The Right to be Forgotten in Pruning: Unveil Machine Unlearning on Sparse Models

Yang Xiao
University of Tulsa

Gen Li
Clemson University

Jie Ji
Clemson University

Ruimeng Ye
University of Tulsa

Xiaolong ma
The University of Arizona

Bo Hui*
University of Tulsa

Abstract

Machine unlearning aims to efficiently eliminate the memory about deleted data from trained models and address the right to be forgotten. Despite the success of existing unlearning algorithms, unlearning in sparse models has not yet been well studied. In this paper, we empirically find that the deleted data has an impact on the pruned topology in a sparse model. Motivated by the observation and the right to be forgotten, we define a new terminology "un-pruning" to eliminate the impact of deleted data on model pruning. Then we propose an un-pruning algorithm to approximate the pruned topology driven by retained data. We remark that any existing unlearning algorithm can be integrated with the proposed un-pruning workflow and the error of un-pruning is upper-bounded in theory. Also, our un-pruning algorithm can be applied to both structured sparse models and unstructured sparse models. In the experiment, we further find that Membership Inference Attack (MIA) accuracy is unreliable for assessing whether a model has forgotten deleted data, as a small change in the amount of deleted data can produce arbitrary MIA results. Accordingly, we devise new performance metrics for sparse models to evaluate the success of un-pruning. Lastly, we conduct extensive experiments to verify the efficacy of un-pruning with various pruning methods and unlearning algorithms. Our code is released at `https://github.com/NKUShaw/SparseModels`.

1 Introduction

The *Right to be Forgotten* is an emerging principle in artificial intelligence (AI) outlined by regulations such as the General Data Protect Regulation (GDPR), the California Consumer Privacy Act (CCPA), and Canada's proposed Consumer Privacy Protection Act (CPPA) [1–3]. A straightforward "forgetting" strategy is to retrain new models from scratch on the remaining data as if the deleted data has never been seen by the model. However, the retraining method is impractical due to the expensive cost of frequent deletion requests over complex models. To resolve this challenge, a plethora of machine unlearning methods have been developed to efficiently update the model without retraining [4–21].

Despite the success of existing unlearning algorithms, unlearning on sparse models has not been extensively studied yet. We remark that unlearning on sparse models is nontrivial. In Figure 1(a), we visualize the difference between a sparse model based on the original data and a retrained model based on the remaining data (5% data deletion). Specifically, we leverage the Lottery Ticket Hypothesis

*Corresponding Author: bo-hui@utulsa.edu

Workshop on Regulatable ML at the 39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Table 1: Terminologies.

Terminology	Data used	Input model	Description
Retraining	Retained data	Dense Model	Train the model from scratch
Unlearning	Retained and/or deleted data	Dense Model	Generate an unlearned model as though it has never seen the deleted data
Pruning	Full dataset	Dense Model	Prune a dense model to a sparse model with a sparse topology
Retraining + repruning	Retained dataset	Dense Model	Train and prune the model from scratch
Un-pruning	Retained and/or deleted data	Sparse Model	Update the topology of the sparse model to align with retraining + repruning

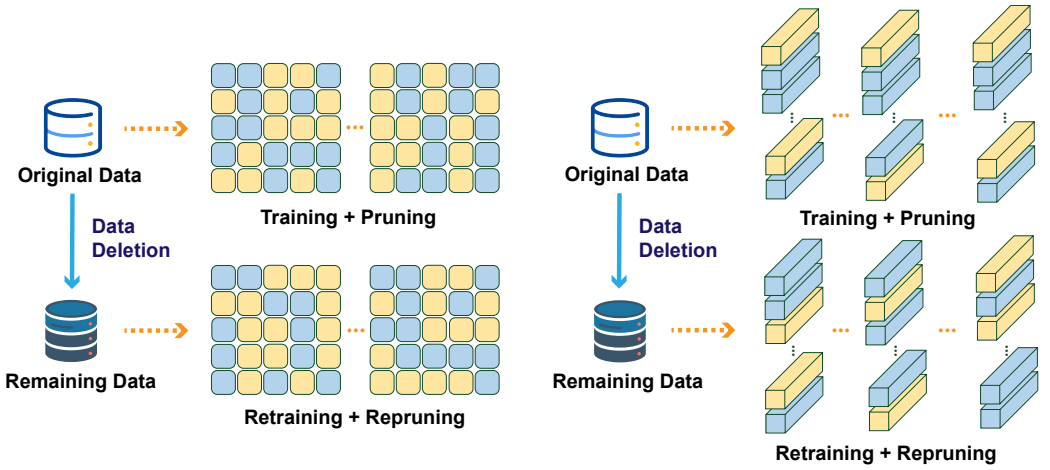

Figure 1: (a) Pruning with the original data vs. repruning with the remaining data. The blue blocks indicate pruned parameters. (b) Structured pruning with the original data vs. the remaining data. The blue blocks indicate pruned channels.

(LTH) [22] to prune a model with the original training data and the remaining data respectively. We can observe that most indexes of the pruned parameters are different between two sparse models. There is only around a 58% overlap of pruned indexes across two sparse models (60% sparsity) of a ResNet-18. The observation also holds for structured pruning. As shown in Figure 1(b), there are significant differences regarding pruned channels. The results indicate that the pruned indexes are data-dependent.

Motivated by this observation, we raise a new research problem: **how to eliminate the influence of deleted data on the pruned topology in a sparse model?**More specifically, the problem is how to generate a new sparse model as if the model has never seen the forgotten data. We define "un-pruning" as a new terminology for eliminating the impact of deleted data on pruning. Table 1 lists the difference between un-pruning and other terminologies such as unlearning. We highlight that un-pruning is non-trivial because: (1) while retraining + repruning serves as the gold standard, it is impractical to perform retraining + repruning on a large-scale model for frequent data deletions. (2) existing unlearning algorithms on the remaining parameters in a sparse model cannot change the pruned topology. These algorithms based on approximating activation values may only align the model's accuracy with the retrained model but fail to align the topology with the standard retraining + repruning [23, 24]; (3) users have the right to withdraw the influence of their data on pruning processes. Our empirical results demonstrate that deleted data has a significant impact on the topology of the sparse model. Consequently, the continued use of sparse models derived from deleted data instances can be deemed illegal [25]. (4) pruning has been considered as a process of learning from the training data [26–28]. Correspondingly, un-pruning can be considered as a process of unlearning. Thus, updating the topology of a sparse model provides us with a new way of unlearning. We use [29] to reconstruct training data with the sparse model. Compared with the original sparse model (ResNet-18, 40% sparsity), the un-pruned model is less accurate in reconstructing the training data.

Table 2: Reconstruction with pruned models

Method	Del. Ratio	Extraction-Score(↓)	SSIM(↑)
Original	0	5295.46	0.4438
Un-prune	20%	5786.37	0.3787
Un-prune	40%	5841.08	0.3521

In this paper, we propose an "un-pruning" algorithm that allows us to integrate any existing unlearning methods into our un-pruning process. Specifi-

cally, we activate the pruned parameters to enable topology change in the sparse model. Then we leverage existing unlearning algorithms to approximate the new parameters and new topology by estimating the influence of deleted data on learning and pruning. We grow the topology in an interactive way to allow smooth and steady topology change. Note that the updated model has to be pruned to the original sparsity so that it can be compared with the retaining + repruning strategy. We highlight that our algorithms can be applied to both unstructured and structured topology. We also analyze our un-pruning algorithm in theory. Compared with retaining + repruning, the un-pruning error regarding the mask is upper bounded and characterized by the model sparsity.

To verify the effectiveness of un-pruning, we follow existing unlearning works to report TA (test accuracy), UA (unlearning accuracy), and MIA (Membership Inference Attack). However, we empirically find that MIA is not a reliable metric to measure the quality of unlearning. With a small change in the data proportion of the binary classification model (i.e., training data vs. test data), the MIA value can fall anywhere within the range of 0 to 1 randomly. Research in large language models also found that the results of MIA are almost equivalent to guesswork [30, 14, 31, 32]. In this paper, we devise new performance metrics to evaluate the quality of unlearning. Specifically, we calculate the structural similarity between neural network representations [33–35] to measure the unlearning quality. Since retraining + repruning serves as the golden standard, we calculate the intersection of pruned indexes between a repruned sparse model based on remaining data and an "un-pruned" sparse model at the same sparsity level. Intuitively, a high intersection ratio means a high similarity between two topologies. In the experiment, we verify the effectiveness and efficiency of our algorithms with various unlearning algorithms and various unlearning algorithms. The contribution of this work can be summarized as:

- We conduct empirical studies to investigate the data-dependency of pruning and the vulnerability of MIA.
- We raise a new research problem: how to eliminate the influence of deleted data on the pruned topology of a sparse model?
- We propose an un-pruning algorithm to approximate the pruned topology driven by the retained data without costly retraining and repruning. Our algorithm can integrate any unlearning algorithm into both unstructured topology and structured topology.
- We design new performance metrics and conduct extensive experiments to verify the effectiveness and efficiency of our un-pruning algorithm.

2 Preliminary

Machine Unlearning. Given a model A trained on a complete dataset D, the user can request to remove their data D_f from D. Machine unlearning aims to eliminate the influence of D_f on A and unlearn a new model that forgets what has been learned from D_f. Denote D_r as the remaining data where $D = D_f \cup D_r$. A naive solution is to retrain a new model A_r based on D_r Dr) from scratch. However, this solution is impractical due to the high computational cost for frequent data deletions over complex models. Recently, approximate unlearning algorithms [8–10, 36, 37] have been developed to directly generate a new model A_u from A without retraining such that the unlearned model A_u approximates A_r as though it had never seen D_r: $A_r(\cdot) \approx A_u(\cdot)$.

Pruning. Pruning can be categorized into two types: unstructured and structured pruning [38–44]. Denote Θ as the parameters of a model A. Unstructured pruning aims to associate a binary mask M where $|M| = |\Theta|$ and a zero value in M indicates that the corresponding parameter is masked out by $\Theta \odot M$. For example, the Lottery Ticket Hypothesis (LTH) [22] masks out the parameters with the smallest magnitude. Instead of pruning parameters distributed irregularly on the model framework, structured pruning removes entire filters, channels, or even layers. Given a specific prune ratio and a neural network with $A = \{s_1, s_2, \cdots, s_L\}$, where s_i can be the set of channels, filters, neurons, or dense layers, the target is to search for $A' = \{s'_1, s'_2, \cdots, s'_L\}$ to minimize performance degeneration and maximize speed improvement under the given prune ratio, where $s'_i \subseteq s'_i$ and $i \in \{1, 2, \cdots, L\}$.

Problem setup. How to eliminate the influence of deleted data on pruning is an open problem. As evidenced by our empirical study in Figure 1, the pruned structure is data-dependent. The user has the

right to withdraw the influence of their data on pruning as the continued use of sparsed models pruned based on deleted data instances can be deemed illegal [25]. Let A be the sparse model parametrized by $\Theta \odot M$ where Θ is the set of weights and M is the mask. Suppose the sparse A is trained and pruned based on the original dataset D. Given a sub-dataset $D_f \subseteq D$ removed from D, the problem is to generate a new sparse model A_u parametrized by $\Theta_u \odot M$ where Θ_u is the new parameters and M_u is the new mask. Considering that the retraining+repruning strategy is impractical for frequent data deletions and expensive pruning strategies such as iterative pruning, our target is to directly generate Θ_u and M_u without retraining and repruning such that M_u is distinguishable from M and $\Theta_u \odot M_u$ is indistinguishable from the result of retraining+repruning.

3 Un-pruning

The first challenge of this research problem is that the pruned parameters are frozen during the training process. Therefore, traditional unlearning algorithms that perform parameter updates can not change the value of frozen parameters. A prior work [45] has also studied unlearning on sparse models. Unfortunately, this work does not update the mask during the unlearning process (i.e., the right to be forgotten in pruning has not been addressed). In this paper, we propose an "un-pruning" algorithm to directly generate the new parameters and the new mask without retraining and repruning. Without loss of generality, Algorithm 1 demonstrates our un-pruning process.

Algorithm 1 Un-pruning
Input: Model parameters Θ, Mask M, original dataset D, deleted dataset D_f, original sparsity $s_\Theta\%$, un-pruning ratio $p_\Theta\%$, unlearning algorithm U, , un-pruning iterations T
Output: New parameters Θ_u, new mask M_u

1: **for** each iteration t= in $0, 1, 2, \cdots, T-1$ **do**
2: Randomly re-initialize pruned parameters as non-zero values: $\Theta \leftarrow \Theta + (1-M)\Theta_0$
3: Perform unlearning (any unlearning algorithm) on the dense full model $\Theta \leftarrow U(\Theta, D_f, D)$
4: Update mask M: set $p_\Theta\%$ mask indexes whose corresponding parameters have the highest magnitude values as 1;
5: Update parameters: $\Theta \leftarrow \Theta \odot M$
6: **end for**
7: Prune the new model to the original sparsity $s_\Theta\%$ by masking the lowest magnitude values; set the corresponding index in M as 0.
8: $\Theta_u = \Theta$, $M_u = M$

In each iteration, we first unfreeze and re-initialize the pruned parameters to perform unlearning algorithms. We re-initialize these parameters because we find that zero-value parameters will remain unchanged with most unlearning algorithms and the pruned indexes will not change after we perform unlearning. In this paper, we have introduced two initialization strategies: original initialization and random initialization. We have investigated the difference in the experiment. Then we update the sparse model by rolling back $p_\Theta\%$ pruned indexes whose corresponding parameters have the highest magnitude values, i.e., the sparsity will be $(s_\Theta - p_\Theta)\%$ after the first iteration. Note that we will update the mask and parameters after unlearning in each iteration. After T iterations, the sparsity will be $(s_\Theta - T * p_\Theta)\%$. Note that we prune the model to the original sparsity $s_\Theta\%$ with one-shot pruning for a fair comparison with repruning.

In this paper, we have investigated 6 most popular approximate unlearning methods: GradientAscent [46], SCRUB [47], Fisher [48–50], WoodFisher [51], CertifiedUnlearning [52], SFRON [53]. While these unlearning methods varies on the objective function in the unlearning process, all existing unlearning algorithms can be integrated with our method.

We remark that our un-pruning strategy works for these unstructured turning algorithms that do not require extra parameters for pruning. Similarly, in each iteration, we unfreeze the most important channels, filters, or dense layers. Specifically, we use the same standard of pruning structures to choose important structures. For example, Soft Filter Pruning [54] selects filters to be pruned based on the l_2 norm of filters in the training stage. Accordingly, we can select the filters to recover based on the l_2 norm. Note that there are some structured pruning algorithms with extra parameters that are used to learn which part to pruneBiP [55–57]. For those algorithms with extra learnable parameters, we advocate for new algorithms to address the extra parameters since exiting unlearning algorithms only update the model itself.

Evaluation of un-pruning. While the similarity between the unlearned model and the retrained model is usually used to measure the quality of unlearning algorithms [35], it is still an open problem to measure the structural similarity between the "un-pruned" model and the standard "repruned" model. First, we define the intersection ratio and the union ratio of two pruning models:

$$IoM = \frac{\|M_u \odot M_r\|_1}{N} \tag{1}$$

$$UoM = \frac{\|M_u + M_r - M_u \odot M_r\|_1}{\mathrm{N}}, \tag{2}$$

where $||\cdot||_1$ is the number of 1 in the mask and N is the total number of parameters. We use M_u and M_r to represent the un-pruned mask and re-pruned mask. Based on the intersection and the union, we can further define IoU (intersection of union):

$$IoU = \frac{\|M_u \odot M_r\|_1}{\|M_u + M_r - M_u \odot M_r\|_1} \tag{3}$$

Figure 2 visualizes the intersection ratio between "un-pruning" integrated with different unlearning algorithms. We also show the unlearning accuracy and the results indicate that our un-pruning algorithm can also guarantee the unlearning accuracy.

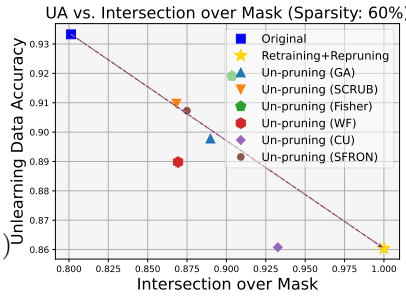

Figure 2: Intersection ratio and unlearning accuracy.

Besides the structural similarity, we have also introduced the KL-divergence [48] to measure the difference:

$$K(\Theta_u, \Theta_r, M_u, M_r) = \mathrm{KL}(P(M_{u,i} \odot \Theta_{u,i}) || P(M_{r,i} \odot \Theta_{r,i})) \tag{4}$$

where $P(\cdot)$ is the distribution probability introduced by the randomness in the learning algorithm. Intuitively, the unlearning process will provide an exact unlearning if $K(\theta_u, \theta_r) = 0$.

Analysis of un-pruning. Given weights Θ and pruning mask M, the pruning target can be simplified as:

$$\min L(\Theta; M; D) = \min \frac{1}{|D|} \sum_{i=1}^{|D|} l(\Theta \odot M; x_i, y_i), \tag{5}$$

where (x_i, y_i) is a data point in D and l is the loss function. Given k components in the model: $\Theta = \Theta_0, \Theta_1, ..., \Theta_k$, we follow Unrolling SGD: [4] to write the update the j-th parameter of i-th component in pruning iteration s in the unlearning process as:

$$\Theta_{i,j}^s = M_{i,j}^{s-1}(\Theta_{i,j}^{s-1} - \eta \nabla_\Theta l(\Theta \odot M; D_f), \tag{6}$$

where η represents the unlearning rate. To establish a connection between un-pruning and un-learning, we rewrite the pruning strategy as:

$$M = g(\Theta) = Sigmoid(\frac{\log P(\Theta) + \epsilon}{\tau}), \epsilon \sim Gumbel(0, 1) \tag{7}$$

where probability p represents the sampling probability of the parameters over the random learning algorithm.

Start with $\tilde{\Theta}^0 = \Theta^S$ after S iterations of training and pruning, we roll back the parameters by:

$$\begin{aligned} \tilde{\Theta}^t &= g(\tilde{\Theta}^{t-1})(\tilde{\Theta}^{t-1} + \eta \frac{\partial L((g(\tilde{\Theta}^{t-1})\tilde{\Theta}^{t-1}; D_f)}{\tilde{\Theta}^{t-1}}) \\ &= g(\tilde{\Theta}^{t-1})(\tilde{\Theta}^{t-1} + \eta \frac{\partial \frac{1}{|D|} \sum_{i=1}^{|D|} l(\tilde{\Theta} \odot M; x_i, y_i)}{\tilde{\Theta}^{t-1}}) \end{aligned} \tag{8}$$

Since the new mask M_t is data-dependent, the final mask of the given parameter is:

$$M = g(\tilde{\Theta}^t) g(\tilde{\Theta}^{t-1}) \dots g(\tilde{\Theta}^1) g(\tilde{\Theta}^0) = \prod_{i=0}^{t} g(\tilde{\Theta}^i) \tag{9}$$

To estimate the un-pruning error regarding M, we approximate the unlearning error in [45] in the un-pruning process. Specifically, we first approximate the loss function using the second-order Taylor expansion around the initial mask $M^0 = M$:

$$L(M) = L(\mathbf{1}) + \frac{1}{2}(\mathbf{1} - M)^T \mathbb{E}[\frac{\partial^2 L(\mathbf{1})}{\partial M^2}](\mathbf{1} - M) \tag{10}$$

Denote the Hessian matrix of the loss function with:

$$H = \nabla^2 L(\tilde{\Theta}; D_f) \tag{11}$$

At this moment, the changes in parameters are:

$$\Delta(\tilde{\Theta}) := \tilde{\Theta} - \tilde{\Theta}^0 \approx H^{-1} \nabla L(\tilde{\Theta}_o; D_f) \tag{12}$$

Then let the σ be the largest singular value:

$$\lambda(M) := max\{\lambda_j(H), 1\} \tag{13}$$

According to [4], the overall parameters changes can be written as:

$$\tilde{\Theta}^t \approx \tilde{\Theta}_0 + \eta \sum_{i=0}^{t-1} \frac{\partial L}{\partial \tilde{\Theta}} + \sum_{i=1}^{t-1} k(i) \tag{14}$$

where $k(i)$ is defined recursively as:

$$k(i) = -\eta \frac{\partial^2 L}{\partial^2 \tilde{\Theta}}(k(i-1)),\ k(0) = 0 \tag{15}$$

By introducing the mask, the parameter change can be rewritten as:

$$\tilde{\Theta}^t \approx \tilde{\Theta}_0 + \eta M^{t-1} \sum_{i=1}^{t-1} \nabla L(M^{t-1} \odot \tilde{\Theta}; D_f) + M^0 \sum_{i=1}^{t-1} k(i) \tag{16}$$

By comparing Eq. (6) and Eq. (16), we can see that the un-pruning error between $\tilde{\Theta}^t$ and the original Θ_0 is:

$$e(g(\Theta)) = \left\| M^0 \sum_{i=1}^{t-1} k(i) \right\|_2 \approx \eta^2 \left\| \text{diag}(^0) \sum_{i=1}^{t-1} \nabla^2 l(\Theta_0, D_f) \sum_{j=0}^{i-1} M^0 \odot \nabla l(\Theta_0, D_f) \right\|_2 \tag{17}$$

According to the Triangle Inequality:

$$e(g(\Theta)) \leq \frac{\eta^2}{2}(t-1) \| M^0 \odot (\Theta^t - \Theta_0) \|_2 \lambda(M^0) \tag{18}$$

In summary, the upper bound of the un-pruning error regarding the mask can be characterized by the largest singular value $\lambda(M)$ and the model sparsity:

$$e(g(\Theta)) = O(\eta^2 t \| M \odot \Theta \|_2 \lambda(M)) \tag{19}$$

The theory can be empirically justified by Table 3. As the model sparsity increases, the KL divergence between "un-pruning" and "repruning" also increases. It verifies that the un-pruning error bound is related to the sparsity.

Table 3: KL divergence at different sparsity levels.

Method	KL		
	40%	60%	95%
Original	19.06	19.68	21.79
Retrain	0.00	0.00	0.00
Fisher	16.47	16.85	15.09
WoodFisher	19.06	19.68	21.79
GradientAscent	19.05	19.66	21.79
CertifiedUnlearning	16.49	19.32	19.08
SCRUB	19.04	19.57	21.54
SFRon	19.03	19.60	21.59

To further investigate the KL divergence after un-pruning, we measure the difference between the repruned model and retrained model, we follow [58] to introduce the inequality:

$$\begin{aligned} \mathbb{E}_\Theta L(\Theta; M; D) &\leq \mathbb{E}_\Theta L(\Theta; M; D_r) \\ &+ \sqrt{\frac{\text{KL}(Q||P) + \log \frac{|D_f|}{\delta}}{2(|D_r| - 1)}}, \forall \delta > 0, \end{aligned} \tag{20}$$

where Q and P are prior and posterior distribution on Θ.

$$\mathrm{KL}(Q||P) \leq \frac{1}{2}\sum_i (1 + k_i log(\frac{1+\alpha_1^2||\Theta_i^t - \Theta_i^0||}{k_i \sigma_{Q,i}^2}), \tag{21}$$

where k_i is the number of parameters in module i of Θ^t and σ is the standard deviation of Gaussian noise. Then we have the following inequality [59]:

$$\begin{aligned} &\mathbb{E}_\Theta L(\Theta; M; D) - \mathbb{E}_\Theta L(\Theta; M; D_r) \\ &\leq \sqrt{\frac{\frac{1}{4}\sum_{i=1} k_i \log(1 + \frac{\mu_{i,\epsilon}(\Theta)}{n_i}) + \log(\frac{|D_r|}{\delta}) + \tilde{O}(1)}{|D_r| - 1}}, \end{aligned} \tag{22}$$

where

$$\begin{aligned} \mu_{i,\epsilon}(\Theta) = \min_{0 \leq \alpha_i, \sigma_i \leq 1} \{ &\frac{\alpha_i^2 \|\Theta^t - \Theta_0^i\|^2}{\sigma_i} : \\ &\mathbb{E}_{u \sim N(0,\sigma_i^2)} L(\theta_i; D_r) \leq \epsilon \} \end{aligned} \tag{23}$$

and α_i is the re-initialized weights. In inclusion, the difference between un-pruning and repruning is bounded.

4 Experiment

In this paper, we have followed [45] to set up the experiment. Specifically, we have introduced 3 models to be pruned: **ResNet-18** and **AlexNet**, and **ViT**. The models are trained and pruned on 3 datasets: **CIFAR-10**, **FashionMNIST**, and **ImageNet**. For all the datasets and model architectures, we randomly deleted 10% data. We use (LTH) [22] and Soft Filter Pruning [54] as the default unstructured pruning and structured pruning method respectively. The sparse model can reconstruct training data, suggesting that their reconstruction ability is linked to data memorization [28]. Inspired by this, we evaluate whether a sparse ResNet-18 model (with 40% sparsity) retains training data by testing its reconstruction ability. Besides the intersection between the repruned mask and the "un-pruned" mask, we have also introduced IoM(Intersection over Mask), TA (test accuracy), UA (unlearning accuracy), and MIA. We highlight that MIA is fragile as an evaluation of unlearning. We will demonstrate our findings in Section 5.

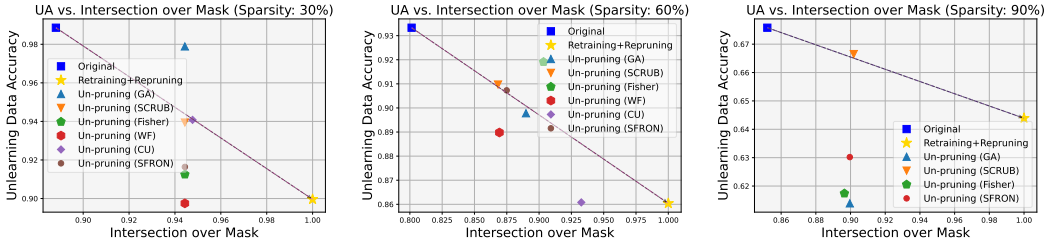

Figure 3: Structured un-pruning (IoM).

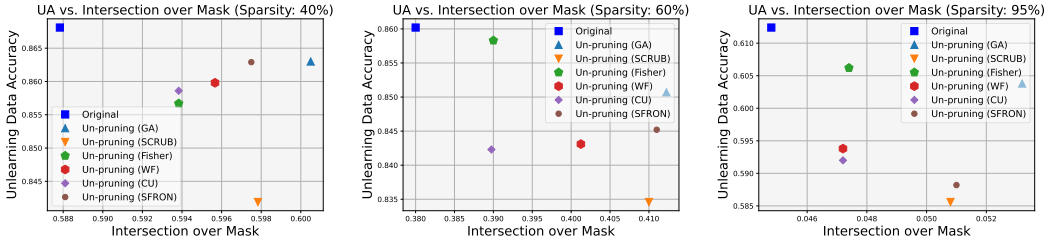

Figure 4: Unstructured un-pruning (IoM).

Results and analysis. For Table 2, Extraction-Score refers to the distance between generated images and real images. SSIM refers to the structural similarity between generated images and real images.

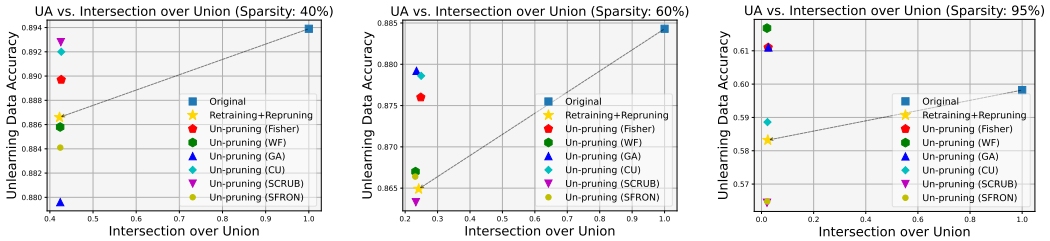

Figure 5: Unstructured un-pruning (IoU).

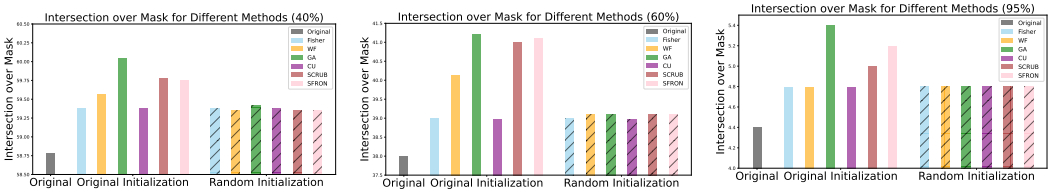

Figure 6: Original initialization vs. random initialization.

The experiments shows that the new (un-pruned) sparse model is less accurate in reconstructing the training data compared with the original sparse model. We integrated various unlearning algorithms including Fisher [48], WoodFisher (WF) [51], Gradient Ascent (GA) [46], Certified Unlearning (CU) [52], SCRUB [47], and SFRON [53] into our un-pruning method. Figure 3, 4 and 5 show the experiment result for 3 sparsity levels (i.e., 40%, 60%, and 95%) with structured pruning and structured pruning, respectively. The x-axis indicates IOU and the y-axis measures UA. Figure 7 visualizes the overall performance including sparsity, TA, and UA. Compared with the original model, the un-pruned model is closer to the retrained + repruned model in terms of both IoM and UA. At the same time, the un-pruned model has a very different topology and UA from the original model. Also, we find that our un-pruning has a better performance on structured pruning. This is because structured topology is easier to approximate than unstructured topology where there is a marginal difference in the parameter's magnitude within a component (e.g., channel). The results verify that our un-pruning algorithm can generate a new sparse model that is similar to retraining + repruning.

Original initialization vs. random initialization. We analyze experiments with two initialization methods. For the original initialization, we save the original initialization weights and restore the initialization weights in the un-pruning process. For the random initialization, we assign the random values to pruned parameters. As shown in Figure 6, the original initialization outperforms the random initialization. This is because the original initialization provides a more stable parameter distribution, thereby achieving better performance in the IoM metric. The result aligns with the viewpoint of [60] that for a given initialized network, there exists a supermask that performs best on a specific dataset. The original initialization helps us approach the supertask for the remaining dataset, thereby facilitating the forgetting of the unlearning data.

Table 4: Running time.

Method	Running Time (s)		
	40%	60%	95%
Retraining+Repruning	6749	12217	39235
Fisher	189	182	184
WF	50	48	48
GA	1793	1778	1779
CU	878	908	846
SCRUB	133	132	134
SFRon	91	89	88

Running time analysis In this part, we present and analyze the running time of experiments in Table 4. For an iterative pruning method, the running time of retraining increases as the sparsity increases. Compared with retraining + repruning, our un-pruning method can significantly reduce the computational complexity. At the same time, there is a difference in terms of the running time when we integrate different unlearning algorithms.

5 Vulnerability of MIA in machine unlearning

We initially planned to use MIA as an evaluation metric to calculate our baseline results. However, during the experiments, we found that MIA does not effectively verify the model's unlearning performance. We recorded five MIA evaluation indicators: MIA's Correctness, Confidence, Entropy,

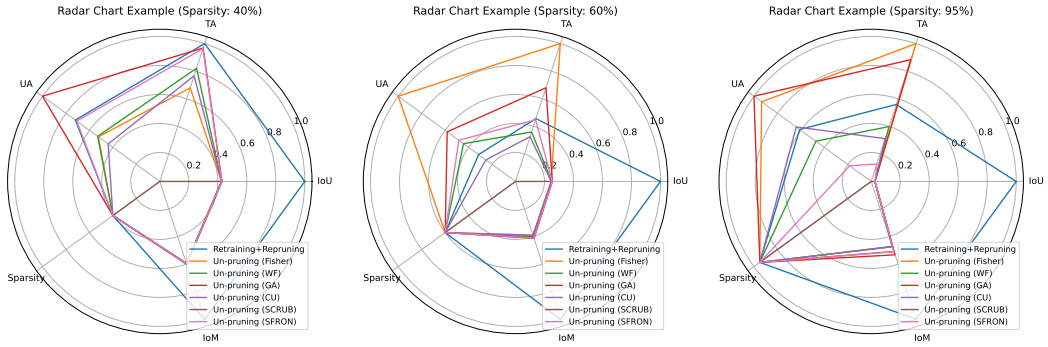

Figure 7: Radar charts of 5 performance metrics.

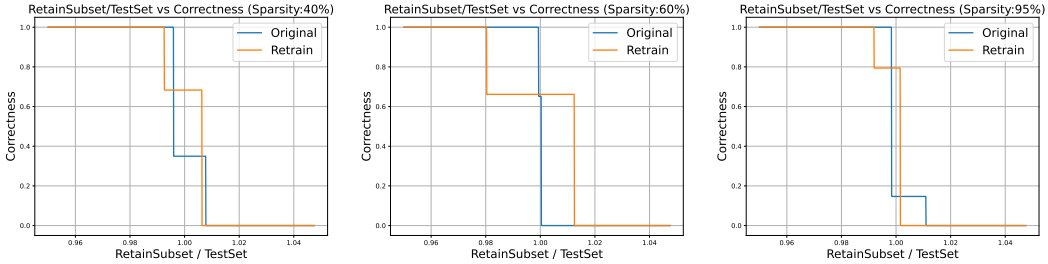

Figure 8: MIA correctness.

m-Entropy, and Probability. However, during the evaluation process of correctness, we found that MIA is close to random guessing, which means it is difficult to determine whether the data belongs to the training set. Then, by fine-tuning the ratio of the training set to the test set, we found that when the sample size exceeded a certain range, MIA would maintain its original value. However, when the sample size exceeded or fell below a certain threshold, MIA would rise to 1.0 or decrease to 0 (See in Figure 8). For the other four indicators, by fine-tuning the ratios, we can obtain values in any range from 0 to 1 (See in Figure 9 and appendix). At the same time, it should be noted that the ratio is always maintained at around 1.0, which means that only a few samples need to be reduced or increased to manipulate MIA. Therefore, using MIA's evaluation on unlearning is fragile, and it is even impossible to distinguish whether the retrained model has forgotten the deleted data.

6 Related Work

Pruning. As the size of deep learning models increases, model pruning has attracted attention to reduce the parameter and improve computational efficiency without compromising accuracy [61–66]. Existing pruning technologies can be classified into two types: structured pruning and unstructured pruning. In general, structured pruning [67–76] remove sparse patterns such as layers and channels from the full model. Unstructured pruning [77–83] removes irregularly distributed parameters instead of entire weight structures (e.g., channels). Recent study shows that there is a dependence between pruned and the training data. Specific masks are more sensitive to the training data [60]. [9] proves that model Sparsitization can simplify the machine unlearning. Experiments by [84] conclude that the influence of deleted data on the connections between layers of the model remains unchanged.

Machine unlearning. Machine unlearning aims to remove the impact of deleted on trained models. While the retraining strategy is intuitive and costly, exact unlearning aims to learn a model with the same performance as the retrained model [6, 53, 85–93]. Approximate unlearning methods update the model by approximating the impact the deleted data on the training model [94, 95, 46, 96–98, 90, 99, 100]. Despite the success of these unlearning algorithms on dense models, unlearning on sparse models has not been extensively investigated. To the best the author's knowledge, [45] has also studied unlearning on sparse models. However, this work does not update the mask during the unlearning process (i.e., the right to be forgotten in pruning has not been addressed).

7 Conclusion

In this paper, we empirically find that pruning is data-dependent. To address the right to be forgotten in pruning, we propose an un-pruning algorithm to approximate the pruned topology driven by the retained data without costly retraining and repruning. We prove the difference between un-pruning and repruning is bounded in theory. The experiment verifies the effectiveness of our method.

References

[1] A. J. Biega and M. Finck, "Reviving purpose limitation and data minimisation in data-driven systems," *arXiv preprint arXiv:2101.06203*, 2021.

[2] P. Regulation, "Regulation (eu) 2016/679 of the european parliament and of the council," *Regulation (eu)*, vol. 679, p. 2016, 2016.

[3] C. OAG, "Ccpa regulations: Final regulation text," *Office of the Attorney General, California Department of Justice*, p. 1, 2021.

[4] 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.

[5] S. Schelter, "amnesia–towards machine learning models that can forget user data very fast," in *1st International Workshop on Applied AI for Database Systems and Applications (AIDB19)*, 2019.

[6] L. Bourtoule, V. Chandrasekaran, C. A. Choquette-Choo, H. Jia, A. Travers, B. Zhang, D. Lie, and N. Papernot, "Machine unlearning," in *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2021, pp. 141–159.

[7] A. Mahadevan and M. Mathioudakis, "Certifiable machine unlearning for linear models," *arXiv preprint arXiv:2106.15093*, 2021.

[8] B. Zhang, Y. Dong, T. Wang, and J. Li, "Towards certified unlearning for deep neural networks," *arXiv preprint arXiv:2408.00920*, 2024.

[9] J. Liu, P. Ram, Y. Yao, G. Liu, Y. Liu, P. SHARMA, S. Liu *et al.*, "Model sparsity can simplify machine unlearning," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[10] J. Liu, J. Lou, Z. Qin, and K. Ren, "Certified minimax unlearning with generalization rates and deletion capacity," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[11] A. Golatkar, 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.

[12] H. Xu, T. Zhu, L. Zhang, W. Zhou, and P. S. Yu, "Machine unlearning: A survey," 2023. [Online]. Available: https://arxiv.org/abs/2306.03558

[13] S. Pal, C. Wang, J. Diffenderfer, B. Kailkhura, and S. Liu, "Llm unlearning reveals a stronger-than-expected coreset effect in current benchmarks," *arXiv preprint arXiv:2504.10185*, 2025.

[14] S. Liu, Y. Yao, J. Jia, S. Casper, N. Baracaldo, P. Hase, Y. Yao, C. Y. Liu, X. Xu, H. Li *et al.*, "Rethinking machine unlearning for large language models," *Nature Machine Intelligence*, pp. 1–14, 2025.

[15] C. Fan, J. Jia, Y. Zhang, A. Ramakrishna, M. Hong, and S. Liu, "Towards llm unlearning resilient to relearning attacks: A sharpness-aware minimization perspective and beyond," *arXiv preprint arXiv:2502.05374*, 2025.

[16] C. Sun, R. Wang, Y. Zhang, J. Jia, J. Liu, G. Liu, Y. Yan, and S. Liu, "Forget vectors at play: Universal input perturbations driving machine unlearning in image classification," *arXiv preprint arXiv:2412.16780*, 2024.

[17] H. Zhuang, Y. Zhang, K. Guo, J. Jia, G. Liu, S. Liu, and X. Zhang, "Uoe: Unlearning one expert is enough for mixture-of-experts llms," *arXiv preprint arXiv:2411.18797*, 2024.

[18] C. Fan, J. Liu, L. Lin, J. Jia, R. Zhang, S. Mei, and S. Liu, "Simplicity prevails: Rethinking negative preference optimization for llm unlearning," *arXiv preprint arXiv:2410.07163*, 2024.

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

[20] Y. H. Khalil, L. Brunswic, S. Lamghari, X. Li, M. Beitollahi, and X. Chen, "Not: Federated unlearning via weight negation," *arXiv preprint arXiv:2503.05657*, 2025.

[21] C. N. Spartalis, T. Semertzidis, E. Gavves, and P. Daras, "Lotus: Large-scale machine unlearning with a taste of uncertainty," *arXiv preprint arXiv:2503.18314*, 2025.

[22] J. Frankle and M. Carbin, "The lottery ticket hypothesis: Finding sparse, trainable neural networks," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. [Online]. Available: https://openreview.net/forum?id=rJl-b3RcF7

[23] W. Wang, Z. Tian, and S. Yu, "Machine unlearning: A comprehensive survey," *arXiv preprint arXiv:2405.07406*, 2024.

[24] Y. Qu, X. Yuan, M. Ding, W. Ni, T. Rakotoarivelo, and D. Smith, "Learn to unlearn: A survey on machine unlearning," *arXiv preprint arXiv:2305.07512*, 2023.

[25] E. Fosch-Villaronga, P. Kieseberg, and T. Li, "Humans forget, machines remember: Artificial intelligence and the right to be forgotten," *Comput. Law Secur. Rev.*, vol. 34, no. 2, pp. 304–313, 2018. [Online]. Available: https://doi.org/10.1016/j.clsr.2017.08.007

[26] H. Zhou, J. Lan, R. Liu, and J. Yosinski, "Deconstructing lottery tickets: Zeros, signs, and the supermask," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 3592–3602.

[27] F. Tung and G. Mori, "CLIP-Q: deep network compression learning by in-parallel pruning-quantization," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 7873–7882. [Online]. Available: http://openaccess.thecvf.com/content_cvpr_2018/html/Tung_CLIP-Q_Deep_Network_CVPR_2018_paper.html

[28] N. Haim, G. Vardi, G. Yehudai, O. Shamir, and M. Irani, "Reconstructing training data from trained neural networks," *Advances in Neural Information Processing Systems*, vol. 35, pp. 22 911–22 924, 2022.

[29] ——, "Reconstructing training data from trained neural networks," in *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., 2022. [Online]. Available: http://papers.nips.cc/paper_files/paper/2022/hash/906927370cbeb537781100623cca6fa6-Abstract-Conference.html

[30] M. Duan, A. Suri, N. Mireshghallah, S. Min, W. Shi, L. Zettlemoyer, Y. Tsvetkov, Y. Choi, D. Evans, and H. Hajishirzi, "Do membership inference attacks work on large language models?" *arXiv preprint arXiv:2402.07841*, 2024.

[31] D. Das, J. Zhang, and F. Tramèr, "Blind baselines beat membership inference attacks for foundation models," *arXiv preprint arXiv:2406.16201*, 2024.

[32] J. Zhang, D. Das, G. Kamath, and F. Tramèr, "Membership inference attacks cannot prove that a model was trained on your data," *arXiv preprint arXiv:2409.19798*, 2024.

[33] M. Klabunde, T. Schumacher, M. Strohmaier, and F. Lemmerich, "Similarity of neural network models: A survey of functional and representational measures," *CoRR*, vol. abs/2305.06329, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.06329

[34] S. Kornblith, M. Norouzi, H. Lee, and G. E. Hinton, "Similarity of neural network representations revisited," in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 2019, pp. 3519–3529. [Online]. Available: http://proceedings.mlr.press/v97/kornblith19a.html

[35] H. Xu, T. Zhu, L. Zhang, W. Zhou, and P. S. Yu, "Machine unlearning: A survey," *ACM Comput. Surv.*, vol. 56, no. 1, pp. 9:1–9:36, 2024. [Online]. Available: https://doi.org/10.1145/3603620

[36] Y. Park, S. Yun, J. Kim, J. Kim, G. Jang, Y. Jeong, J. Jo, and G. Lee, "Direct unlearning optimization for robust and safe text-to-image models," *CoRR*, vol. abs/2407.21035, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2407.21035

[37] E. Chien, H. Wang, Z. Chen, and P. Li, "Langevin unlearning: A new perspective of noisy gradient descent for machine unlearning," *CoRR*, vol. abs/2401.10371, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2401.10371

[38] H. Cheng, M. Zhang, and J. Q. Shi, "A survey on deep neural network pruning: Taxonomy, comparison, analysis, and recommendations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 10 558–10 578, 2024. [Online]. Available: https://doi.org/10.1109/TPAMI.2024.3447085

[39] Y. He and L. Xiao, "Structured pruning for deep convolutional neural networks: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 46, no. 5, pp. 2900–2919, 2023.

[40] S. Xu, A. Huang, L. Chen, and B. Zhang, "Convolutional neural network pruning: A survey," in *2020 39th Chinese control conference (CCC)*. IEEE, 2020, pp. 7458–7463.

[41] H. Cheng, M. Zhang, and J. Q. Shi, "A survey on deep neural network pruning: Taxonomy, comparison, analysis, and recommendations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[42] Y. Tang, Y. Wang, J. Guo, Z. Tu, K. Han, H. Hu, and D. Tao, "A survey on transformer compression," *arXiv preprint arXiv:2402.05964*, 2024.

[43] R. Ma and L. Niu, "A survey of sparse-learning methods for deep neural networks," in *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. IEEE, 2018, pp. 647–650.

[44] L. Deng, G. Li, S. Han, L. Shi, and Y. Xie, "Model compression and hardware acceleration for neural networks: A comprehensive survey," *Proceedings of the IEEE*, vol. 108, no. 4, pp. 485–532, 2020.

[45] J. Jia, J. Liu, P. Ram, Y. Yao, G. Liu, Y. Liu, P. Sharma, and S. Liu, "Model sparsity can simplify machine unlearning," in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023.

[46] L. Graves, V. Nagisetty, and V. Ganesh, "Amnesiac machine learning," in *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press, 2021, pp. 11 516–11 524. [Online]. Available: https://doi.org/10.1609/aaai.v35i13.17371

[47] M. Kurmanji, P. Triantafillou, J. Hayes, and E. Triantafillou, "Towards unbounded machine unlearning," in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023.

[48] A. Golatkar, A. Achille, and S. Soatto, "Eternal sunshine of the spotless net: Selective forgetting in deep networks," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 2020, pp. 9301–9309. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2020/html/Golatkar_Eternal_Sunshine_of_the_Spotless_Net_Selective_Forgetting_in_Deep_CVPR_2020_paper.html

[49] J. Foster, S. Schoepf, and A. Brintrup, "Fast machine unlearning without retraining through selective synaptic dampening," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 38, no. 11, 2024, pp. 12 043–12 051.

[50] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[51] S. P. Singh and D. Alistarh, "Woodfisher: Efficient second-order approximation for neural network compression," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[52] B. Zhang, Y. Dong, T. Wang, and J. Li, "Towards certified unlearning for deep neural networks," in *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. [Online]. Available: https://openreview.net/forum?id=1mf1ISuyS3

[53] Z. Huang, X. Cheng, J. Zheng, H. Wang, Z. He, T. Li, and X. Huang, "Unified gradient-based machine unlearning with remain geometry enhancement," *arXiv preprint arXiv:2409.19732*, 2024.

[54] Y. He, G. Kang, X. Dong, Y. Fu, and Y. Yang, "Soft filter pruning for accelerating deep convolutional neural networks," *arXiv preprint arXiv:1808.06866*, 2018.

[55] Y. Zhang, Y. Yao, P. Ram, P. Zhao, T. Chen, M. Hong, Y. Wang, and S. Liu, "Advancing model pruning via bi-level optimization," *Advances in Neural Information Processing Systems*, vol. 35, pp. 18 309–18 326, 2022.

[56] V. Sehwag, S. Wang, P. Mittal, and S. Jana, "Hydra: Pruning adversarially robust neural networks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 19 655–19 666, 2020.

[57] H. Wang and Y. Fu, "Trainability preserving neural pruning," *arXiv preprint arXiv:2207.12534*, 2022.

[58] G. K. Dziugaite and D. M. Roy, "Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data," in *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence, UAI 2017, Sydney, Australia, August 11-15, 2017*, G. Elidan, K. Kersting, and A. Ihler, Eds. AUAI Press, 2017. [Online]. Available: http://auai.org/uai2017/proceedings/papers/173.pdf

[59] N. S. Chatterji, B. Neyshabur, and H. Sedghi, "The intriguing role of module criticality in the generalization of deep networks," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. [Online]. Available: https://openreview.net/forum?id=S1e4jkSKvB

[60] H. Zhou, J. Lan, R. Liu, and J. Yosinski, "Deconstructing lottery tickets: Zeros, signs, and the supermask," *Advances in neural information processing systems*, vol. 32, 2019.

[61] Y. LeCun, J. S. Denker, and S. A. Solla, "Optimal brain damage," in *Advances in Neural Information Processing Systems 2, [NIPS Conference, Denver, Colorado, USA, November 27-30, 1989]*, D. S. Touretzky, Ed. Morgan Kaufmann, 1989, pp. 598–605. [Online]. Available: http://papers.nips.cc/paper/250-optimal-brain-damage

[62] T. Liang, J. Glossner, L. Wang, S. Shi, and X. Zhang, "Pruning and quantization for deep neural network acceleration: A survey," *Neurocomputing*, vol. 461, pp. 370–403, 2021. [Online]. Available: https://doi.org/10.1016/j.neucom.2021.07.045

[63] L. Yin, Y. Wu, Z. Zhang, C.-Y. Hsieh, Y. Wang, Y. Jia, G. Li, A. Jaiswal, M. Pechenizkiy, Y. Liang *et al.*, "Outlier weighed layerwise sparsity (owl): A missing secret sauce for pruning llms to high sparsity," *arXiv preprint arXiv:2310.05175*, 2023.

[64] T. Kumar, K. Luo, and M. Sellke, "No free prune: Information-theoretic barriers to pruning at initialization," *arXiv preprint arXiv:2402.01089*, 2024.

[65] H. Wang, J. Zhang, and Q. Ma, "Exploring intrinsic dimension for vision-language model pruning," in *Forty-first International Conference on Machine Learning*.

[66] G. Bai, Y. Li, C. Ling, K. Kim, and L. Zhao, "Sparsellm: Towards global pruning of pre-trained language models," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[67] Y. He, X. Zhang, and J. Sun, "Channel pruning for accelerating very deep neural networks," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. IEEE Computer Society, 2017, pp. 1398–1406. [Online]. Available: https://doi.org/10.1109/ICCV.2017.155

[68] Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan, and C. Zhang, "Learning efficient convolutional networks through network slimming," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. IEEE Computer Society, 2017, pp. 2755–2763. [Online]. Available: https://doi.org/10.1109/ICCV.2017.298

[69] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf, "Pruning filters for efficient convnets," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. [Online]. Available: https://openreview.net/forum?id=rJqFGTslg

[70] H. Hu, R. Peng, Y. Tai, and C. Tang, "Network trimming: A data-driven neuron pruning approach towards efficient deep architectures," *CoRR*, vol. abs/1607.03250, 2016. [Online]. Available: http://arxiv.org/abs/1607.03250

[71] W. Wen, C. Wu, Y. Wang, Y. Chen, and H. Li, "Learning structured sparsity in deep neural networks," in *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, and R. Garnett, Eds., 2016, pp. 2074–2082.

[72] C. Hong, A. Sukumaran-Rajam, B. Bandyopadhyay, J. Kim, S. E. Kurt, I. Nisa, S. Sabhlok, Ü. V. Çatalyürek, S. Parthasarathy, and P. Sadayappan, "Efficient sparse-matrix multi-vector product on gpus," in *Proceedings of the 27th International Symposium on High-Performance Parallel and Distributed Computing, HPDC 2018, Tempe, AZ, USA, June 11-15, 2018*, M. Zhao, A. Chandra, and L. Ramakrishnan, Eds. ACM, 2018, pp. 66–79. [Online]. Available: https://doi.org/10.1145/3208040.3208062

[73] M. Nonnenmacher, T. Pfeil, I. Steinwart, and D. Reeb, "SOSP: efficiently capturing global correlations by second-order structured pruning," in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. [Online]. Available: https://openreview.net/forum?id=t5EmXZ3ZLR

[74] M. E. Halabi, S. Srinivas, and S. Lacoste-Julien, "Data-efficient structured pruning via submodular optimization," in *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., 2022.

[75] M. Yin, B. Uzkent, Y. Shen, H. Jin, and B. Yuan, "GOHSP: A unified framework of graph and optimization-based heterogeneous structured pruning for vision transformer," in *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, B. Williams, Y. Chen, and J. Neville, Eds. AAAI Press, 2023, pp. 10 954–10 962. [Online]. Available: https://doi.org/10.1609/aaai.v37i9.26298

[76] A. Nova, H. Dai, and D. Schuurmans, "Gradient-free structured pruning with unlabeled data," in *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, ser. Proceedings of Machine Learning Research, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, Eds., vol. 202. PMLR, 2023, pp. 26 326–26 341. [Online]. Available: https://proceedings.mlr.press/v202/nova23a.html

[77] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *arXiv preprint arXiv:1510.00149*, 2015.

[78] Z. Liu, M. Sun, T. Zhou, G. Huang, and T. Darrell, "Rethinking the value of network pruning," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. [Online]. Available: https://openreview.net/forum?id=rJlnB3C5Ym

[79] S. Cao, C. Zhang, Z. Yao, W. Xiao, L. Nie, D. Zhan, Y. Liu, M. Wu, and L. Zhang, "Efficient and effective sparse LSTM on FPGA with bank-balanced sparsity," in *Proceedings of the 2019 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, FPGA 2019, Seaside, CA, USA, February 24-26, 2019*, K. Bazargan and S. Neuendorffer, Eds. ACM, 2019, pp. 63–72. [Online]. Available: https://doi.org/10.1145/3289602.3293898

[80] A. Ren, T. Zhang, S. Ye, J. Li, W. Xu, X. Qian, X. Lin, and Y. Wang, "ADMM-NN: an algorithm-hardware co-design framework of dnns using alternating direction methods of multipliers," in *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS 2019, Providence, RI, USA, April 13-17, 2019*, I. Bahar, M. Herlihy, E. Witchel, and A. R. Lebeck, Eds. ACM, 2019, pp. 925–938. [Online]. Available: https://doi.org/10.1145/3297858.3304076

[81] T. Zhang, S. Ye, K. Zhang, J. Tang, W. Wen, M. Fardad, and Y. Wang, "A systematic DNN weight pruning framework using alternating direction method of multipliers," in *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VIII*, ser. Lecture Notes in Computer Science, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., vol. 11212. Springer, 2018, pp. 191–207. [Online]. Available: https://doi.org/10.1007/978-3-030-01237-3_12

[82] M. Alizadeh, S. A. Tailor, L. M. Zintgraf, J. van Amersfoort, S. Farquhar, N. D. Lane, and Y. Gal, "Prospect pruning: Finding trainable weights at initialization using meta-gradients," in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. [Online]. Available: https://openreview.net/forum?id=AIgn9uwfcD1

[83] Z. Liao, V. Quétu, V. Nguyen, and E. Tartaglione, "Can unstructured pruning reduce the depth in deep neural networks?" *CoRR*, vol. abs/2308.06619, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2308.06619

[84] C. Zhang, S. Bengio, and Y. Singer, "Are all layers created equal?" *Journal of Machine Learning Research*, vol. 23, no. 67, pp. 1–28, 2022.

[85] Y. Yao, X. Xu, and Y. Liu, "Large language model unlearning," *arXiv preprint arXiv:2310.10683*, 2023.

[86] K. Zhao, M. Kurmanji, G.-O. Bărbulescu, E. Triantafillou, and P. Triantafillou, "What makes unlearning hard and what to do about it," *arXiv preprint arXiv:2406.01257*, 2024.

[87] Y. Li, C. Wang, and G. Cheng, "Online forgetting process for linear regression models," in *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event*, ser. Proceedings of Machine Learning Research, A. Banerjee and K. Fukumizu, Eds., vol. 130. PMLR, 2021, pp. 217–225. [Online]. Available: http://proceedings.mlr.press/v130/li21a.html

[88] A. Mahadevan and M. Mathioudakis, "Certifiable machine unlearning for linear models," *CoRR*, vol. abs/2106.15093, 2021. [Online]. Available: https://arxiv.org/abs/2106.15093

[89] J. Brophy and D. Lowd, "Machine unlearning for random forests," in *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 2021, pp. 1092–1104. [Online]. Available: http://proceedings.mlr.press/v139/brophy21a.html

[90] A. Ginart, M. Y. Guan, G. Valiant, and J. Zou, "Making AI forget you: Data deletion in machine learning," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 3513–3526.

[91] Y. Cao and J. Yang, "Towards making systems forget with machine unlearning," in *2015 IEEE symposium on security and privacy*. IEEE, 2015, pp. 463–480.

[92] E. Ullah, T. Mai, A. Rao, R. A. Rossi, and R. Arora, "Machine unlearning via algorithmic stability," in *Conference on Learning Theory, COLT 2021, 15-19 August 2021, Boulder, Colorado, USA*, ser. Proceedings of Machine Learning Research, M. Belkin and S. Kpotufe, Eds., vol. 134. PMLR, 2021, pp. 4126–4142. [Online]. Available: http://proceedings.mlr.press/v134/ullah21a.html

[93] M. Chen, Z. Zhang, T. Wang, M. Backes, M. Humbert, and Y. Zhang, "When machine unlearning jeopardizes privacy," in *CCS '21: 2021 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, Republic of Korea, November 15 - 19, 2021*, Y. Kim, J. Kim, G. Vigna, and E. Shi, Eds. ACM, 2021, pp. 896–911. [Online]. Available: https://doi.org/10.1145/3460120.3484756

[94] E. Chien, H. P. Wang, Z. Chen, and P. Li, "Certified machine unlearning via noisy stochastic gradient descent," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

[95] Y.-H. Park, S. Yun, J.-H. Kim, J. Kim, G. Jang, Y. Jeong, J. Jo, and G. Lee, "Direct unlearning optimization for robust and safe text-to-image models," *arXiv preprint arXiv:2407.21035*, 2024.

[96] N. G. Marchant, B. I. P. Rubinstein, and S. Alfeld, "Hard to forget: Poisoning attacks on certified machine unlearning," in *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*. AAAI Press, 2022, pp. 7691–7700. [Online]. Available: https://doi.org/10.1609/aaai.v36i7.20736

[97] Y. Wu, E. Dobriban, and S. B. Davidson, "Deltagrad: Rapid retraining of machine learning models," in *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, ser. Proceedings of Machine Learning Research, vol. 119. PMLR, 2020, pp. 10 355–10 366. [Online]. Available: http://proceedings.mlr.press/v119/wu20b.html

[98] S. Neel, A. Roth, and S. Sharifi-Malvajerdi, "Descent-to-delete: Gradient-based methods for machine unlearning," in *Algorithmic Learning Theory, 16-19 March 2021, Virtual Conference, Worldwide*, ser. Proceedings of Machine Learning Research, V. Feldman, K. Ligett, and S. Sabato, Eds., vol. 132. PMLR, 2021, pp. 931–962. [Online]. Available: http://proceedings.mlr.press/v132/neel21a.html

[99] J. Foster, S. Schoepf, and A. Brintrup, "Fast machine unlearning without retraining through selective synaptic dampening," in *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, M. J. Wooldridge, J. G. Dy, and S. Natarajan, Eds. AAAI Press, 2024, pp. 12 043–12 051. [Online]. Available: https://doi.org/10.1609/aaai.v38i11.29092

[100] V. S. Chundawat, A. K. Tarun, M. Mandal, and M. S. Kankanhalli, "Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher," in *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, B. Williams, Y. Chen, and J. Neville, Eds. AAAI Press, 2023, pp. 7210–7217. [Online]. Available: https://doi.org/10.1609/aaai.v37i6.25879

[101] J. Frankle and M. Carbin, "The lottery ticket hypothesis: Finding sparse, trainable neural networks," *arXiv preprint arXiv:1803.03635*, 2018.

A Technical Appendices and Supplementary Material

As Figure 10 11 and 12 shown, MIA would be significantly changed by slightly adjusting a bit of samples. Therefore, even when the training and test sets in the shadow dataset are approximately in a 1:1 ratio, adjusting the data within the error margin (±5%) can yield completely opposite results. Due to the low robustness of the results, we consider MIA to be unreliable for evaluating unlearning in sparse models.

As Tab 7 mentioned, each row contains several sparsifications (i.e., dense and a few sparsity-level) and 8 situations (Original, retraining, and 6 unlearn methods). We ensure that all experimental conditions remain consistent, with the only difference being in the training data (the original experiment uses the full dataset, while the retrain experiment uses a training set where 20% of the data is removed based on random seed 42). In some cases, high sparsity leads to a catastrophic collapse in accuracy (e.g., 90% and 95% sparsity sometimes result in only 10% accuracy). Therefore, we omit these clearly broken results when presenting our results.

Discussion We still need to find or establish better metrics to evaluate whether some unlearn methods remove the impact of the deleted data or not. We should comprehensively consider the impact of data on all aspects such as the performance of downstream tasks or the topology of sparse models (weight distribution, connectivity patterns, and gradient dynamics). Furthermore, expanding experiments to a wider range of model architectures and datasets would enhance the generalizability of findings. While our study focuses on a specific setting, evaluating unlearned methods in architectures like Transformers or DDPM could reveal model-dependent behaviors.

Table 5: Performance comparison on the ImageNet dataset (ResNet18)

Method	IOU	IOM	TA	UA	Sparsity	Del. Ratio
Retraining + repruning	–	–	56.57%	54.87%	19.03%	20%
Un-pruning (WF)	84.22%	85.06%	57.42%	55.42%	19.03%	20%

Table 6: Performance on ViT

Method	IOU	IOM	TA	UA	Sparsity	Del. Ratio
Retraining + repruning	–	–	72.68%	85.29%	41%	20%
Un-pruning (WF)	45.78%	64.05%	70.32%	86.44%	41%	20%

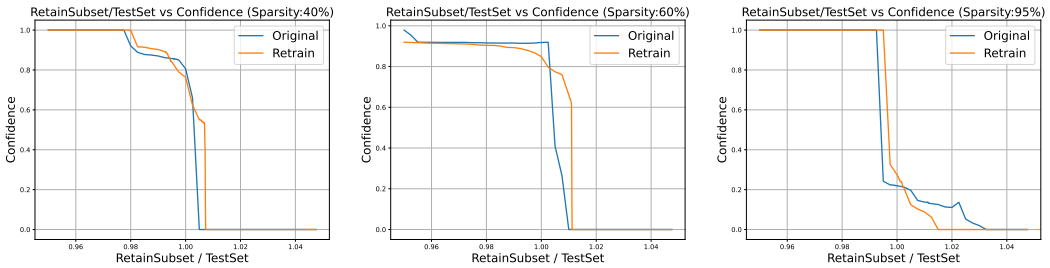

Figure 9: MIA confidence.

Table 7: Hyper Parameters Settings

Type	Method	Model	Dataset	Learning Rate	Random Seed	Deleted Ratio
Structured	[54]	ResNet-20	Cifar10	1e-3	42	20%
Unstructured	[101]	ResNet-18	Cifar10	1e-3	42	20%
Unstructured	[101]	ResNet-18	FashionMNIST	1e-3	42	20%

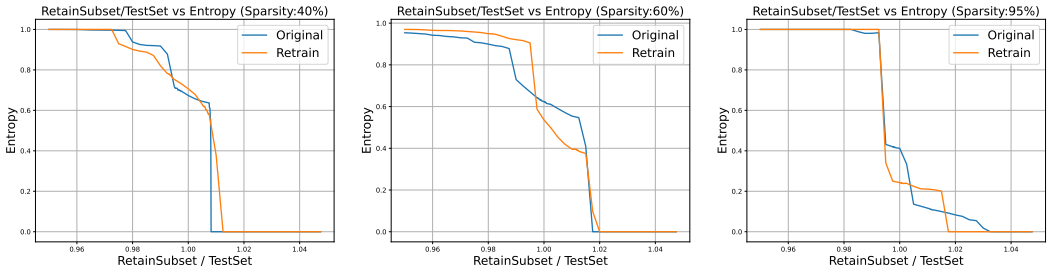

Figure 10: MIA Entropy.

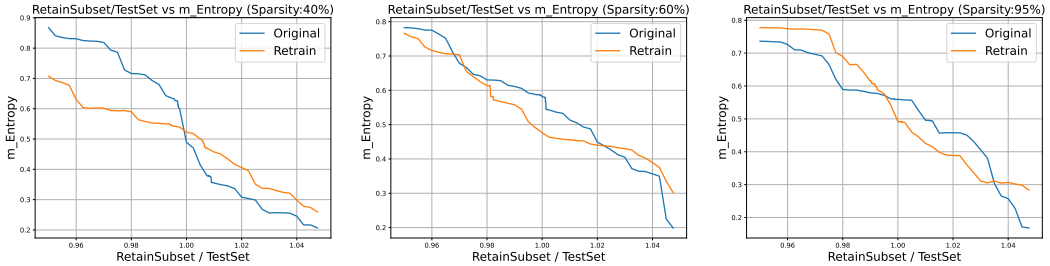

Figure 11: MIA m Entropy.

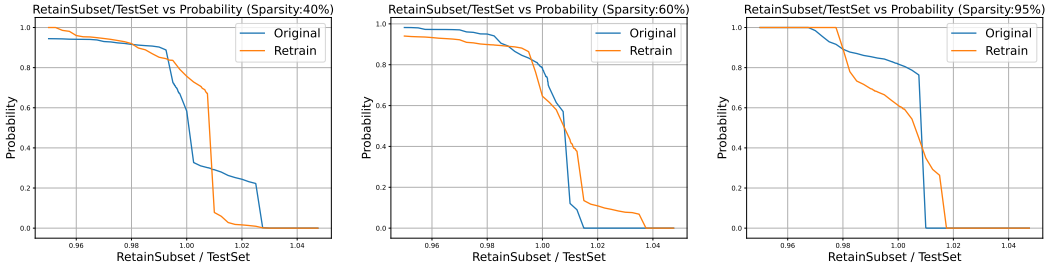

Figure 12: MIA Probability.

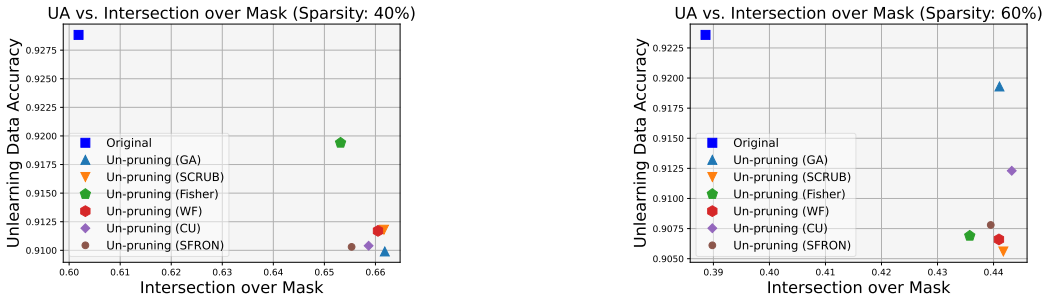

Figure 13: Unstructured un-pruning (IoM) in FashionMNIST

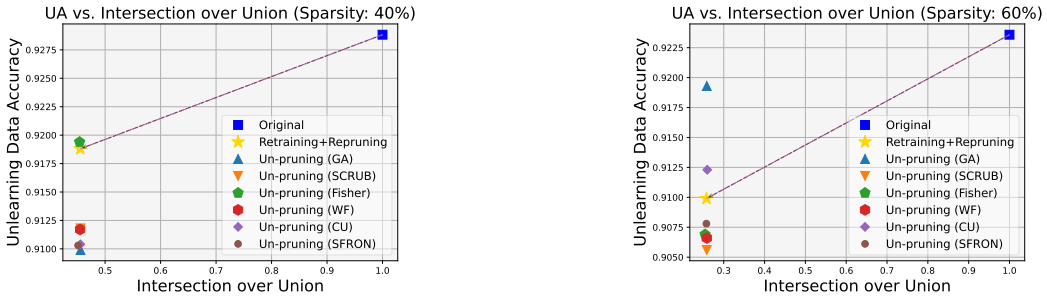

Figure 14: Unstructured un-pruning (IoU) in FashionMNIST.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and/or introduction clearly states the claims made, including the contributions made in the paper and important assumptions and limitations.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations are explicitly discussed in Section 5 including.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: All theoretical results (see Section 2 and Section 3) are accompanied by clearly stated assumptions and full proofs.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We describe all implementation details in Section 4, Section 5 and Appendix **??**, including hyper-parameters and training setup.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code is available at `https://anonymous.4open.science/r/UnlearningSparseModels-FBC5/`.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We describe all implementation details in Section 4, Section 5 and Appendix **??**, including hyper-parameters and training setup.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: We do not report error bars or statistical significance tests due to computational constraints.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We report the training time of all unlearning methods in Table 4 Experiments were conducted on a server with NVIDIA L40S GPU (48GB). Each method was trained using a single GPU.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and confirm that our research adheres to all ethical guidelines. Our study does not involve human subjects, sensitive data, or potentially harmful applications, and we have ensured transparency, reproducibility, and fairness throughout.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: Our work contributes to trustworthy AI by improving the ability of sparse models to forget specific training data, which is valuable for compliance with data privacy regulations such as GDPR and for reducing memory footprint in deployment.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our work does not involve pretrained generative models, scraped datasets, or other assets that pose a high risk of misuse. Therefore, no special safeguards are necessary.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All third-party assets used in our work are properly credited and licensed. We use the CIFAR10, CIFAR100 and Imagenet dataset, which is publicly available under a CC BY 4.0 license, and cite the original paper.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: This paper does not introduce any new datasets, codebases, or models. We rely solely on existing open-source assets which are properly cited.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

Justification: This paper does not involve any human subjects research and therefore does not require IRB approval.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: This paper does not involve any important, original, or non-standard use of LLMs in the core methodology. Any language models were used only for minor editing or formatting purposes.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.