

# Stake the Points: Structure-Faithful Instance Unlearning

Kiseong Hong  
 Chung-Ang University  
 ghdr1tjd@cau.ac.kr

JungKyo Shin  
 Chung-Ang University  
 neo293@cau.ac.kr

Eunwoo Kim\*  
 Chung-Ang University  
 eunwoo@cau.ac.kr

## Abstract

Machine unlearning (MU) addresses privacy risks in pre-trained models. The main goal of MU is to remove the influence of designated data while preserving the utility of retained knowledge. Achieving this goal requires preserving semantic relations among retained instances, which existing studies often overlook. We observe that without such preservation, models suffer from progressive structural collapse, undermining both the deletion–retention balance. In this work, we propose a novel structure-faithful framework that introduces stakes, i.e., semantic anchors that serve as reference points to maintain the knowledge structure. By leveraging these anchors, our framework captures and stabilizes the semantic organization of knowledge. Specifically, we instantiate the anchors from language-driven attribute descriptions encoded by a semantic encoder (e.g., CLIP). We enforce preservation of the knowledge structure via structure-aware alignment and regularization: the former aligns the organization of retained knowledge before and after unlearning around anchors, while the latter regulates updates to structure-critical parameters. Results from image classification, retrieval, and face recognition show average gains of 32.9%, 22.5%, and 19.3% in performance, balancing the deletion–retention trade-off and enhancing generalization.

## 1. Introduction

Requests to erase sensitive or personally related learned information are rising due to stronger data protection regulations [35]. A straightforward solution is to retrain from scratch without the designated data [4]. However, this is often impractical due to limited data access and the high cost of large-scale retraining [4, 5]. These challenges fuel the rapid growth of machine unlearning (MU), which aims to selectively remove the influence of designated data while preserving the utility of retained knowledge [12, 19, 31, 33].

Research on MU advances from exact retraining ap-

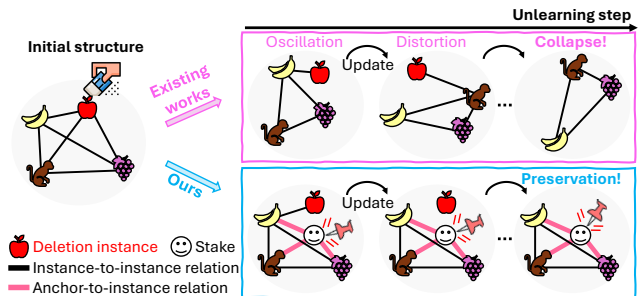


Figure 1. Conceptual illustration of structural collapse and our structure-faithful remedy in MU. Existing works delete the designated instance but ignore semantic relations between retained instances. During unlearning, model updates induce oscillations in the representation space. Without relational awareness, these oscillations distort the instance-level semantic relations (e.g., a monkey embedding drifting toward grape while moving away from banana), collapsing the original knowledge organization. We introduce semantic anchors (i.e., stakes) that preserve key relational constraints by keeping the relative positions between anchors and retained instances (i.e., anchor-to-instance relations).

proaches [4, 13, 36] to more efficient approximate methods [6, 17, 22, 31]. To improve efficiency, these approximate methods update parameters affected by the forget set [10], leverage weight saliency [9], or apply selective perturbations [12], avoiding full retraining. Recently, growing attention has been devoted toward instance-level unlearning [6, 31], as real-world deletion requests typically involve specific individuals rather than entire categories [24, 34].

Despite these advances, the core challenge of MU lies in preserving structure, particularly the semantic organization among the retained instances. This is because knowledge in deep models is encoded not in isolated knowledge but in the semantic relations that contextualize them [3, 15, 42]. However, existing methods [6, 12] largely neglect this aspect. As shown in Figure 1, removing an instance (e.g., apple) can distort the semantic relations between other instances (e.g., banana or grape). Such distortion through the representation space disrupts coherence, leading to progressive structural collapse. This highlights the necessity of structure-preserving unlearning to maintain relational structure.

\*Corresponding author.

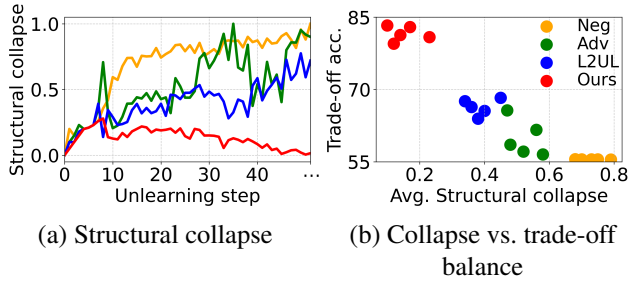


Figure 2. Structural collapse and its impact on the deletion–retention balance on CIFAR-100 (256 designated instances). (a) We observe structural collapse during unlearning in prior MU studies (Neg [12], Adv [6], and L2UL [6]). We quantify it as the change in affinities between retained-instance embeddings and anchors before and during unlearning. Larger values indicate greater semantic shift. (b) We compare average structural collapse (mean over unlearning steps) with the trade-off accuracy defined as the retention–deletion accuracy gap. Each dot denotes a random trial; Neg lies along the x-axis, with accuracy below 5%.

Furthermore, we empirically observe that structural collapse is closely associated with the deletion–retention trade-off. The existing works (orange, green, and blue in Figure 2 (a)) [6, 12] progressively exhibit structural collapse as unlearning proceeds. This collapse arises from drastic embedding shifts, which make it difficult to preserve their semantic connections. As illustrated in Figure 2 (b), the average structural collapse is negatively correlated with accuracy. Notably, reducing structural collapse mitigates representational drift, enabling the model to retain meaningful semantic relations throughout unlearning. This highlights that structural preservation plays a decisive role for MU.

In this work, we propose a novel structure-faithful unlearning framework. The core idea is to introduce semantic anchors, which prevent retained instances from drifting during unlearning by binding each instance to these anchors (i.e., stakes). This preserves semantic organization and prevents structural collapse (see Figure 1). Specifically, the semantic anchors are instantiated from the class attributes, such as texture, shape, or typical context. These attributes are used to prompt a large language model, which generates human-interpretable descriptions [26]. The resulting descriptions are encoded with a semantic encoder (e.g., CLIP) to form anchors. Based on these anchors, we define structure as the affinities between embeddings of retained instances and semantic anchors (as illustrated by the anchor–instance relations in Figure 1).

Our objective is to ensure that this structure remains preserved throughout unlearning. To this end, we introduce two complementary constraints. The first enforces structure-aware alignment to maintain structural consistency before and after unlearning, thereby ensuring semantic consistency among retained instances. The second ap-

plies structure-aware regularization, which penalizes model updates in proportion to their structural importance. This prioritizes the preservation of parameters most critical to semantic consistency. While existing methods neglect structure preservation, our framework explicitly stabilizes this organization and avoids collapse by jointly aligning representations and regularizing updates.

To evaluate the effectiveness of ours, we conduct experiments on three tasks: image classification, face recognition, and image-to-image retrieval. Across all tasks, our method consistently outperforms existing methods, with the margin widening as the number of instances to unlearn increases. The results show that retaining semantic structure during unlearning not only mitigates structural collapse but also enhances deletion–retention balance and performance. The main contributions of this work are as follows:

- We conceptualize structural preservation in instance unlearning and observe that it plays a decisive role in improving the deletion–retention balance.
- We propose a structure-faithful unlearning framework that leverages semantic anchors to preserve the semantic organization of knowledge.
- We present two constraints that retain the relational structure between embeddings and anchors and regularize updates to structure-sensitive model parameters.
- Extensive experimental results show that our method consistently improves both deletion–retention balance and generalization performance across diverse tasks.

## 2. Related Work

Machine unlearning (MU) removes the influence of specified data while preserving performance on the remainder [4]. The literature falls into two lines of research: exact unlearning [4, 13, 36] and approximate unlearning [6, 10, 12]. Exact MU [4, 13, 36] aims to produce a model that behaves as if the deleted samples were never used, typically by retraining on data excluding those samples. Although this guarantees complete removal, it is computationally expensive and requires access to retained data, which limits its practicality [4, 5]. In contrast, approximate MU [17, 22, 31] seeks to remove the residual influence of the forget set without full retraining. This approach, which also includes ours, can be categorized based on various characteristics, as summarized in Table 1 and discussed below.

*Goal.* The goal of approximate MU can be divided into two directions: undoing [25, 43] and misclassification [6, 37]. Undoing aims to reproduce the model that would result from retraining from scratch without using the forget set. Misclassification seeks to remove targeted information by enforcing incorrect predictions on designated forget instances. Note that undoing is inherently limited: the retrained model used as a reference still classifies forget instances and varies across runs [11], making it diffi-

Table 1. Comparison of MU methods. SP denotes whether structural preservation is considered, while TG and DA indicate target granularity and data availability, respectively.

Method	Goal	TG	DA	SP
Selective Forget [12]	undo	instance	✓	n/a
UNSIR [32]	undo	class	✓	n/a
Boundary [7]	undo	class	✗	n/a
L2UL [6]	misclassify	instance	✗	n/a
LoTUS [31]	undo	instance	✓	n/a
Ours	misclassify	instance	✗	✓

cult both to guarantee information removal and to serve as a reliable evaluation baseline. A key observation motivating our work concerns whether removing undesirable knowledge inevitably disrupts the structure of retained knowledge. We find that preserving the structure of representations among retained instances is essential for achieving a strong deletion–retention balance (see Figure 2). Thus, we remove targeted information while keeping the structure of retained knowledge, called structure-preserving unlearning, and adopt misclassification as the deletion criterion.

*Target Granularity.* Approximate MU can also be distinguished by its target granularity: class-level [32, 38, 39] or instance-level [6, 12, 18, 25, 31]. Class-level unlearning removes all samples of a class while maintaining performance on the remaining classes. Instance-level unlearning removes specific samples, which can be distributed across multiple classes. Deletion requests usually target individuals rather than entire classes, making instance-level unlearning both realistic and challenging [6, 34]. For these reasons, we focus on instance-level unlearning.

*Data Availability.* Some studies assume access to the retention dataset during unlearning [12, 25, 31], whereas others prohibit such access [6, 17, 22]. The latter is more practical, since original training data are often unavailable at the time of deletion due to policy restrictions, storage regulations, or capacity limitations [29, 44]. We assume that the retention set is inaccessible, and only the pre-trained model and the requested forget data are available.

### 3. Structure-Faithful Instance Unlearning

We define the training set as  $D_{\text{train}} = \{(x^i, y^i)\}_{i=1}^N$ , where each input  $x^i \in \mathcal{X}$  has its corresponding label  $y^i \in \mathcal{Y}$ , and  $N$  is the total number of instances. We denote the forget set by  $D_f = \{(x_f^i, y_f^i)\}_{i=1}^{N_f}$ , which is a subset of  $D_{\text{train}}$ . The retention set is defined as  $D_r := D_{\text{train}} \setminus D_f$ . We denote the pretrained model as  $f_{\theta}^{\text{ori}}(\cdot) = h_{\phi}^{\text{ori}}(g_{\psi}^{\text{ori}}(\cdot))$ , where  $g_{\psi}^{\text{ori}}$  and  $h_{\phi}^{\text{ori}}$  are the feature extractor and classifier, respectively. The model is parameterized by  $\theta = \{\psi, \phi\}$ , where  $\psi$  and  $\phi$  represent the sets of parameters of the feature extractor and classifier, respectively. We aim to obtain an unlearned model  $f_{\theta}^{\text{unl}}$  by updating the pretrained model  $f_{\theta}^{\text{ori}}$ .  $f_{\theta}^{\text{unl}}$  should eliminate the influence of  $D_f$  while pre-

serving performance on  $D_r$ . We adopt a realistic regime: during unlearning, we have access only to the pretrained model  $f_{\theta}^{\text{ori}}$  and the forget set  $D_f$ .

As shown in Figure 2 (a), existing MU approaches [6, 12] suffer from progressive structural collapse. Such collapse indicates that performance degradation arises from a compromised representation structure, underscoring the need for structure-preserving MU. To this end, we propose a structure-faithful MU framework. We introduce semantic anchors, which serve as reference points linking retention instances and preserving the overall structure. To preserve this structure, we propose two constraints: a structure-aware alignment that upholds the distribution of semantic relationships between retained instances and the anchors, thereby maintaining their relative positions, and a regularization that prevents model updates from disrupting them, as detailed in Section 3.3. The overall framework is shown in Figure 3.

#### 3.1. Anchor Generation

We aim to remove designated instances while preserving the common semantic relational structure of the retained ones using semantic anchors. Motivated by finding that interpretable semantic descriptions provide stable reference cues across instances [26], we define an attribute set for each class  $c$ ,  $E_c = \{e_{c,j}\}_{j=1}^{M_c}$ , by prompting a large language model with a class-aware query [26]. We concatenate the attributes into a single description  $d_c = \text{concat}(e_{c,1}, \dots, e_{c,M_c})$ . A frozen semantic encoder  $T(\cdot)$  maps  $d_c$  to the class anchor  $a_c$ . We assemble the ( $\ell_2$  normalized) anchors into  $A = \{a_c\}_{c \in \mathcal{Y}} \in \mathbb{R}^{b \times d}$ , where  $b$  is the number of classes and  $d$  is the embedding dimension. During unlearning,  $A$  remains fixed as a stable and data-frugal reference independent of  $D_r$ , as illustrated in Figure 3.

#### 3.2. Definition of Structure

We define *structure* as the semantic relations linking anchors and embeddings, inspired by prior studies on relational organization in knowledge [1, 23]. Specifically, we compute affinities that represent the degree of semantic relatedness between anchors  $A$  and the embeddings of retention instances produced by the feature extractor. Since measuring these affinities requires  $D_r$ , when unavailable, we approximate its embeddings by generating adversarial variants<sup>1</sup> from the forget inputs that are pushed toward target-class features [16]. This produces a surrogate probe set  $D_s = \{(x_s^i, y_s^i)\}_{i=1}^{N_s}$ , taking the role of  $D_r$  [6].

Let  $V^{\text{ori}} = \text{norm}(g_{\psi}^{\text{ori}}(x_s)) \in \mathbb{R}^{N_s \times d}$  denote the embeddings from the original pretrained model  $f_{\theta}^{\text{ori}}$ , where  $\text{norm}(\cdot)$  is a normalization operator. We define the original structure  $S^{\text{ori}}$ , which represents the state before unlearning, as  $V^{\text{ori}} \cdot A^{\text{T}} \in \mathbb{R}^{N_s \times b}$ , encoding how the pretrained model

<sup>1</sup>Generating  $N_{\text{adv}}$  adversarial examples for each  $x_f^i$ ,  $N_s = N_f \times N_{\text{adv}}$ .

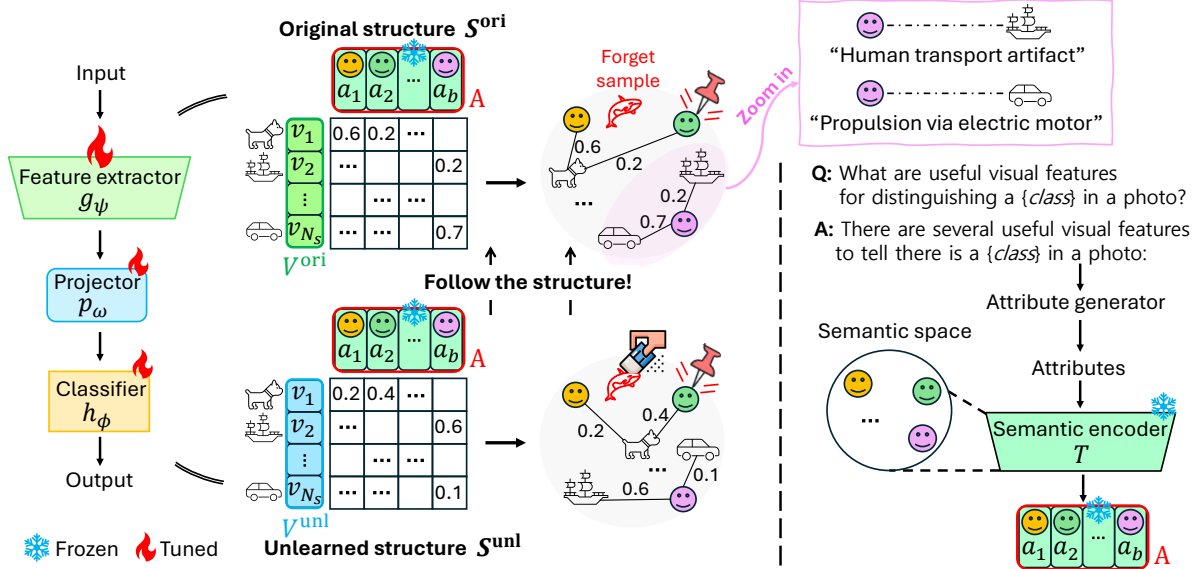


Figure 3. An illustration of the proposed structure-faithful unlearning framework. The left side illustrates the unlearning process, where we aim to preserve the original structure,  $S^{\text{ori}}$ , defined by the affinities between visual embeddings  $V$  and semantic anchors  $A$ , by ensuring that these affinities remain consistent in the unlearned structure  $S^{\text{unl}}$ . The right side shows the procedure for collecting class-wise anchors: a large language model generates attribute descriptions, which are embedded into anchor vectors via a frozen semantic encoder  $T$ .

organizes knowledge relative to  $A$ . Each element of  $S^{\text{ori}}$  represents the relevance between an instance and an anchor. Let  $v_i^{\text{ori}}$  and  $a_j$  denote the  $i$ -th and  $j$ -th rows of  $V^{\text{ori}}$  and  $A$ , respectively, then  $S_{i,j}^{\text{ori}} = \langle v_i^{\text{ori}}, a_j \rangle$ . Note that  $S^{\text{ori}}$  remains unchanged throughout unlearning.

**Unlearned Structure.** During unlearning, the original structure may shift into an updated relational state due to representation drift. We quantify this change by measuring the affinities between  $A$  and the embeddings obtained during unlearning. The resulting relational configuration, which we term the unlearned structure, captures how the underlying knowledge organization evolves through unlearning. Let  $V^{\text{unl}}$  denote the embeddings obtained during unlearning, defined as  $\text{norm}(p_\omega(g_\psi^{\text{unl}}(x_s))) \in \mathbb{R}^{N_s \times d}$ , where  $p_\omega : \mathbb{R}^d \rightarrow \mathbb{R}^d$  is a learnable projector parameterized by  $\omega$  that bridges the feature extractor  $g_\psi^{\text{unl}}(\cdot)$  and the semantic encoder  $T(\cdot)$  to calibrate embeddings for structural preservation. This projection embeds unlearned features in a semantically coherent space, yielding semantically grounded representations [40]. We define the unlearned structure as

$$S^{\text{unl}} = V^{\text{unl}} \cdot A^\top \in \mathbb{R}^{N_s \times b}. \quad (1)$$

While  $S^{\text{ori}}$  captures the relational pattern learned by the pre-trained model with respect to the anchors,  $S^{\text{unl}}$  quantifies the updated affinities between embeddings and semantic anchors, revealing the structural changes relative to  $S^{\text{ori}}$ .

### 3.3. Structure Preservation

To preserve the structure of retained knowledge during unlearning, we ensure that the semantic relations between

the retained instances and the anchors remain consistent with those observed before unlearning. To achieve this, we propose structure-aware alignment. It enforces consistency between the original and unlearned structures by minimizing their distributional divergence:

$$\mathcal{L}_{\text{align}} = -\frac{1}{b} \sum_{i=1}^b \cos(S_i^{\text{ori}}, S_i^{\text{unl}}). \quad (2)$$

By maximizing the average cosine similarity between  $S^{\text{ori}}$  and  $S^{\text{unl}}$ , our alignment preserves relative patterns across anchors and instances. Empirically, cosine-based alignment yields more stable structural preservation than other measures, such as KL-divergence [21] or Wasserstein distance [2], as discussed in Section 4.3.

While structure-aware alignment provides distributional alignment, it does not account for how model updates affect structural preservation. Consequently, unregulated model updates can alter structure-critical parameters and weaken semantic consistency. To complement this, we introduce structure-aware regularization, which restricts model updates to safeguard the semantic information underlying  $S^{\text{ori}}$ :

$$\mathcal{L}_{\text{reg}} = \frac{1}{2} \sum_i I_i \cdot (\psi_i^{\text{unl}} - \psi_i^{\text{ori}})^2, \quad (3)$$

where

$$I_i = \frac{1}{N_s} \sum_{x_s \in D_s} \left| \frac{\partial \mathcal{L}_{\text{align}}(x_s)}{\partial \psi_i^{\text{unl}}} \right|. \quad (4)$$

Table 2. Results of the image classification tasks on CIFAR-10, CIFAR-100, and ImageNet-1K.  $\mathcal{A}_{\text{test}}$  and  $\mathcal{A}_r$  represent the accuracies on the test set and the retention set, respectively, evaluated under each value of  $k$ .  $\mathcal{A}_f$  represents the accuracy on the forget set, and we report it as  $100 - \mathcal{A}_f$ , where a higher value indicates more effective unlearning. The best performance is highlighted in bold.

Method	CIFAR-10				CIFAR-100				ImageNet-1K				
	$k = 16$	$k = 64$	$k = 128$	$k = 256$	$k = 16$	$k = 64$	$k = 128$	$k = 256$	$k = 16$	$k = 64$	$k = 128$	$k = 256$	
$\mathcal{A}_{\text{test}}(\uparrow)$	BEFORE	92.59	92.59	92.59	92.59	77.10	77.10	77.10	77.10	69.79	69.79	69.79	69.79
	ORACLE	90.21	91.01	89.44	38.59	64.41	67.06	66.88	65.31	63.92	59.16	48.18	43.35
	FISHER	17.10	14.97	17.20	15.09	6.25	4.69	2.86	1.18	53.62	48.63	1.44	1.70
	NEGGRAD	15.87	9.28	7.11	6.47	48.07	21.11	10.19	1.71	53.48	43.11	30.74	2.09
	RAWP	63.10	35.47	12.91	9.12	50.99	14.41	2.79	1.12	20.66	14.28	4.47	2.07
	ADV	65.14	62.23	49.47	36.69	63.17	57.43	53.89	46.45	63.44	57.96	49.01	21.27
	L2UL	79.65	67.08	50.82	45.44	63.69	62.83	58.44	48.71	63.92	59.89	53.86	31.19
	STRUCTGUARD	<b>84.38</b>	<b>69.47</b>	<b>58.40</b>	<b>56.32</b>	<b>66.27</b>	<b>65.07</b>	<b>62.33</b>	<b>56.91</b>	<b>65.53</b>	<b>62.57</b>	<b>57.59</b>	<b>41.15</b>
$\mathcal{A}_r(\uparrow)$	BEFORE	99.60	99.60	99.60	99.60	99.98	99.98	99.98	99.98	79.18	79.18	79.18	79.18
	ORACLE	98.74	99.72	98.97	39.90	99.96	96.17	96.74	96.43	72.58	66.37	52.88	46.43
	FISHER	17.82	15.07	17.43	15.05	5.31	4.27	2.89	1.01	60.30	54.87	1.56	1.84
	NEGGRAD	15.79	9.22	7.11	6.34	66.97	26.20	11.64	1.70	60.07	48.05	34.11	2.13
	RAWP	67.15	37.32	13.28	9.12	72.44	17.39	2.82	1.12	23.48	16.45	5.10	2.38
	ADV	69.70	66.97	53.49	39.33	89.18	81.07	76.28	65.67	72.00	64.88	53.90	23.43
	L2UL	85.75	72.77	54.51	48.95	89.81	89.48	82.86	67.60	72.56	67.28	59.81	35.02
	STRUCTGUARD	<b>91.43</b>	<b>76.32</b>	<b>64.28</b>	<b>61.67</b>	<b>92.98</b>	<b>92.93</b>	<b>89.55</b>	<b>83.30</b>	<b>74.62</b>	<b>70.56</b>	<b>65.01</b>	<b>44.91</b>
$\mathcal{A}_f(\uparrow)$	BEFORE	0.00	0.62	0.47	0.62	0.00	0.00	0.00	0.00	0.00	18.75	17.97	19.54
	ORACLE	100.00	100.00	100.00	99.37	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	FISHER	81.25	84.38	78.91	83.59	93.75	95.31	98.44	98.83	43.75	29.69	98.44	97.66
	NEGGRAD	100.00	100.00	100.00	96.17	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	RAWP	100.00	100.00	95.94	93.83	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	ADV	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	L2UL	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	STRUCTGUARD	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00

$I_i$  quantifies the structural importance of the  $i$ -th parameter in the feature extractor. This suppresses large updates to crucial parameters while allowing moderate changes in less important ones. Our regularization is distinguished from a recent approach [6] in that it constrains parameters crucial for preserving structure, whereas the prior work targets parameters sensitive to the forget samples.

To ensure structural integrity while balancing retention and deletion, we jointly optimize the structure-preserving constraints (i.e.,  $\mathcal{L}_{\text{align}}$  and  $\mathcal{L}_{\text{reg}}$ ) with the corresponding objectives. The retention objective encourages prediction of retained instances via the projector to preserve semantic relationships among representations:  $\mathcal{L}_{\text{ret}} = \text{CE}(h_{\phi}^{\text{unl}}(p_{\omega}(g_{\psi}^{\text{unl}}(x_s))), y_s)$ , where CE is the cross-entropy loss. Conversely, the deletion objective bypasses the projector for effective erasing:  $\mathcal{L}_{\text{del}} = -\text{CE}(h_{\phi}^{\text{unl}}(g_{\psi}^{\text{unl}}(x_f)), y_f)$ . The total loss is the sum of the above losses and is minimized over  $\theta = \{\psi, \omega, \phi\}$ .

## 4. Experiments

### 4.1. Setup

**Scenarios and Datasets.** To evaluate our method termed STRUCTGUARD, we adopted the instance-level unlearning scenario, where misclassification is used as the unlearning criterion [6]. We conducted experiments across three distinct tasks: image classification, face recognition, and image-to-image retrieval. For image classification, we used

CIFAR-10 [20], CIFAR-100 [20], and ImageNet-1K [8]. For face recognition, we evaluated on Lacuna-10 [12], and for image-to-image retrieval, we utilized CIFAR-10 [20]. In each dataset, we constructed the forget set  $D_f$  by randomly selecting  $k$  samples from the training data. Specifically, following [6] for image classification and image-to-image retrieval, we used  $k \in \{16, 64, 128, 256\}$ , while for face recognition, we adopted  $k \in \{3, 7, 9, 12, 64\}$ . The remaining samples were treated as the set  $D_r$  [6].

**Compared Methods.** We evaluated our method against established unlearning baselines. BEFORE denotes the pre-trained model before unlearning, while ORACLE denotes a retrained reference model optimized with positive gradients from  $D_r$  and negative gradients from  $D_f$ . Our comparison also includes diverse unlearning methods, FISHER [12], NEGGRAD [12], RAWP [37], ADV [6], and L2UL [6]. We report classification accuracy on the test set  $D_{\text{test}}$ , retention set  $D_r$ , and forget set  $D_f$ . Further details on comparison methods are available in the supplementary material A.

**Implementation Details.** We followed the training details and adversarial sample generation procedure from [6]. As the feature extractor, we used ResNet-18 [14] for CIFAR-10 and Lacuna-10, and ResNet-50 [14] for CIFAR-100 and ImageNet-1K. The projector,  $p_{\omega}$ , comprised two linear layers with a ReLU activation. For attribute description generation, we adopted GPT-4o to produce class-level attributes [26], with the prompting method and sample descriptions provided in supplementary material E. Each class was as-

sociated with a single anchor, encoded using a ViT-B/32 semantic encoder [27]. An analysis with alternative semantic encoders is included in the supplementary material C. To stabilize the classifier  $h_{\phi}^{\text{unl}}$ , we applied the elastic net penalty [45] as classifier regularization, as discussed in Section 4.3.

## 4.2. Main Results

**Image Classification.** We evaluated STRUCTGUARD on CIFAR-10, CIFAR-100, and ImageNet-1K for image classification, as presented in Table 2. Most methods achieve  $\mathcal{A}_f$  to 100.00% across all  $k$ , except FISHER, which keeps incomplete deletion for every  $k$ , and NEGGRAD and RAWP on CIFAR-10 at  $k = 256$ . For CIFAR-10, ours achieves the highest accuracy on both  $\mathcal{A}_{\text{test}}$  and  $\mathcal{A}_r$  across all  $k$ . Notably, at  $k = 256$ , it surpasses ORACLE by 17.73% on  $\mathcal{A}_{\text{test}}$  and 21.77% on  $\mathcal{A}_r$ , demonstrating robust knowledge preservation without a retention set. FISHER, NEGGRAD, and RAWP deteriorate rapidly as  $k$  increases, since forgetting more samples without a retention set leads to cumulative representation drift. We achieve average gains of 13.76% and 16.05% over ADV on  $\mathcal{A}_{\text{test}}$  and  $\mathcal{A}_r$ , respectively, and 6.39% and 7.93% over L2UL, averaged across different values of  $k$ . These gains show that structure-preserving improves deletion–retention balance and generalization.

For CIFAR-100, our method consistently outperforms all baselines across all  $k$ . As  $k$  increases, FISHER, NEGGRAD, and RAWP exhibit a consistent decline on both  $\mathcal{A}_{\text{test}}$  and  $\mathcal{A}_r$ , and the degradation becomes more pronounced when the model must handle a larger number of classes than CIFAR-10. Compared with the strongest baseline, L2UL, ours achieves average gains of 4.22% on  $\mathcal{A}_{\text{test}}$  and 7.25% on  $\mathcal{A}_r$  across  $k$ . Notably, at  $k = 256$ , our retention accuracy exceeds L2UL by 15.70%, demonstrating the benefit of structure preservation. Furthermore, while L2UL suffers a 22.21% drop on  $\mathcal{A}_r$  when  $k$  increases from 16 to 256, ours degrades by only 9.68%, indicating enhanced stability under larger deletion instances.

For ImageNet-1K, across all  $k$ , our method outperforms all baselines by an average of 21.57% on  $\mathcal{A}_{\text{test}}$  and 25.91% on  $\mathcal{A}_r$ . Notably, compared with L2UL, while its average performance drops by 10.91% and 12.51% for  $\mathcal{A}_{\text{test}}$  and  $\mathcal{A}_r$ , respectively, ours shows smaller declines of 8.12% and 9.90%. These results show that ours remains effective on large-scale datasets, underscoring the need to preserve structure for reliable instance unlearning. We present class-level unlearning results for image classification in the supplementary material B.

**Face Recognition.** To validate the generality of STRUCTGUARD, we conducted instance unlearning on a face recognition task. We used the Lacuna-10 dataset, which is derived from VGG-Faces [5], following the procedure in [12]. As shown in Table 3, except for FISHER at  $k = 9$  and  $k = 64$ , all methods, including STRUCTGUARD, success-

Table 3. Results of the face recognition task on Lacuna-10.

Method	Lacuna-10					
	$k = 3$	$k = 7$	$k = 9$	$k = 12$	$k = 64$	
$\mathcal{A}_{\text{test}}(\uparrow)$	BEFORE	94.34	94.34	94.34	94.34	94.34
	ORACLE	91.36	91.47	92.00	91.89	86.78
	FISHER	23.02	15.24	21.42	19.82	23.24
	NEGGRAD	61.51	46.05	45.09	44.88	10.02
	RAWP	72.94	66.84	59.91	41.36	10.66
	ADV	72.28	59.48	55.62	51.81	9.27
	L2UL	75.37	69.08	64.81	58.42	12.26
	STRUCTGUARD	<b>77.29</b>	<b>70.36</b>	<b>70.14</b>	<b>64.07</b>	<b>27.71</b>
$\mathcal{A}_r(\uparrow)$	BEFORE	100.00	100.00	100.00	100.00	100.00
	ORACLE	99.19	99.54	99.43	99.52	98.44
	FISHER	19.84	13.73	23.50	16.97	22.66
	NEGGRAD	68.14	45.94	43.74	40.86	9.57
	RAWP	81.82	76.60	66.64	48.01	11.82
	ADV	80.38	64.33	58.49	55.68	11.31
	L2UL	83.60	76.67	70.01	64.83	14.66
	STRUCTGUARD	<b>84.82</b>	<b>77.01</b>	<b>77.17</b>	<b>66.98</b>	<b>29.97</b>
$\mathcal{A}_f(\uparrow)$	BEFORE	0.00	0.00	0.00	0.00	0.00
	ORACLE	100.00	100.00	100.00	100.00	100.00
	FISHER	100.00	100.00	66.67	100.00	75.00
	NEGGRAD	100.00	100.00	100.00	100.00	100.00
	RAWP	100.00	100.00	100.00	100.00	100.00
	ADV	100.00	100.00	100.00	100.00	100.00
	L2UL	100.00	100.00	100.00	100.00	100.00
	STRUCTGUARD	100.00	100.00	100.00	100.00	100.00

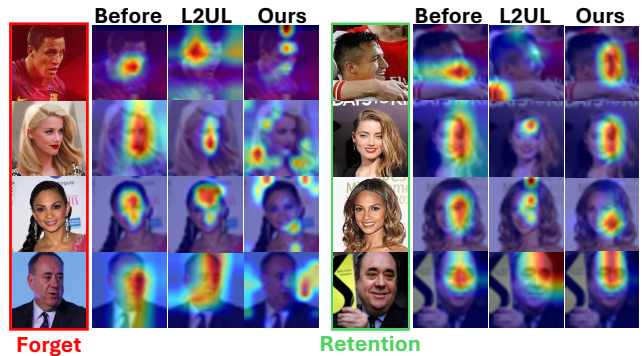


Figure 4. Grad-CAM visualizations on Lacuna-10. Red boxes denote forget instances, and green boxes denote retention instances.

fully remove the designated instances. Across all  $k$ , our method maintains stable recognition performance, achieving gains of 5.92% and 5.23% over the strongest baseline, L2UL, on  $\mathcal{A}_{\text{test}}$  and  $\mathcal{A}_r$ , respectively. These results show that ours achieves a strong deletion–retention balance and superior generalization, as further evidenced by the Grad-CAM [30] results provided in Figure 4.

**Image-to-Image Retrieval.** To extend the evaluation beyond classification, we applied our method to an image-to-image retrieval task as shown in Figure 5. We used the unlearned model under  $k = 256$  from Table 2. We consider two cases: (i) both the query and retrieved samples belong to the retention set (green), and (ii) the query comes from the forget set while the retrieved samples come from the retention set (red). The first evaluates knowledge retention quality, while the second examines whether erased instances are isolated in embedding space.

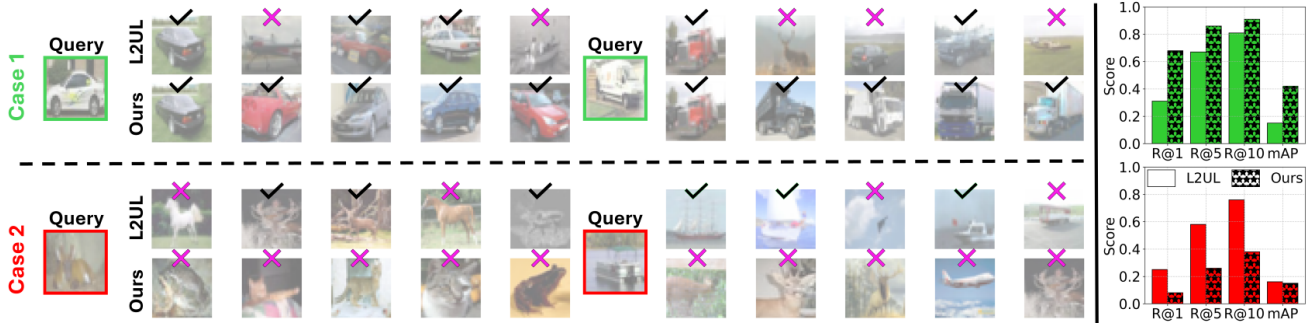


Figure 5. Qualitative and quantitative results of the image-to-image retrieval task on CIFAR-10. The left side shows top-5 retrieval examples given query images. Green-bordered queries correspond to retained samples, while red-bordered queries denote forgotten ones. For each query, retrieved images with correct matches are marked with check marks, and incorrect ones with crosses. The right bar plots report the quantitative retrieval performance in terms of R@1, R@5, R@10, and mAP, comparing our method with L2UL.

In the first case, our method retrieves same-class samples, while L2UL occasionally returns semantically related but incorrect classes (e.g., *airplane* or *ship* for a *car*), suggesting class confusion. In the second case, ours retrieves only different-class samples for all forgotten queries (with the supplementary material D providing a Streisand effect [12] analysis), while L2UL still retrieves same-class samples, implying incomplete deletion. These results show that ours precisely removes forgotten instances while maintaining feature alignment for retained data. Quantitatively, Figure 5 (right) shows that ours achieves higher Recall and mAP for case (i) and lower scores for case (ii), indicating both reliable retention and effective forgetting.

### 4.3. Analyses

**Representation Consistency.** To assess how unlearning alters the retained representations, we examined their consistency with BEFORE, as shown in Figure 6. Representation consistency is quantified by estimating the kernel density of cosine similarities between the embeddings of identical retained samples from BEFORE and post-unlearning models, where higher similarity indicates smaller deviation from BEFORE. Across both datasets, NEGGRAD shows a broad distribution skewed toward low similarity values, showing substantial representation drift. Both ADV and L2UL yield a higher proportion of samples with high similarity scores than NEGGRAD, yet their distributions stay dispersed, with mid-range similarities still appearing on CIFAR-10 and high-similarity values spread widely on CIFAR-100. Such dispersion indicates inconsistency in the retained representations. In contrast, ours shows a remarkable result: a sharp peak near 1.0 (unchanged retained representation), indicating that retained features remain well aligned with BEFORE. These results show that ours effectively removes target information while retaining the consistency of retained representations with BEFORE.

**Anchor Type.** To evaluate the role of semantic anchors in

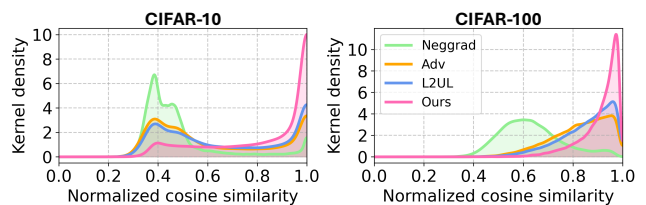


Figure 6. Representation consistency between BEFORE and each unlearning method under  $k = 256$ .

Table 4. Comparison of different anchor designs under  $k = 256$ .

Method	CIFAR-10			ImageNet-1K		
	$\mathcal{A}_{\text{test}}(\uparrow)$	$\mathcal{A}_r(\uparrow)$	$\mathcal{A}_f(\uparrow)$	$\mathcal{A}_{\text{test}}(\uparrow)$	$\mathcal{A}_r(\uparrow)$	$\mathcal{A}_f(\uparrow)$
L2UL	45.44	48.95	100.00	31.19	35.02	100.00
Ours-vis	48.78	54.23	100.00	40.95	42.12	100.00
Ours	<b>56.32</b>	<b>61.67</b>	100.00	<b>41.15</b>	<b>44.91</b>	100.00

preserving structure, we replaced them with visual prototypes obtained by averaging retention embeddings per class, without linguistic guidance. As shown in Table 4, while the visual-anchor variant (Ours-vis) substantially outperforms the non-anchor baseline L2UL on both datasets, semantic anchors (Ours) yield further improvements, increasing  $\mathcal{A}_{\text{test}}$  and  $\mathcal{A}_r$  by 7.84% and 7.44% on CIFAR-10, and by 0.20% and 2.79% on ImageNet-1K compared to Ours-vis. These results indicate that although anchors are crucial for structural preservation, semantic anchors additionally enhance feature alignment by providing semantically grounded guidance.

**Anchor-Guided Structure Preservation.** To investigate whether semantic anchors guide structure preservation, we examined the affinities between randomly selected CIFAR-10 test samples and anchors at  $k=256$ , as shown in Figure 7. For both instances, our method yields affinities closely aligned with BEFORE, showing stable semantic associations after unlearning. In contrast, ADV and L2UL exhibit noticeable shifts in semantic relations and reduced correspondence to the original anchors, showing structural

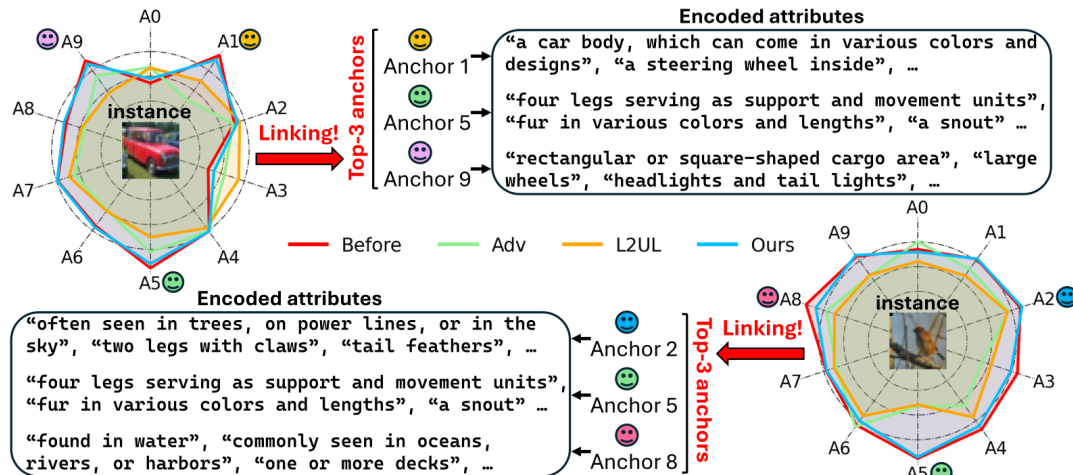


Figure 7. Comparison of how the relational structure in BEFORE changes under each unlearning method. Greater radial extent indicates stronger association with the anchor ( $A_i$  corresponds to the  $i$ -th anchor). We present the top-3 anchors of the highest affinities along with their corresponding attribute descriptions.

Table 5. Ablation study on CIFAR-10 and CIFAR-100.

SA	SR	CR	CIFAR-10			CIFAR-100		
			$\mathcal{A}_{\text{test}}(\uparrow)$	$\mathcal{A}_r(\uparrow)$	$\mathcal{A}_f(\uparrow)$	$\mathcal{A}_{\text{test}}(\uparrow)$	$\mathcal{A}_r(\uparrow)$	$\mathcal{A}_f(\uparrow)$
✓	✓	✓	37.75	42.36	100.00	51.49	80.52	100.00
✓	✓	✓	55.01	60.67	100.00	55.86	82.33	100.00
✓	✓	✓	54.08	59.83	100.00	56.42	82.96	100.00
✓	✓	✓	<b>56.32</b>	<b>61.67</b>	100.00	<b>56.91</b>	<b>83.30</b>	100.00

drift due to the absence of semantic guidance. We observe strong connections to the anchors corresponding to their respective classes (Anchor 1 for the left and Anchor 2 for the right). This reflects that semantic anchors, which encode visual attributes, serve effectively as reference points that support coherent and interpretable structure preservation.

**Ablation Study.** We conducted an ablation study to analyze the contribution of each component in STRUCTGUARD: Structure-aware Alignment (SA), Structure-aware Regularization (SR), and Classifier Regularization (CR), with results in Table 5. Removing SA causes the largest performance drop on both datasets, indicating that alignment is the key factor in keeping structural consistency. On CIFAR-10, the larger performance drop from CR than SR suggests that improvement mainly depends on refining the classifier rather than constraining model updates. On CIFAR-100, SR becomes more influential than CR as the number of classes increases. Overall, SA drives structure preservation, while SR and CR enhance stability by regulating model updates and classifier behavior across different class scales.

**Structural Alignment Loss.** To validate the design of the structure-aware alignment loss, we compared cosine similarity (CS) against other measures, such as Mean Squared Error (MSE), Maximum Mean Discrepancy (MMD), Kullback–Leibler (KL) divergence, and Wasserstein distance (WD), as shown in Table 6. By enforcing directional consistency

Table 6. Comparison of different loss functions for structure-aware alignment under  $k = 256$ .

Loss	CIFAR-10			CIFAR-100		
	$\mathcal{A}_{\text{test}}(\uparrow)$	$\mathcal{A}_r(\uparrow)$	$\mathcal{A}_f(\uparrow)$	$\mathcal{A}_{\text{test}}(\uparrow)$	$\mathcal{A}_r(\uparrow)$	$\mathcal{A}_f(\uparrow)$
MSE	43.80	49.67	100.00	54.50	79.72	100.00
MMD	43.58	47.99	100.00	55.52	80.74	100.00
KL	47.41	52.13	100.00	55.46	80.35	100.00
WD	53.16	58.73	100.00	55.51	80.73	100.00
CS (Ours)	<b>56.32</b>	<b>61.67</b>	100.00	<b>56.91</b>	<b>83.30</b>	100.00

between features and anchors, CS preserves semantic relationships and maintains structural integrity during unlearning. On CIFAR-10, CS improves  $\mathcal{A}_{\text{test}}$  and  $\mathcal{A}_r$  by an average of 9.33% and 9.54% over other losses, and it maintains consistently strong performance on CIFAR-100. These results show that CS provides a simple and effective alignment objective for structure-faithful unlearning.

## 5. Conclusion

We have proposed STRUCTGUARD, a novel method that removes designated information while preserving the integrity of retained knowledge. We introduce anchors constructed from attribute sets generated by a language model and encoded with a semantic encoder. These anchors act as reference points to preserve representation structure, capturing the semantic relationships between anchors and the embeddings of retained instances. To maintain structure, we propose two constraints: structure-aware alignment, which aligns the structure before and after unlearning, and structure-aware regularization, which limits changes to parameters critical to structural integrity. Experiments on diverse tasks show that our method achieves superior deletion–retention balance and generalization over baselines.

**Acknowledgements.** This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) [RS-2021-II211341, Artificial Intelligence Graduate School Program (Chung-Ang University)].

## References

- [1] John R Anderson. A spreading activation theory of memory. *Journal of verbal learning and verbal behavior*, 22(3):261–295, 1983. 3
- [2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017. 4
- [3] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250, 2008. 1
- [4] Lucas Bourtole, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *2021 IEEE symposium on security and privacy (SP)*, pages 141–159. IEEE, 2021. 1, 2
- [5] Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE symposium on security and privacy*, pages 463–480. IEEE, 2015. 1, 2, 6
- [6] Sungmin Cha, Sungjun Cho, Dasol Hwang, Honglak Lee, Taesup Moon, and Moontae Lee. Learning to unlearn: Instance-wise unlearning for pre-trained classifiers. In *Proceedings of the AAAI conference on artificial intelligence*, pages 11186–11194, 2024. 1, 2, 3, 5, 11
- [7] Min Chen, Weizhuo Gao, Gaoyang Liu, Kai Peng, and Chen Wang. Boundary unlearning: Rapid forgetting of deep networks via shifting the decision boundary. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7766–7775, 2023. 3
- [8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5
- [9] 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. *arXiv preprint arXiv:2310.12508*, 2023. 1
- [10] Jack Foster, Stefan Schoepf, and Alexandra Brintrup. Fast machine unlearning without retraining through selective synaptic dampening. In *Proceedings of the AAAI conference on artificial intelligence*, pages 12043–12051, 2024. 1, 2
- [11] Shashwat Goel, Ameya Prabhu, Amartya Sanyal, Ser-Nam Lim, Philip Torr, and Ponnurangam Kumaraguru. Towards adversarial evaluations for inexact machine unlearning. *arXiv preprint arXiv:2201.06640*, 2022. 2
- [12] 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*, pages 9304–9312, 2020. 1, 2, 3, 5, 6, 7, 11, 12
- [13] Laura Graves, Vineel Nagisetty, and Vijay Ganesh. Amnesiac machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11516–11524, 2021. 1, 2
- [14] 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*, pages 770–778, 2016. 5
- [15] Kiseong Hong, Gyeong-hyeon Kim, and Eunwoo Kim. Rainbowprompt: Diversity-enhanced prompt-evolving for continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1130–1140, 2025. 1
- [16] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *Advances in neural information processing systems*, 32, 2019. 3
- [17] Hyoseo Kim, Dongyoon Han, and Junsuk Choe. Negmerge: Consensual weight negation for strong machine unlearning. *arXiv preprint arXiv:2410.05583*, 2024. 1, 2, 3
- [18] Junyaup Kim and Simon S Woo. Efficient two-stage model retraining for machine unlearning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4361–4369, 2022. 3
- [19] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pages 1885–1894. PMLR, 2017. 1
- [20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images.(2009), 2009. 5
- [21] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1): 79–86, 1951. 4
- [22] Tae-Young Lee, Sundong Park, Minwoo Jeon, Hyoseok Hwang, and Gyeong-Moon Park. Esc: Erasing space concept for knowledge deletion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5010–5019, 2025. 1, 2, 3, 11
- [23] Hugo Liu and Push Singh. Conceptnet—a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4): 211–226, 2004. 3
- [24] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)*, 54(6):1–35, 2021. 1
- [25] Ronak Mehta, Sourav Pal, Vikas Singh, and Sathya N Ravi. Deep unlearning via randomized conditionally independent Hessians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10422–10431, 2022. 2, 3
- [26] Sachit Menon and Carl Vondrick. Visual classification via description from large language models. *arXiv preprint arXiv:2210.07183*, 2022. 2, 3, 5, 12

- [27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 6
- [28] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019. 11
- [29] Sara Rosenbaum. Data governance and stewardship: designing data stewardship entities and advancing data access. *Health services research*, 45(5p2):1442–1455, 2010. 3
- [30] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 6
- [31] Christoforos N Spertalis, Theodoros Semertzidis, Efstratios Gavves, and Petros Daras. Lotus: Large-scale machine unlearning with a taste of uncertainty. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 10046–10055, 2025. 1, 2, 3
- [32] Ayush K Tarun, Vikram S Chundawat, Murari Mandal, and Mohan Kankanhalli. Fast yet effective machine unlearning. *IEEE Transactions on Neural Networks and Learning Systems*, 35(9):13046–13055, 2023. 3
- [33] Anvith Thudi, Gabriel Deza, Varun Chandrasekaran, and Nicolas Papernot. Unrolling sgd: Understanding factors influencing machine unlearning. In *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*, pages 303–319. IEEE, 2022. 1
- [34] Eleni Triantafillou, Fabian Pedregosa, Isabelle Guyon, Sergio Escalera, Julio C. S. Jacques Junior, Gintare Karolina Dziugaite, Peter Triantafillou, Vincent Dumoulin, Ioannis Mitliagkas, Lisheng Sun Hosoya, Meghdad Kurmanji, Kairan Zhao, Jun Wan, and Peter Kairouz. Neurips 2023 machine unlearning challenge. <https://unlearning-challenge.github.io/>, 2023. 1, 3
- [35] Paul Voigt and Axel Von dem Bussche. The eu general data protection regulation (gdpr). *A practical guide, 1st ed.*, Cham: Springer International Publishing, 10(3152676):10–5555, 2017. 1
- [36] Junxiao Wang, Song Guo, Xin Xie, and Heng Qi. Federated unlearning via class-discriminative pruning. In *Proceedings of the ACM web conference 2022*, pages 622–632, 2022. 1, 2
- [37] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. *Advances in neural information processing systems*, 33:2958–2969, 2020. 2, 5, 11
- [38] Jingwen Ye, Yifang Fu, Jie Song, Xingyi Yang, Songhua Liu, Xin Jin, Mingli Song, and Xinchao Wang. Learning with recoverable forgetting. In *European Conference on Computer Vision*, pages 87–103. Springer, 2022. 3
- [39] Youngsik Yoon, Jinhwan Nam, Hyojeong Yun, Jaeho Lee, Dongwoo Kim, and Jungseul Ok. Few-shot unlearning by model inversion. *arXiv preprint arXiv:2205.15567*, 2022. 3
- [40] Lu Yu, Haoyu Han, Zhe Tao, Hantao Yao, and Changsheng Xu. Language guided concept bottleneck models for interpretable continual learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14976–14986, 2025. 4
- [41] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023. 11
- [42] Yizhen Zhang, Kuan Han, Robert Worth, and Zhongming Liu. Connecting concepts in the brain by mapping cortical representations of semantic relations. *Nature communications*, 11(1):1877, 2020. 1
- [43] Yu Zhou, Dian Zheng, Qijie Mo, Renjie Lu, Kun-Yu Lin, and Wei-Shi Zheng. Decoupled distillation to erase: A general unlearning method for any class-centric tasks. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 20350–20359, 2025. 2
- [44] Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. *Advances in neural information processing systems*, 32, 2019. 3
- [45] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(2):301–320, 2005. 6