Smart, M. A., Chaudhuri, K., and Zou, J. Approximate data deletion from machine learning models. In *AISTATS*, 2021.

Jia, J., Liu, J., Ram, P., Yao, Y., Liu, G., Liu, Y., Sharma, P., and Liu, S. Model sparsity can simplify machine unlearning. In *Proceedings of the 37th Conference on Neural Information Processing Systems (NeurIPS)*, 2023.

Jia, J., Liu, J., Yao, Y., Liu, G., and Liu, S. Revisiting machine unlearning with dimensional alignment. *arXiv preprint arXiv:2407.17710*, 2024.

Le, A., Peng, C., Liu, Y., and Noble, J. A. Pour: A provably optimal method for unlearning representations via neural collapse. *arXiv preprint arXiv:2511.19339*, 2025.

Lee, T.-Y., Park, S., Jeon, M., Hwang, H., and Park, G.-M. Esc: Erasing space concept for knowledge deletion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5010–5019, 2025.

Li, N., Zhou, C., Gao, Y., Chen, H., Zhang, Z., Kuang, B., and Fu, A. Machine unlearning: Taxonomy, metrics, applications, challenges, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

Moon, H., Kwon, H., Kim, S., and Yoo, J. Feature unlearning for pre-trained gans and vaes. *arXiv preprint arXiv:2303.05699*, 2023.

PIPL. Personal information protection law of the people's republic of china, 2021. URL https://www.chinalawtranslate.com/en/personal-information-protection-law-of-the-peoples-republic-of-china/. Adopted at the 30th Meeting of the Standing Committee of the Thirteenth National People's Congress.

Qin, J., Lu, R., Ma, J., and Zheng, W.-S. Machine unlearning on pre-trained models by residual feature alignment using lora. *arXiv preprint arXiv:2411.08443*, 2024.

Schelter, S. "Amnesia" – Towards Machine Learning Models That Can Forget User Data Very Fast. In *Workshop on Applied AI for Database Systems and Applications (AIDB)*, Los Angeles, CA, June 2019.

Sun, G., Manakul, P., Zhan, X., and Gales, M. Unlearning vs. obfuscation: Are we truly removing knowledge? *arXiv preprint arXiv:2505.02884*, 2025.

Voigt, P. and Von dem Bussche, A. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 2017.

Xu, H., Zhu, T., Zhang, L., Zhou, W., and Yu, P. S. Machine unlearning: A survey. *arXiv preprint arXiv:2306.03558*, jun 2023.

Zhou, Y., Zheng, D., Mo, Q., Lu, R., Lin, K.-Y., and Zheng, W.-S. Decoupled distillation to erase: A general unlearning method for any class-centric tasks. *arXiv preprint arXiv:2503.23751*, March 2025. Preprint, submitted March 31.

A. Related Work

A.1. Machine Unlearning

Machine Unlearning (MU) (Bourtoule et al., 2021b; Cao & Yang, 2015; Voigt & Von dem Bussche, 2017) aim to remove the influence of specific data from trained models to support privacy requirements such as the "right to be forgotten." While retraining a model from scratch without the target data is a straightforward solution (Graves et al., 2021), it is computationally expensive and often impractical. Early MU efforts focused on traditional machine learning models, such as linear regression (Baumhauer et al., 2022), k-means (Ginart et al., 2019), and SVMs (Chen et al., 2019), where convexity and simplicity allow exact or approximate unlearning. However, these approaches do not generalize well to complex visual data, and they are also incompatible with deep neural networks (DNNs), which lack the tractable properties exploited by traditional models (Zhou et al., 2025). Recent DNN-based MU methods, as categorized by the survey in (Xu et al., 2023), are broadly divided into two categories: Data Reorganization and Model Manipulation. Data Reorganization methods (Graves et al., 2021; Basaran et al., 2025; Bourtoule et al., 2021a) focus on modifying the training data to reduce the model's dependence on specific samples. Some approaches (Graves et al., 2021; Sun et al., 2025) perturb data labels or features to obscure the influence of the target samples; some approaches (Bourtoule et al., 2021a; Chowdhury et al., 2024) partition the dataset into disjoint subsets to isolate the effect of forget requests—though such methods often impose restrictive assumptions on the training process; and others (Basaran et al., 2025; Cao & Yang, 2015) replace the original data with transformed surrogates to facilitate more efficient unlearning. Model Manipulation methods (Cadet et al., 2024; Jia et al., 2023) directly modify model parameters after training. These techniques either adjust the weights to counteract the influence of the target samples (Guo et al., 2020; Schelter, 2019; Cadet et al., 2024), or prune parameters most correlated with the forget set (Jia et al., 2023). In this work, we focus on deep learning–based unlearning under practical constraints, where direct retraining or restrictive training assumptions are infeasible.

A.2. Unlearning Feature Analysis

Recently, Feature Manipulation has emerged as a promising direction in the unlearning literature. These methods aim to remove the influence of forgettable data by modifying the learned feature representations, rather than relying on retraining or directly altering model weights. They leverage knowledge distillation (Zhou et al., 2025) or feature alignment (Jia et al., 2024) to disentangle and suppress feature activations associated with the forget set while retaining general knowledge. For example, Zhou et al. (Zhou et al., 2025) propose a decoupled distillation framework that decomposes the unlearning objective into forgetting and retention terms, allowing for masked knowledge transfer that preserves class-discriminative features unrelated to the forget class. Similarly, Jia et al. (Jia et al., 2024) introduce dimensional alignment as a mechanism to align the latent manifolds of retained and forgotten samples, enabling unlearning via geometric regularization. Qin et al. (Qin et al., 2024) further explore residual feature erasure in pre-trained models by applying low-rank adaptation (LoRA) to isolate and update only the task-specific residuals associated with the forget set. Le et al. (Le et al., 2025) study selective removal of unwanted information while preserving useful representations, emphasizing the importance of fine-grained control in feature space. Other works target fine-grained attribute unlearning by suppressing sensitive features from the internal representations (Guo et al., 2022), or by editing latent codes in generative models (Moon et al., 2023). These methods provide a generalizable and architecture-agnostic framework for unlearning, particularly suited to class-centric and representation-focused tasks.

However, despite their emphasis on feature-space manipulation, most existing methods lack a rigorous, quantitative evaluation of how well the feature representations of the forget set have been erased. In contrast, we propose an unsupervised, clustering-based metric suite that explicitly measures the residual structure of forgotten features in the latent space—offering a principled and label-free diagnostic for representational forgetting.