Model Unlearning Objectives Vary for Distinct Language Functions

Anonymous ACL submission

Abstract

Large language models (LLMs) learn undesirable properties during pretraining, including dangerous knowledge and toxic text generation. Just as post-training uses different objectives to shape different behaviors, we argue that unlearning methods should be designed for the language function at issue. To study this, we consider two mechanistically distinct unlearning goals, dangerous-knowledge unlearning and toxicity unlearning. For dangerous knowledge, we introduce a cosine-based, meta-learned variant of RMU. For toxicity, we propose a multi-layer objective based on layer-specific probe directions. Across four open-source 7-8B models, our methods achieve strong results, based on distinct training objectives for the two types of unlearning. Overall, our results suggest that unlearning should be studied as a family of problems, analogous to the multiple types of LLM post-training.

1 Introduction

Language serves a wide range of functions in human society: it is used not only to convey information, but also to coordinate action, manage social relationships, and express attitudes, intentions, and norms. Large language models (LLMs) acquire great linguistic fluency through large-scale pretraining, and post-training methods are then used to shape behaviors such as helpfulness, instruction-following, and safer response behavior (Ouyang et al., 2022; Rafailov et al., 2023; Du et al., 2025). However, these methods do not cover the full range of communicative abilities and norms that human language use fulfills. Models learn hazardous knowledge and socially harmful behaviors, such as toxic language generation, from their training data (Brown et al., 2020; Gehman et al., 2020; Li et al., 2024). This has motivated growing interest in *unlearning*: post hoc interventions that aim to remove specific knowledge, capabilities, or behaviors from already-trained models through finetuning (Cao and Yang, 2015; Bourtoule et al., 2021; Liu et al., 2024a; Maini et al., 2024).

Our central claim is that, similar to post-training itself, *unlearning is goal-dependent*. Modern LLM pipelines have distinct post-training procedures for a range of distinct functions: instruction following, preference alignment to human values, refusal behavior, and style control require different objectives (Ouyang et al., 2022; Rafailov et al., 2023; Du et al., 2025). Mechanistic work further suggests that some of these properties are altered more substantially by post-training than others. In particular, Du et al. (2025) show that factual knowledge storage locations remain largely stable across base and post-trained models, and that a truthfulness direction is also highly similar across the two, whereas refusal directions change substantially after SFT and instruction tuning. We argue that unlearning should take such redmechanistic observations into account. Removing dangerous knowledge is not the same problem as removing socially undesirable behavior, because these functions are represented differently inside the model.

We study two unlearning problems that mechanistic evidence suggests could be different: removal of *dangerous knowledge* versus *toxic language*. Dangerous knowledge concerns the model's ability to access factual or procedural information, as in biosecurity-oriented settings such as WMDP (Li et al., 2024). Toxicity, by contrast, concerns the tendency to generate abusive or harmful language (Gehman et al., 2020; Hartvigsen et al., 2022). Existing work (Kadhe et al., 2024) has treated both as instances of the same unlearning problem, but our results suggest otherwise, which is supported by recent mechanistic studies. For factual knowledge, prior work points to a relatively structured retrieval mechanism (Meng et al., 2022). Knowledge relevant to a statement is concentrated at subject, object, and last-token positions, with subject

information strongest in earlier layers, object information in early-to-middle layers, and the last token becoming especially important in middle-to-late layers (Meng et al., 2022; Geva et al., 2023; Du et al., 2025). Du et al. (2025) further show that post-training largely preserves these knowledge-storage locations. At the same time, current hazardous-knowledge unlearning methods remain limited: they can be shallow or recoverable, suggesting that simply steering away from undesirable behavior might not be enough (Hu et al., 2024; Deeb and Roger, 2024; Dang et al., 2024, 2025).

The mechanisms underlying toxicity in LLMs differ from those for factual knowledge. Lee et al. (2024) train a linear toxicity probe on averaged final-layer representations and identify value vectors aligned with a toxicity direction. They show that toxicity is largely elicited in later MLP layers, and that subtracting those vectors can reduce toxic outputs. Crucially, after DPO, the toxic vectors largely remain; instead, small accumulated changes in the model alter activations so that the model bypasses toxicity-promoting regions rather than erasing the underlying capability (Lee et al., 2024). These findings suggest **complementary roles for alignment and unlearning**: alignment can reduce toxic behavior, while unlearning can weaken the underlying toxicity-relevant directions.

Motivated by this difference, we argue that effective unlearning requires a better understanding of *what is being forgotten*. At a high level, unlearning methods aim to reduce undesirable behavior while preserving desirable behavior, which is reflected in the standard combination of a forget objective and a retain objective used across prior work (Li et al., 2024; Huu-Tien et al., 2024; Zhang et al., 2024). We claim, however, that the formulation of the forget objective should depend on the unlearning goal. We introduce a new unlearning method for dangerous knowledge that replaces RMU's L2 objective (Li et al., 2024) with a cosine objective, and adaptively learns the retain-forget tradeoff using reinforcement learning. For toxicity unlearning, we propose a forget loss that targets toxicity-relevant signals across multiple layers. For assessment, we introduce a unified evaluation metric that summarizes the tradeoff between forgetting and retention.

2 Related Work

In this section, we review work to understand how LLMS store undesirable knowledge and toxicity.

2.1 LLM Unlearning

Unlearning was originally framed as removing the influence of specific training data without full retraining (Cao and Yang, 2015; Bourtoule et al., 2021). In LLMs, exact retraining-based guarantees are usually infeasible, so recent work instead relies on benchmark-specific forgetting metrics (Liu et al., 2024a). Benchmarks such as TOFU (Maini et al., 2024) and WMDP (Li et al., 2024) have become standard datasets for fictitious-profile forgetting and hazardous-knowledge unlearning, respectively. Existing methods include gradient-based forget objectives, preference-based methods such as NPO (Zhang et al., 2024), and representation-level steering approaches such as RMU (Li et al., 2024) or Spunge (Kadhe et al., 2024).

2.2 Unlearning Dangerous Knowledge

A major line of work studies unlearning of dangerous factual or procedural knowledge, especially through WMDP-style evaluations (Li et al., 2024). However, unlearned knowledge can often be recovered through targeted relearning, substantial information may remain in model weights, and steering-based methods can reduce robustness or induce nonsensical behavior rather than cleanly removing the target capability (Hu et al., 2024; Deeb and Roger, 2024; Dang et al., 2024, 2025). Our method aims to provide a more principled objective for hazardous-knowledge unlearning.

Mechanistic work helps clarify the problem of unlearning dangerous knowledge. Factual recall depends strongly on subject, object, and last-token positions across different layer ranges (Meng et al., 2022; Geva et al., 2023; Du et al., 2025). Du et al. (2025) further shows that post-training largely preserves these knowledge-storage locations, suggesting that factual competence is anchored in relatively stable internal structure. Zou et al. (2023) found that representation-level interventions offer more control on safety-relevant capabilities. This motivates our representation-level approach that aims to weaken dangerous factual competence while preserving general utility, by directly targeting representations that support factual recall.

2.3 Toxicity Behaviour

A separate line of work studies harmful language generation, including toxicity and abusive behavior (Gehman et al., 2020; Hartvigsen et al., 2022). Mechanistic evidence suggests that toxicity is rep-

resented differently from factual knowledge. Lee et al. (2024) show that a toxicity direction can be extracted from final-layer hidden states and that later-layer value vectors aligned with this direction can modulate toxic outputs. Yet apparently, DPO does not remove these toxic vectors; instead, it introduces distributed offsets that bypass toxicity-eliciting regions (Lee et al., 2024). More generally, this suggests that alignment may reduce toxic behavior without fully removing the underlying toxic capability. Du et al. (2025) similarly show that refusal directions shift substantially across post-training, unlike truthfulness directions, reinforcing the view that safety behaviors are more post-training-dependent and less structurally stable than factual knowledge.

When we tried an RMU-like approach for toxicity, the results were poor. Toxicity is distributed across layers, this motivates our multi-layer toxicity unlearning objective.

$$L = L_{\text{F}} + \alpha \cdot L_{\text{R}} \tag{1}$$

$$L_{\text{F}} = \mathbb{E}_{x \sim D_{\text{F}}} \left[\sum_{t \in x_f} \| M_{\text{updated}}(t) - c \cdot \mathbf{u} \|_2^2 \right] \tag{2}$$

$$L_{\text{R}} = \mathbb{E}_{x \sim D_{\text{R}}} \left[\sum_{t \in x_r} \| M_{\text{updated}}(t) - M_{\text{frozen}}(t) \|_2^2 \right] \tag{3}$$

3 Baselines For Unlearning

In this section, we review the main baseline approaches that we build upon.

3.1 RMU

Representation Misdirection for Unlearning (RMU) is a fine-tuning approach aimed at selectively removing unwanted knowledge from LLMs (Li et al., 2024). RMU operates on two datasets: *forget data*, containing the target knowledge or behavior to be unlearned, and *retain data*, containing general examples used to preserve the model's desirable capabilities. It pushes the model's representations on forget data toward randomly initialized vectors, while also encouraging representations on retain data to remain close to those of the original frozen model. The overall objective is shown in Equation 1. The forget loss is shown in Equation 2 where $\mathbf{u}$ is a random unit vector and c is a scaling factor. Similarly, L_{retain} refers to the retain loss and is taken as an expectation over the entire retain dataset, as shown in Equation 3.

3.2 AdapRMU

Huu-Tien et al. (2024), focus on the scaling coefficient c in Equation 2. While the direction $\mathbf{u}$ is fixed before unlearning, c determines the magnitude of the representation shift. AdapRMU adaptively adjusts c based on the norm of the forget representation.

4 Methodology

In this section, we present our modifications to RMU. We make two changes for dangerous-knowledge unlearning and then introduce a separate objective for toxicity. We also present our metric to reflect the tradeoff between unlearning and general capability.

$$\mathcal{L}_{\text{F}} = \mathbb{E}_{x_f \sim D_{\text{F}}} \left[\sum_{t \in x_f} \left(1 - \frac{M_{\text{updated}}(t) \cdot (c \cdot \mathbf{u})}{\| M_{\text{updated}}(t) \|_2 \| c \cdot \mathbf{u} \|_2} \right) \right] \tag{4}$$

$$\mathcal{L}_{\text{R}} = \mathbb{E}_{x_r \sim D_{\text{R}}} \left[\sum_{t \in x_r} \left(1 - \frac{M_{\text{updated}}(t) \cdot M_{\text{frozen}}(t)}{\| M_{\text{updated}}(t) \|_2 \| M_{\text{frozen}}(t) \|_2} \right) \right] \tag{5}$$

4.1 Cosine Loss instead of L2 Loss

We replace the original L2 losses in Equations 2 and 3 with cosine-based losses shown in Equations 4 and 5. Our hypothesis is that the key quantity for both forgetting and retention is representational *direction* rather than exact Euclidean position. For forgetting, cosine loss should work better because it directly optimizes alignment between the updated representation and the target direction, instead of also penalizing norm differences. For retention, cosine loss better preserves the frozen model's representational orientation on retain data while tolerating harmless changes in magnitude induced by fine-tuning. In this sense, cosine-based losses are a better match to our intended geometry: forgetting should steer representations toward or away from a direction, and retention should preserve directional structure rather than exact scale. Prior work similarly notes that L2 distance can be mismatched when the task is fundamentally angular, while cosine-based objectives can be more robust to norm variation (Xu et al., 2018).

4.2 Meta Learning the Coefficient α

In Equation 1, the coefficient α balances the forget and retain objectives. In RMU and AdapRMU, α

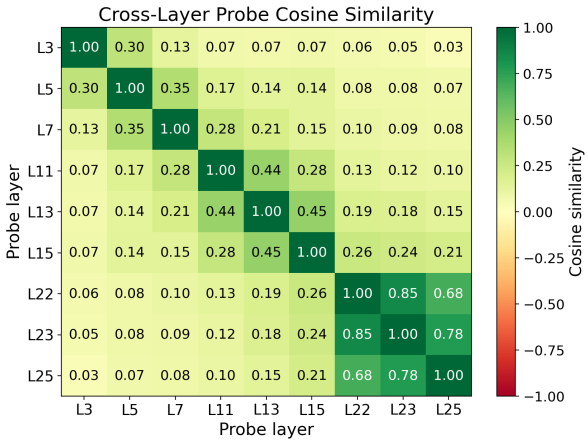


Figure 1: Cosine similarities between logistic-regression probes trained at different layers of Llama3.1-8B.

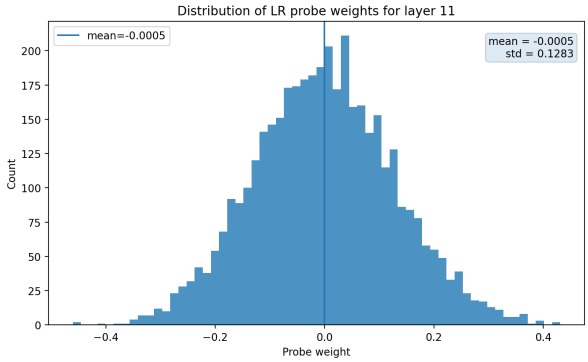


Figure 2: Weight distribution of the logistic-regression probe trained on layer 11 of Llama3.1-8B for toxicity classification.

is fixed throughout training. We found this to be a limitation, because we observed the best value of α to vary substantially across models and datasets. To address this, we treat α as a learnable parameter and update it during fine-tuning using REINFORCE (Williams, 1992). At each step, the policy update adjusts α toward values that improve the overall objective, allowing the model to adaptively balance forgetting and retention. This removes the need for extensive manual tuning and leads to more consistent performance across models.

In more detail, REINFORCE evaluates how the α value affects the total loss, then adjusts α in the direction that lowers the loss most effectively. In this way, the model learns the best possible balance between forgetting unwanted information and preserving useful knowledge. The result is an α value that adapts automatically to different models, thus yielding more consistent improvement.

4.3 Toxicity Unlearning

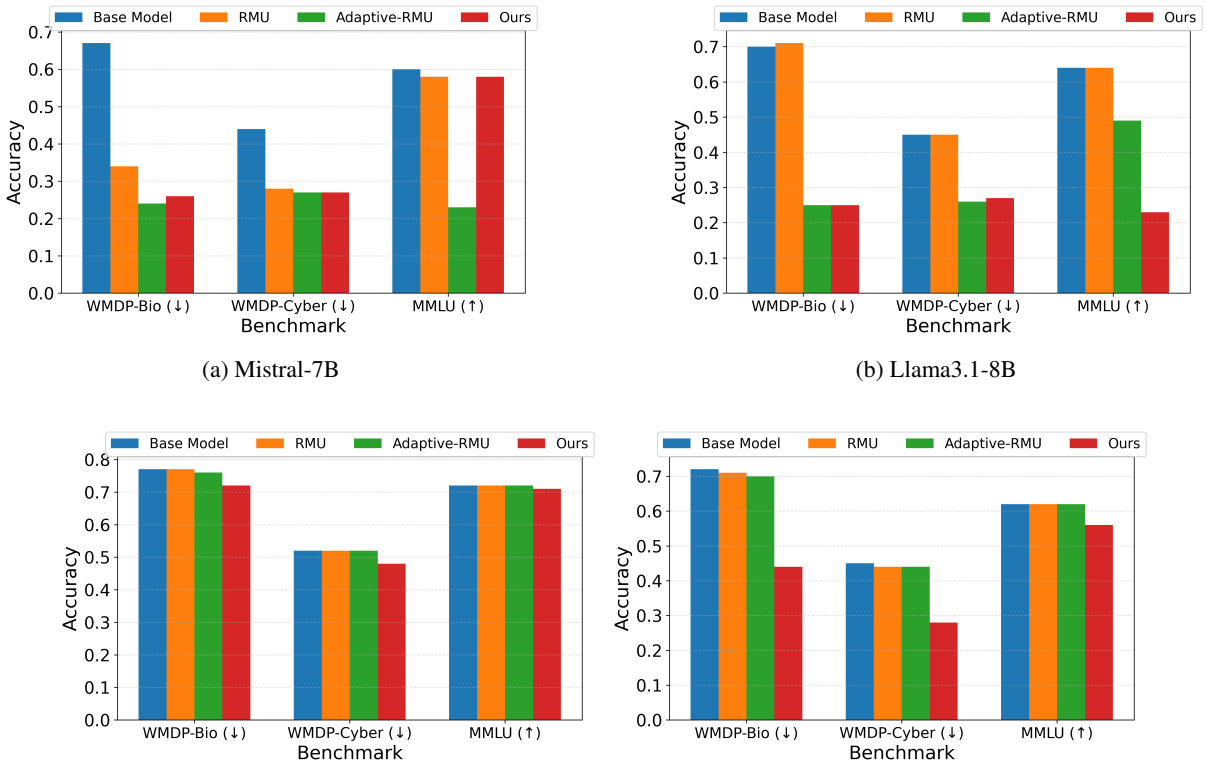
We initially attempted to use the same loss function for toxicity, but we found that simply reusing the same objective designed for hazardous knowledge is ineffective for toxicity (yielding only a 2-3% decrease in toxicity compared to 6-24% we find).

Building on the mechanistic findings in Section 2.3, we directly target the empirical toxicity direction in representation space. We operationalize this by training L2-regularized logistic regression probes on last-token hidden representations from a frozen reference model at multiple layers, obtaining a toxicity direction LR^i for each target layer $i \in I$. We use last-token representations rather than mean pooling because averaging suppresses useful information, which is borne out in our experiments showing that last-token probes consistently achieve higher AUC. We also find that cross-layer cosine similarities between probe directions are often low, indicating that toxicity rotates substantially across depth. This suggests that a single shared direction is not sufficient to characterize toxicity across the network. We therefore train layer-specific probes and intervene at multiple layers at dispersed depths. Below, we report an ablation study for different combinations of layers that verify the utility of both multiple layers, and dispersed depths.

$$\mathcal{L}_{\text{F}} = \mathbb{E}_{x_t \sim D_{\text{toxic}}} \left[\sum_{i \in I} \left(M^i_{\text{updated}}(x_t[-1]) \cdot LR^i \right)^2 \right] \quad (6)$$

Our forget loss, shown in Equation 6, encourages the updated model's representations of toxic inputs to become orthogonal to the toxicity direction at each layer. Squaring the dot product is important for three reasons. First, it ensures a non-negative loss that is minimized when the representation is orthogonal to the toxicity direction. Second, it makes the gradient symmetric around zero. Third, it gives larger gradients to representations that are strongly aligned with the toxicity direction. Averaging across layers in I ensures that the toxic subspace is suppressed at every targeted depth, reducing the chance that toxicity suppressed in one layer re-emerges later in the network.

To preserve general language-modeling ability, we use the same retain loss as in dangerous-knowledge unlearning. The full objective is still given by Equation 1, with α adapted using the meta-learning procedure described above.



(a) Mistral-7B

(b) Llama3.1-8B

(c) Qwen2.5-7B

(d) Olmo3-7B

Figure 3: Dangerous knowledge unlearning results for the four models.

4.4 S-unlearning: a New Unlearning Metric

To summarize the tradeoff between forgetting and retained utility, we introduce **S-unlearning**. Let $U \in [0, 1]$ denote unlearning performance and $R \in [0, 1]$ retained utility. For dangerous knowledge, lower WMDP accuracy indicates better unlearning; for toxicity, lower toxicity indicates better unlearning. Because random guessing on our multiple-choice utility benchmark yields $R_0 = 0.25$, we first chance-correct utility as

$$\bar{R} = \left[\frac{R - R_0}{1 - R_0}\right]_0^1.$$

For dangerous knowledge, we similarly chance-correct unlearning accuracy, while for toxicity we define $\bar{U} = 1 - U$ so that higher is better. Our final score is

$$\text{S-Unlearning} = \bar{U} \cdot \bar{R}.$$

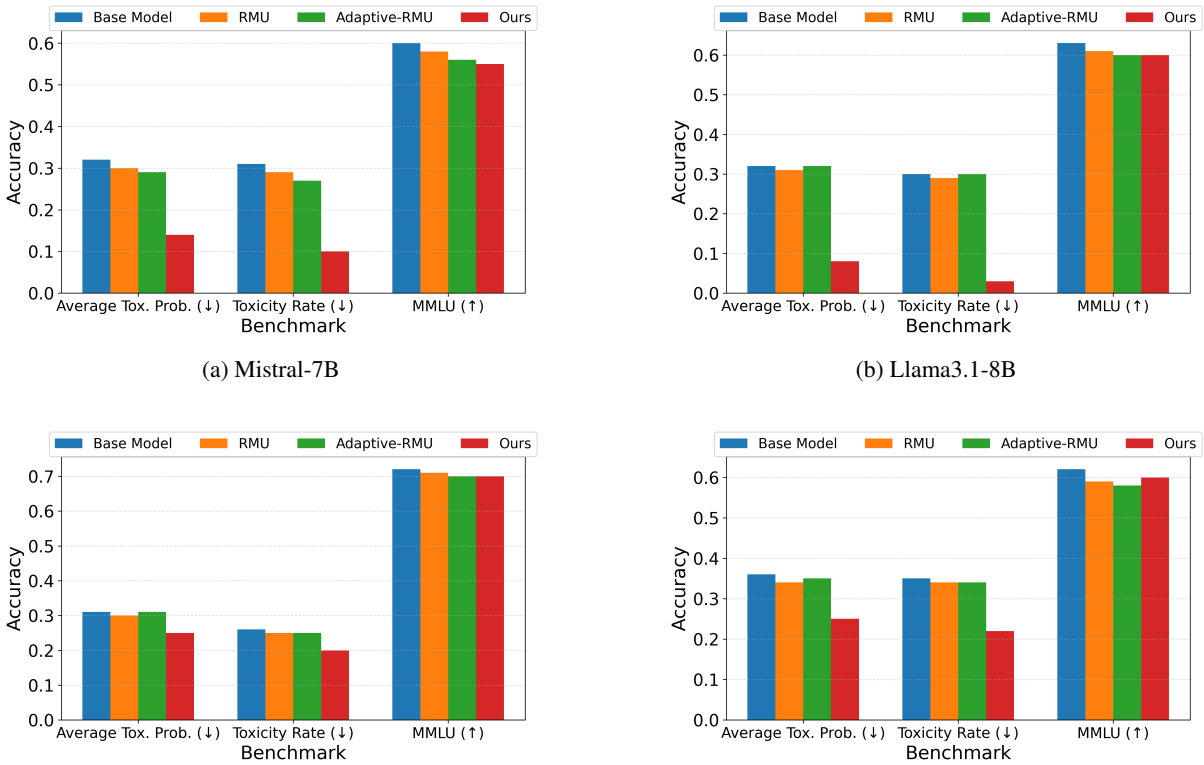
This score lies in $[0, 1]$, is maximized when unlearning is strong and retained utility is high, and becomes zero whenever retained utility is at or below chance. Geometrically, it is the area of the rectangle from the origin to $(\bar{U}, \bar{R})$ in normalized unlearning-utility space, i.e., the single-point two-dimensional special case of the hypervolume indicator (Auger et al., 2012).

5 Experimental Setup

Here we describe the datasets, models, and evaluation metrics used for both unlearning settings.

For dangerous-knowledge unlearning, we follow (Li et al., 2024): we use the biosecurity and cybersecurity forget datasets released with WMDP, evaluate forgetting on WMDP, and use WikiText as retain data. We measure retained general capabilities on MMLU (Hendrycks et al., 2021).

For toxicity unlearning, we use the toxic split of ParaDetox (Logacheva et al., 2022) as forget data. To train the logistic-regression probes, we use the human-annotated training split of TRuST (Atil et al., 2026). We again use WikiText as retain data and evaluate retained utility with MMLU. To evaluate toxicity, we use the challenging subset of RealToxicityPrompts (Gehman et al., 2020) and score outputs with the BERT-based TRuST classifier (Atil et al., 2026), which we chose because it generalizes better than alternative toxicity classifiers. For our main toxicity experiments, we update nine layers, selecting three layers each from early, middle, and late depth regions. We analyze alternative layer choices in Section 6.5. The final α values are 43, 78, 5.7, and 0.41 for Llama, Mistral, Qwen, and Olmo respectively.



(a) Mistral-7B

(b) Llama3.1-8B

(c) Qwen2.5-7B

(d) Olmo3-7B

Figure 4: Toxicity unlearning results for the four models.

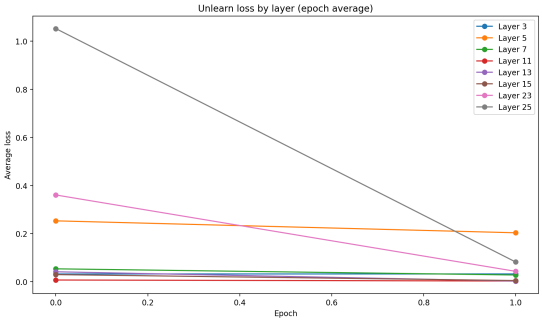
Unlearning Method	S-unlearning WMDP	S-unlearning Toxicity
Llama3.1-8B		
Base	0.29	0.34
RMU	0.29	0.33
Adaptive RMU	0.32	**0.32**
Ours	0.00	**0.43**
Mistral-7B		
Base	0.28	0.32
RMU	0.40	0.31
Adaptive RMU	0.00	0.29
Ours	**0.43**	**0.34**
Qwen2.5-7B		
Base	0.30	0.43
RMU	0.30	0.43
Adaptive RMU	0.30	0.41
Ours	**0.33**	**0.45**
Olmo3-7B		
Base	0.27	0.32
RMU	0.28	0.30
Adaptive RMU	0.28	0.29
Ours	**0.35**	**0.35**

Table 1: S-unlearning scores that combine the tradeoff between unlearning and general capability for both unlearning types.

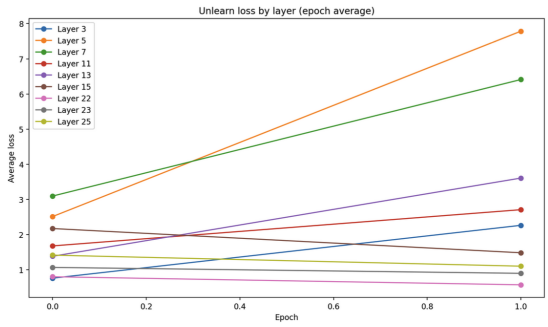
Given our limited compute budget, we focus on four widely used, smaller, open-source 7–8B models: Llama3.1-8B (Grattafiori et al., 2024), Mistral-7B (Jiang et al., 2023), Olmo3-7B (Olmo et al., 2025), and Qwen2.5-7B (Qwen et al., 2025).

6 Results

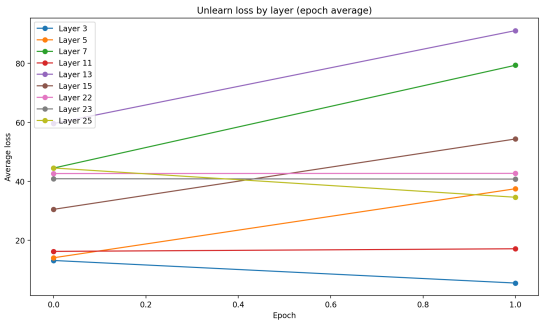
We first analyze the learned toxicity probes to better understand the geometry of toxicity representations. We then evaluate our methods on dangerous-knowledge and toxicity unlearning, examine where toxicity unlearning occurs, and conclude with an analysis of how many and which layers work best.



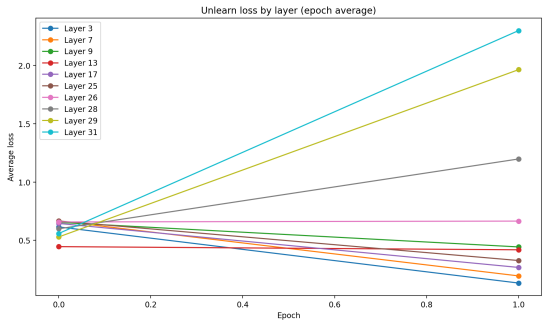
(a) Mistral-7B



(b) Llama3.1-8B



(c) Qwen2.5-7B



(d) Olmo3-7B

Figure 5: Toxicity unlearning loss curves for each model.

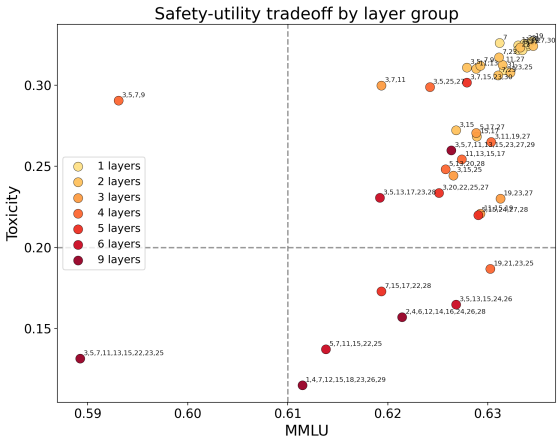


Figure 6: The effect of number and identity of the layers on toxicity unlearning and general capability

6.1 Toxicity Probe Analysis

To better understand the internal geometry of toxicity representations and obtain principled unlearning directions, we train logistic regression probes at multiple layers of Llama-3.1-8B and analyze the pairwise cosine similarity between the learned weight vectors. Figure 1 shows that toxicity directions are largely layer-specific: while adjacent layers share moderate similarity, most cross-layer pairs exhibit cosine similarities below 0.5, especially in the early layers. This indicates that the internal representation of toxicity rotates substantially as information propagates through the network. In the late layers, however, similarities are higher, suggesting a somewhat more consistent toxicity representation near the output. Overall, these results indicate that a single shared direction is insufficient to characterize toxicity across the full depth of the model, motivating our use of layer-specific probes.

Figure 2 shows the weight distribution of the probe trained at layer 11 of Llama3.1-8B. The distribution is approximately symmetric and centered close to zero, suggesting that the learned toxicity direction is dense and distributed rather than dominated by a small number of extreme dimensions. The spread of the distribution, spanning roughly $[-0.45, 0.45]$, indicates that many dimensions contribute meaningfully to the toxicity direction, with both positive and negative weights. This supports our use of the full probe vector as an unlearning direction and helps explain why sparse or single-neuron interventions are unlikely to be sufficient.

6.2 Dangerous Knowledge Unlearning

Figure 3 presents the accuracy results for dangerous-knowledge unlearning. As shown, our approach outperforms prior methods for all models except Llama3.1-8B, while preserving retained

utility well. All methods remain challenging for Qwen and Olmo models. Table 1 shows our method achieves the best S-unlearning tradeoff on all models except Llama3.1-8B.

6.3 Toxicity Unlearning

Figure 4 presents the accuracy results for toxicity unlearning. As shown, our method yields the largest reductions in toxicity balanced by limited reduction in general knowledge. In contrast, RMU and AdapRMU only slightly reduce toxicity. As shown in Table 1, our method also has the highest S-unlearning scores. The magnitude of improvement varies by model, with the strongest gains on Mistral and Llama, but our method consistently reduces toxic generation.

6.4 Where Does Toxicity Unlearning Occur?

To understand which layers are most affected during unlearning, we plot the layer-wise unlearning loss throughout training in Figure 5. Early layers begin with relatively low loss and remain low or decrease further, suggesting that they encode less discriminative toxicity signal. Middle layers exhibit moderate initial losses and more mixed dynamics. Late layers show the strongest dynamics, typically starting with the highest losses and changing the most during training, consistent with the stronger toxicity representations suggested by our probe analysis in Section 6.1.

These broad trends hold across models, but the trajectories are model-specific. Mistral-7B (Figure 5a) shows the most consistent decrease across layers. In Olmo3-7B (Figure 5d), late layers increase sharply, but other layers decrease. In Llama3.1-8B (Figure 5b), earlier layers increase, but mid and late layers decrease. Qwen2.5-7B (Figure 5c) exhibits the strongest divergence, with some late layers reaching loss values above 80, indicating that here the representations resist orthogonalization more strongly, possibly due to architectural differences in how toxicity is distributed. Overall, these results suggest that the geometry of toxicity is architecture-dependent and that the best layer selection strategy may vary across models.

6.5 Layer Ablation Study for Toxicity

We analyze both the *number* and the *identity* of the updated layers using Llama3.1-8B, the model where our method performs best. Figure 7 in the Appendix shows that increasing the number of updated layers generally improves unlearning, although not monotonically: one- and two-layer interventions stay near the baseline, while groups of three or more layers yield larger reductions on average. Variation across layer groups of the same size further shows that *which* layers are updated matters nearly as much as *how many*. This is consistent with prior work showing that changes in transformer representations across depths are not monotonic (Liu et al., 2019, 2024b).

Figure 6, plotting toxicity by accuracy in MMLU for different combinations of unlearning layers, shows that layer identity strongly affects safety-utility tradeoff and unlearning. Single-layer interventions, and many two-layer groups, remain in the high-toxicity region, whereas several larger groups reduce toxicity substantially, especially those involving middle and later layers. The large variation among groups of the same size further suggests non-additive layer interactions. Overall, results suggest that effective toxicity unlearning benefits from coordinated updates across multiple layers, with middle-to-late layers serving as especially useful targets. Further, there is room for large reduction in toxicity with effective choice of layers with little change in general knowledge capabilities.

7 Conclusion

We argued that unlearning in LLMs should be treated as a goal-dependent problem rather than as a single generic intervention. Motivated by mechanistic differences between factual knowledge and toxicity, we introduced a cosine-based, meta-learned variant of RMU for dangerous-knowledge unlearning and then showed that this same approach does not transfer well to toxicity. For toxicity, we instead proposed a multi-layer objective based on layer-specific probe directions.

Across four open-source 7-8B models, our methods performed strongly for both dangerous knowledge and toxicity. Our proposed S-unlearning summary metric makes it easier to compare methods, and to capture the tradeoffs in unlearning. We presented analyses showing that toxicity directions vary across layers and that effective toxicity unlearning benefits from coordinated updates across multiple depths. Finally, our comparison of unlearning for two types of language function that have distinct mechanisms within LLMs suggests that unlearning should be approached as a family of methods that require customization to specific language functions.

Limitations

We focus on two types of unlearning goals in this work, but there are others as well. Further, we focus only on 4 small LLMs because of a limited computational budget.

References

Berk Atil, Namrata Sureddy, and Rebecca J. Passonneau. 2026. Something just like trust : Toxicity recognition of span and target. *Preprint*, arXiv:2506.02326.

Anne Auger, Johannes Bader, Dimo Brockhoff, and Eckart Zitzler. 2012. Hypervolume-based multiobjective optimization: Theoretical foundations and practical implications. *Theoretical Computer Science*, 425:75–103.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. 2021. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 141–159. IEEE.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Yinzhi Cao and Junfeng Yang. 2015. Towards making systems forget with machine unlearning. In *2015 IEEE Symposium on Security and Privacy*, pages 463–480. IEEE.

Huu-Tien Dang, Thanh-Tung Hoang, Le-Minh Nguyen, and Naoya Inoue. 2025. Improving the robustness of representation misdirection for large language model unlearning. *arXiv preprint arXiv:2501.19202*.

Huu-Tien Dang, Trung-Tin Pham, Thanh-Tung Hoang, and Naoya Inoue. 2024. On effects of steering latent representation for large language model unlearning. *arXiv preprint arXiv:2408.06223*.

Ahmed Deeb and François Roger. 2024. Do unlearning methods remove information from language model weights? *arXiv preprint arXiv:2410.08827*.

Hongzhe Du, Weikai Li, Min Cai, Karim Saraipour, Zimin Zhang, Himabindu Lakkaraju, Yizhou Sun, and Shichang Zhang. 2025. How post-training reshapes llms: A mechanistic view on knowledge, truthfulness, refusal, and confidence. In *Conference on Language Modeling (COLM)*.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. *arXiv preprint arXiv:2009.11462*.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12216–12235.

Aaron Grattafiori, Abhimanyu Dubey, ..., and Zhiyu Ma. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 2022. Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. *arXiv preprint arXiv:2203.09509*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Preprint*, arXiv:2009.03300.

Shengyuan Hu, Yiwei Fu, Zhiwei Steven Wu, and Virginia Smith. 2024. Jogging the memory of unlearned llms through targeted relearning attacks. *arXiv preprint arXiv:2406.13356*.

Dang Huu-Tien, Trung-Tin Pham, Hoang Thanh-Tung, and Naoya Inoue. 2024. On effects of steering latent representation for large language model unlearning. *arXiv preprint arXiv:2408.06223*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Swanand Kadhe, Farhan Ahmed, Dennis Wei, Nathalie Baracaldo, and Inkit Padhi. 2024. Split, unlearn, merge: Leveraging data attributes for more effective unlearning in LLMs. In *ICML 2024 Workshop on Foundation Models in the Wild*.

Andrew Lee, Xiaoyan Bai, Itamar Pres, Martin Wattenberg, Jonathan K. Kummerfeld, and Rada Mihalcea. 2024. A mechanistic understanding of alignment algorithms: A case study on dpo and toxicity. In *Proceedings of the 41st International Conference on Machine Learning*, pages 26361–26378. PMLR.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, et al. 2024. The wmdp benchmark: Measuring and reducing malicious use with unlearning. In *Proceedings of the 41st International Conference on Machine Learning*, pages 28525–28550. PMLR.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Yuguang Yao, et al.

2024a. Rethinking machine unlearning for large language models. *arXiv preprint arXiv:2402.08787*.

Zhu Liu, Cunliang Kong, Ying Liu, and Maosong Sun. 2024b. Fantastic semantics and where to find them: Investigating which layers of generative LLMs reflect lexical semantics. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14551–14558, Bangkok, Thailand. Association for Computational Linguistics.

Varvara Logacheva, Daryna Dementieva, Sergey Ustyantsev, Daniil Moskovskiy, David Dale, Irina Krotova, Nikita Semenov, and Alexander Panchenko. 2022. ParaDetox: Detoxification with parallel data. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6804–6818, Dublin, Ireland. Association for Computational Linguistics.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. 2024. Tofu: A task of fictitious unlearning for llms. *arXiv preprint arXiv:2401.06121*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *arXiv preprint arXiv:2202.05262*.

Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2025. Olmo 3. *Preprint*, arXiv:2512.13961.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, et al. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256.

Yanwu Xu, Mingming Gong, Tongliang Liu, Kayhan Batmanghelich, and Chaohui Wang. 2018. Robust angular local descriptor learning. In *Asian Conference on Computer Vision*, pages 420–435. Springer.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. 2024. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2404.05868*.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. 2023. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.

A Figures

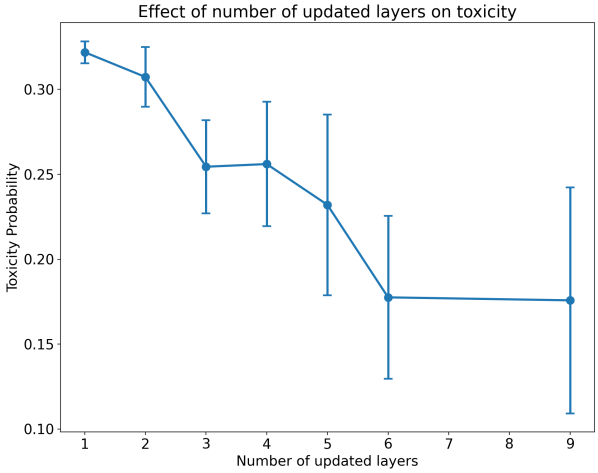


Figure 7: The effect of number of layers on toxicity unlearning

B Hyperparameters and Computing Infrastructure

We have finetuned the models for two epochs on up to 4 Nvidia RTX A6000. Each run took about an hour. For the reinforcement learning, we used a learning rate of 1e-2, and for the main finetuning, we used 5e-5. We used Adam optimizer for all experiments.