Contrastive Representation Shaping for LLM Unlearning

Haoran Tang
Department of Computer Science
Purdue University
thr@purdue.edu

Rajiv Khanna
Department of Computer Science
Purdue University
rajivak@purdue.edu

Abstract

Most LLM unlearning methods aim to approximate retrain-from-scratch behaviors with minimal distribution shift, often via alignment-style objectives defined in the prediction space. While effective at reducing forgotten content generation, such approaches may act as suppression: forgotten concepts can persist in representations and remain entangled with retained knowledge. We introduce CLReg, a contrastive representation regularizer that identifies forget features while pushing them away from retain features, reducing forget–retain interference while empirically preserving the scale and shape of retain features. As light motivation for the mechanism, we provide a one-step analysis showing that CLReg decreases a simple entanglement proxy in the embedding space. Across unlearning benchmarks and LLMs of different sizes, CLReg decreases forget-retain representation entanglement to enhance mainstream unlearning methods without extra privacy risks, inspiring future unlearning work to remove forget concepts via representation shaping. Code is available at `https://github.com/HaoranTang/CLReg`.

1 Introduction

Removing the influence of specific training data after deployment—*machine unlearning*—is increasingly important for privacy compliance and model maintenance. Large language models (LLMs) sharpen this need: they memorize and regenerate verbatim sequences from training corpora, so objectionable or proprietary content must be deletable on demand. Retraining from scratch on retained data is the gold-standard, but its cost is prohibitive for modern LLMs, motivating a surge of *approximate unlearning* methods that aim to mimic a retrained model at a fraction of the cost.

Early methods range from fine-tuning on the retain set and gradient ascent on the forget set to student–teacher distillation and saliency-based weight masking [17, 6]. While effective on small models, they often degrade utility and require careful tuning. Recent investigations show that unlearning becomes harder as forget and retain distributions grow more *entangled* or as forget examples are heavily memorized [37], so that separating these representations becomes central to selective forgetting.

Unlearning in LLMs is particularly challenging: naively maximizing loss on the forget set leads to instability and catastrophic degradation of general capabilities [36, 35]. Recent alignment-based methods dominate the literature—Negative Preference Optimization (NPO) [36] reweights the forgetting objective to discourage forgotten content while preserving utility; SimNPO [7] simplifies it to remove reference-model bias; others use self-distillation with adjusted logits [3] or primal–dual constrained entropic unlearning [5]. Despite their differences, these methods share a single philosophy: align the unlearned model's *prediction distribution* with the retrained model's, treating any deviation in representation space as undesirable. They succeed at suppressing forgotten outputs, but suppression need not remove the underlying knowledge: forgotten concepts can persist in the representation space, entangled with retained ones. Indeed, a relearning attack of only 32 fine-tuning steps closes over half of the gap back to the fine-tuned model for SimNPO (Sec. 4.5).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

An emerging view is that *representation shaping* can address this limitation. Empirical evidence shows that unlearning difficulty correlates with the degree of forget–retain entanglement [37, 29]: separating these clusters should make it easier to erase one without distorting the other. In representation learning, contrastive objectives are known to simultaneously *align* similar examples and *disperse* representations on the hypersphere [31], with simple augmentations and a cosine-similarity loss yielding tight positive clusters and uniform negatives [2, 8]. Building on this, contrastive unlearning has been proposed for small classifiers: Lee et al. [18] push forget embeddings away from their original class toward alternative regions, and Khalil et al. [16] align forget examples with retain semantics. These works suggest that explicitly shaping the feature space can make forgetting more targeted, but they operate in **supervised** classification settings with modest model sizes and rely on clean labels for positives and negatives. Whether similar benefits transfer to generative LLMs with self-supervision—where forget targets are instance-specific and the representation space is high-dimensional—has remained unclear.

We propose *contrastive representation regularization* (**CLReg**). We treat the *forget concept* as the coherent feature subspace spanned by embeddings of forget examples (Def. 3.1), rather than per-example memorization targets. Our key idea is to isolate this subspace and push it away from retain features, reducing entanglement while minimally perturbing the retain representation. We construct positive pairs for forget examples via lightweight augmentations (dropout masks and paraphrases), treat retain embeddings as negatives, and apply a contrastive loss that clusters forget anchors with their own augmentations and repels them from retain features. CLReg integrates with existing unlearning algorithms across families. As a light mechanistic motivation, we show that a single contrastive update strictly decreases anchor–negative similarity and a simple distributional entanglement proxy, giving intuition for why CLReg reduces forget–retain entanglement in practice. Empirically, across multiple benchmarks and LLMs, CLReg reduces entanglement and improves forget quality on top of state-of-the-art unlearning methods without introducing privacy risk. These findings challenge the prevailing belief that representation distributions must stay close to a retrained model: explicit representation shaping can *help* unlearning, and the clear forget/retain separation visualized in our results points to surgical latent-space interventions as a promising future direction.

Our contributions can be summarized as follows:

Rethinking representation shaping: Our study effectively shows that regularizing forget concepts in the representation space will not deviate the goal of unlearning or lead to collapse. Instead, our CLReg effectively separates forget features with minimal shifts on retain features, and improves unlearning performance with little privacy concerns.

Empirical analysis on entanglement with light mechanistic motivation: We formalize the forget concept as a feature subspace (Def. 3.1) and provide a simple one-step analysis in the embedding space that explains *why* CLReg reduces forget–retain entanglement; we present this as motivation and intuition rather than a standalone theoretical contribution. Our empirical evidence is where the entanglement claim is substantiated: quantitative entanglement metrics (variance-based entanglement, MK-MMD, W_2) and UMAP visualizations consistently show that CLReg pushes away forget features while keeping retain features close to their original scale and shape. Beyond observing this effect, we intervene on it: shaping representations with no forgetting objective at all improves the subsequent unlearning of an unmodified base method, and CLReg makes forgotten knowledge harder to recover under a relearning attack.

CLReg as a plug-and-play regularizer: We propose CLReg as a plug-in regularizer compatible with a wide range of unlearning objectives. We adapt contrastive construction to the generative, self-supervised LLM unlearning regime – where clean class labels for positives/negatives are unavailable – through dropout and paraphrase augmentations on forget anchors, with optional DPO-style, symmetric, or InfoNCE variants.

Empirical validation: Across TOFU, MUSE and WMDP on four LLMs (Llama-3-8B/3B, Llama-2-7B, Zephyr-7B-β), CLReg improves the `UnlearningScore` of all five mainstream unlearning methods (GradDiff, UNDIAL, NPO, SimNPO, PDU) in all 25 method–benchmark pairs. SimNPO+CL achieves the best `UnlearningScore` on every benchmark, beating the strongest CLReg-free baseline by $+0.039$ on TOFU-8B ($0.774 \rightarrow 0.813$, vs. 0.704 for the representation-shaping baseline RMU) and improving SimNPO by $+0.110$ on MUSE-News ($0.461 \rightarrow 0.572$); GradDiff+CL delivers the largest single-method gain, lifting `UnlearningScore` from 0.411 to 0.738 on MUSE-Books. CLReg cuts forget–retain entanglement by up to 88% (SimNPO on MUSE-News, $92.2 \rightarrow 11.1$) without

degrading retain-side utility, and pulls $|\texttt{PrivLeak}|$ closer to zero in most settings (e.g., $62.0 \to 3.7$ for SimNPO+CL on TOFU-3B). Ablations on contrastive variants, the number and selection of negatives, layer choice, and runtime confirm the gains are robust to design choices, and CLReg preserves general capability and sharply reduces utility erosion under sequential unlearning requests.

2 Related Work

Foundations in unlearning. Early *approximate* unlearning methods aim to erase designated forget examples without retraining. A common baseline fine-tunes on retain data and relies on catastrophic forgetting [9, 33]; more direct approaches perform *gradient ascent* on the forget set, which actively degrades memory but can harm utility. NegGrad+ balances the two by jointly maximizing forget loss and minimizing retain loss, with student–teacher training following the same scheme [17]. SalUn instead restricts updates to weights most responsible for forget-set predictions, limiting collateral damage [6]. Other paradigms include label remixing and second-order influence estimates [12, 15].

LLM unlearning. For LLMs, gradient-difference objectives (GradDiff) counteract the forget set while preserving retained behavior [22]. To improve stability, NPO [36] reframes forgetting as preference optimization to control dynamics; SimNPO [7] removes a reference-model bias that favors easy-to-forget instances. UnDIAL [3] replaces explicit loss maximization with a distillation-style objective to prevent collapse, and Entesari et al. [5] cast unlearning as a primal–dual constrained problem for better Pareto trade-offs. A parallel line studies losses and data strategies beyond the standard forget–retain split [35, 32, 19] (see Liu et al. [21] for a broader survey), but operates at the logit or loss level. Li et al. [20] introduce the WMDP benchmark together with RMU, which unlearns by *misdirecting* forget-layer activations toward a random direction—an early representation-level intervention that validates the premise but differs fundamentally from CLReg's *structured* contrastive separation. CLReg is complementary: it shapes the representation space rather than the output distribution and can be layered on top of logit-level objectives.

Contrastive learning for unlearning. Contrastive objectives offer a complementary path that *explicitly shapes representations* to reduce forget–retain entanglement. Lee et al. [18] push forget embeddings away from their original class clusters toward alternative regions, and CoUn [16] restructures the latent space so that forget examples align with retain-induced semantics. Both demonstrate that representation-space restructuring sharpens forgetting on *supervised*, small-to-medium classifiers, where class labels supply a natural contrastive signal. Extending this to LLM unlearning is non-trivial: forget targets are often instance- or concept-level without clean labels, the model is vastly larger, and the success criterion is *behavioral* equivalence to a retain-only retrained model rather than mere representation separation. CLReg fills this gap by isolating and pushing away forget features from retain features in LLM representations.

Theoretical analyses of contrastive learning. Arora et al. [1] derive downstream guarantees for contrastive learning under a latent-class assumption, and Wang and Isola [31] characterize InfoNCE through *alignment* of positive pairs and *uniformity* on the hypersphere; both analyze representation-level proxies rather than end-to-end task guarantees. Our mechanistic motivation (Sec. 3.3) follows this methodology, using a one-step decrease in anchor–negative similarity as a transparent proxy.

3 CLReg: Contrastive Regularization

3.1 Preliminaries

Let $\mathbf{W}_{\mathrm{FT}}$ denote the finetuned model, $\mathbf{W}_{\mathrm{RT}}$ the retrained (target) model trained from scratch on the retain set, and $\mathbf{W}_{\mathrm{UL}}$ the model undergoing unlearning. We have access to an original training set $\mathcal{S}$ drawn from distribution $\mathcal{D}$, which is later decomposed into a *retain* set $\mathcal{R}$ and a *forget* set $\mathcal{F}$ after finetuning, with $\mathcal{R} = \mathcal{S} \setminus \mathcal{F}$ and $|\mathcal{R}| > |\mathcal{F}|$. For any model $\mathbf{W}$, we denote by $h_{\mathbf{W}}(x) \in \mathbb{R}^{T \times d}$ the hidden-state matrix of $\mathbf{W}$ for input x with T tokens and d hidden dimension. We define $\mathrm{Pool}(h, m) = \sum_t m_t h_t / \sum_t m_t$ to be mean pooling of hidden states with attention mask m and token position index t. For brevity, we write $\zeta_{\mathbf{W}}(x) = \mathrm{norm}\big(\mathrm{Pool}(h_{\mathbf{W}}(x), m(x))\big) \in \mathbb{S}^{d-1}$ for the ℓ_2–normalized embedding (unit sphere) and use cosine similarity $s(u, v) = u^\top v$.

Definition 3.1 (Forget concept)**.** Given a forget set $\mathcal{F}$ and a model with parameters θ, we define the *forget concept* as the coherent feature subspace

$$\mathcal{C}_{\mathcal{F}}^{\theta} = \mathrm{span}\big\{\zeta_\theta(x) \;\big|\; x \in \mathcal{F}\big\} \subseteq \mathbb{R}^d,$$

i.e. the subspace spanned by the pooled, normalized embeddings of forget inputs in the latent space.

This definition makes the object of unlearning explicit: CLReg does not target the verbatim reproduction of individual sequences, but the *feature cluster* $\mathcal{C}_{\mathcal{F}}^{\theta}$ that carries the high-level knowledge associated with the forget set (e.g., author biographies in TOFU, book/news content in MUSE, hazardous domain expertise in WMDP). Correspondingly, our target is to reduce the entanglement between $\mathcal{C}_{\mathcal{F}}^{\theta}$ and the retain subspace $\mathcal{C}_{\mathcal{R}}^{\theta}$, so that downstream unlearning objectives can remove the forget-side features with less collateral damage to retain features. Per-example augmentations (dropout, paraphrase) serve as proxies that probe this subspace: samples from different augmentations should reside in $\mathcal{C}_{\mathcal{F}}^{\theta}$, whereas retain embeddings lie outside it and can be used as negatives.

3.2 Separating Forget Concepts

We draw inspiration from recent advances in contrastive representation learning that emphasize *alignment* (bringing positive pairs close) and *uniformity* (spreading all representations) [31]. For each forget example $x_f \in \mathcal{F}$ with paraphrase x_f^+ we construct a positive pair (z_f, z_f^+) via light augmentations:

$$z_f = \mathrm{Pool}(\mathrm{Dropout}(h_{\mathrm{UL}}(x_f), p), m(x_f)),$$
$$z_f^+ = \mathrm{Pool}(\mathrm{Dropout}(h_{\mathrm{UL}}(x_f^+), p'), m(x_f^+)).$$

Here $p, p' \sim \mathcal{N}(\mu, \sigma)$ are independently sampled dropout rates which help differentiate z_f from z_f^+ [8]. In practice we pick $\mathcal{N}(0.1, 0.05)$ and clamp p to $[0, 0.2]$ to add randomness as data augmentation. The paraphrase x_f^+ is precomputed once by a language model (Supp. A.4); if paraphrasing is unavailable we simply set $x_f^+ = x_f$, so that the two views differ only in their dropout masks. Each retain item $x_r \in \mathcal{R}$ provides a negative $z_r^- = \mathrm{Pool}(h_{\mathrm{UL}}(x_r), m(x_r))$ (without dropout). A *DPO-style contrastive loss* [26, 34] encourages every positive pair to have higher similarity than each of its negative pairs:

$$\mathcal{L}_{\mathrm{CL}}^{\mathrm{dpo}} = -\frac{2\tau}{B_f B_r} \sum_{i=1}^{B_f} \sum_{j=1}^{B_r} \log \sigma \left(\frac{s(z_{f_i}, z_{f_i}^+) - s(z_{f_i}, z_{r_j}^-)}{\tau} \right), \tag{1}$$

where σ is the sigmoid, B_f and B_r are the numbers of forget anchors and retain negatives, and the temperature $\tau > 0$ controls hardness of negatives. Every anchor is contrasted with all B_r negatives, gathered across data-parallel workers; this gives 32 negatives per anchor in all our experiments (Supp. A.4). Alternatively, a standard *InfoNCE loss* $\mathcal{L}_{\mathrm{CL}}^{\mathrm{info}}$ [25] can be used: the logits concatenate the positive similarity and all cross-retain similarities, and a cross-entropy loss identifies the positive as the correct one. Both forms can be symmetrized by swapping anchor/negative roles (retain vs. forget).

Combined objective. Our CLReg acts as a regularizer on top of any base unlearning algorithm. Suppose $\mathcal{L}_{\text{forget}}$ and $\mathcal{L}_{\text{retain}}$ denote the forget and retain losses (e.g. SimNPO for $\mathcal{L}_{\text{forget}}$ and cross-entropy for $\mathcal{L}_{\text{retain}}$). We minimize

$$\mathcal{L} = \alpha \mathcal{L}_{\text{retain}} + \gamma \mathcal{L}_{\text{forget}} + \lambda \mathcal{L}_{\mathrm{CL}}, \tag{2}$$

with hyperparameters $\alpha, \gamma, \lambda \geq 0$. The CL term shapes the representation space while $\mathcal{L}_{\text{forget}}$ unlearns $\mathcal{F}$ and $\mathcal{L}_{\text{retain}}$ preserves $\mathcal{R}$. On current benchmarks the forget set consists of high-level semantic knowledge that prior work localizes to *later* transformer layers [14], so we apply CLReg there; Sec. 4.6 confirms that regularizing earlier layers degrades utility more than it helps forgetting. For concepts localized earlier (e.g., stylistic or syntactic patterns), concept-dependent layer selection is a natural direction for future work.

3.3 A Mechanistic View through Alignment, Uniformity, and Latent-Class Separation

We recast two established lenses on contrastive learning to the unlearning setting, and work in embedding space where "entanglement" is measured and where both lenses operate. Wang and Isola [31] decompose the contrastive loss into *alignment* (pulling positives together) and *uniformity* (spreading representations on the hypersphere); in CLReg, positives are augmented views of the same forget example, while the "spread" is restricted to forget-vs-retain pairs, yielding a *directional uniformity* that pushes forget mass away from retain mass rather than spreading the entire batch.

Arora et al. [1] analyze contrastive learning through unobserved latent classes and show that the population loss is minimized when class-conditional means are separated; identifying $\mathcal{C}_{\mathcal{F}}^{\theta}$ with one latent class and the retain distribution with the other makes our entanglement reduction a class-separation objective in their sense. Propositions 3.3–3.5 make this concrete at the level of a single update: the embedding-space step is the directional-uniformity component of Wang & Isola, applied across the (forget, retain) class pair of Arora et al.

Definition 3.2. (Distributional separation and entanglement). Let $\zeta_\theta(x) \in \mathbb{R}^d$ denote the embedding of an input x under parameters θ (pooled hidden states and normalized). Let

$$\mathcal{P}_{\mathcal{F}}^{\theta} = \mathrm{Law}\big(\zeta_\theta(x) \mid x \in \mathcal{F}\big), \mathcal{P}_{\mathcal{R}}^{\theta} = \mathrm{Law}\big(\zeta_\theta(x) \mid x \in \mathcal{R}\big),$$

denote the induced distributions of embeddings from forget and retain sets. A *separation measure* between $(\mathcal{P}_{\mathcal{F}}^{\theta}, \mathcal{P}_{\mathcal{R}}^{\theta})$ is any probability metric D such that $D(\mathcal{P}_{\mathcal{F}}^{\theta}, \mathcal{P}_{\mathcal{R}}^{\theta}) = 0$ if and only if $\mathcal{P}_{\mathcal{F}}^{\theta} = \mathcal{P}_{\mathcal{R}}^{\theta}$. High separation $D(\mathcal{P}_{\mathcal{F}}^{\theta}, \mathcal{P}_{\mathcal{R}}^{\theta})$ corresponds to low entanglement, whereas low separation (or high overlap) indicates that the representations of $\mathcal{F}$ and $\mathcal{R}$ are intertwined.

Proposition 3.3. ***(Anchor update for DPO-CL)**. Consider a single $\mathcal{L}_{CL}^{dpo}$ term with i-th forget sample and j-th retain sample*

$$\ell_{ij}(\theta) = -2\tau \log \sigma\Big(\tfrac{m_{ij}}{\tau}\Big), m_{ij} = s(a_i, p_i) - s(a_i, n_j),$$

where $a_i = \zeta_\theta(x_{f_i})$ is the anchor (forget embedding), $p_i = \zeta_\theta(x_{f_i}^{+})$ is its positive (paraphrased or dropout-augmented), $n_j = \zeta_\theta(x_{r_j})$ is a negative (retain embedding), and $s(u, v) = u^\top v$. Suppose $a_i, p_i, n_j \in \mathbb{S}^{d-1}$. The gradient of ℓ_{ij} with respect to a_i is

$$\nabla_{a_i} \ell_{ij} = -2\, \sigma\Big(-\tfrac{m_{ij}}{\tau}\Big)\, (p_i - n_j).$$

In particular, a gradient descent step $a_i' = a_i - \eta\, \nabla_{a_i} \ell_{ij}$ (with small $\eta > 0$) moves a_i toward *p_i and* away from *n_j. If $p_i \neq n_j$ and $\eta > 0$, then ${a_i'}^\top n_j \;<\; a_i^\top n_j$, so the anchor–negative cosine similarity strictly decreases.*

The update direction $p_i - n_j$ splits into $+p_i$ (an alignment pull toward the positive view) and $-n_j$ (a repulsion from a retain negative): the per-sample analogues of the alignment and uniformity gradients in Wang and Isola [31], restricted to one cross-distribution pair from $(\mathcal{F}, \mathcal{R})$.

Multiple negatives. Summed over all B_r negatives in Eq. (1), the anchor is repelled from the *centroid* $\bar{n}_w$ of the negatives weighted by $w_{ij} = \sigma(-m_{ij}/\tau)$, which up-weights hard negatives automatically. Its similarity to $\bar{n}_w$ strictly decreases iff $\cos(p_i, \bar{n}_w) < \|\bar{n}_w\|$ (Prop. A.1, Supp. A.1), which reduces to $p_i \neq n_j$ when $B_r = 1$. This condition is not automatic; measured during training, it holds for 94% of anchor updates (16,660 of 17,728), an empirical regularity rather than a guarantee.

Corollary 3.4. ***(One-step decrease of cross-similarity)**. Under the same setup as Proposition 3.3, consider the expected cross-similarity (linear kernel overlap)*

$$C_{\mathrm{lin}}(\theta) = \mathbb{E}\big[a^\top n\big],$$

where $a = \zeta_\theta(x_f)$ and $n = \zeta_\theta(x_r)$ for independent $x_f \in \mathcal{F}$ and $x_r \in \mathcal{R}$. If a single gradient step on ℓ_{ij} updates a_i to a_i' while leaving p_i and all n_j fixed, then with updated parameters θ',

$$C_{\mathrm{lin}}(\theta') \;\leq\; C_{\mathrm{lin}}(\theta),$$

with strict inequality if $p_i \neq n_j$ for any updated pair. Hence, the DPO-CL update strictly reduces the expected anchor–negative similarity.

Proposition 3.5. ***(Increase of separation under CLReg)**. Let D be any separation measure between distributions satisfying the following:*

- *There exists a continuous cost function $c : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ such that $D(\mathcal{P}, \mathcal{Q})$ is a non-decreasing function of the expected cross-cost $\mathbb{E}_{u\sim\mathcal{P},\, v\sim\mathcal{Q}}[\, c(u, v)\,]$; that is, $c(u, v_1) \leq c(u, v_2)$ implies $D(P, \delta_{v_1}) \leq D(P, \delta_{v_2})$, where δ_v denotes the Dirac measure at v.*
- *The cost c is strictly increasing with respect to the anchor–negative similarity: if $s(u_1, v) < s(u_2, v)$ then $c(u_1, v) > c(u_2, v)$.*

Then a gradient descent step on $\mathcal{L}_{CL}^{dpo}$ reduces $\mathbb{E}[s(a,n)]$ and thereby increases $D(\mathcal{P}_{\mathcal{F}}^{\theta}, \mathcal{P}_{\mathcal{R}}^{\theta})$*:*

$$D\big(\mathcal{P}_{\mathcal{F}}^{\theta'}, \mathcal{P}_{\mathcal{R}}^{\theta'}\big) \;\geq\; D\big(\mathcal{P}_{\mathcal{F}}^{\theta}, \mathcal{P}_{\mathcal{R}}^{\theta}\big),$$

with strict increase when at least one updated anchor has $p_i \neq n_j$.

See Supp. A.1 for all proofs. The cross-similarity $C_{\text{lin}}(\theta) = \mathbb{E}[a^\top n]$ is the inter-class kernel overlap whose contraction underlies the generalization bound of Arora et al. [1], so Prop. 3.5 shows that one CLReg step moves the same quantity their analysis tracks, in the same direction, and explains why this should reduce embedding-level entanglement between $\mathcal{C}_{\mathcal{F}}^{\theta}$ and $\mathcal{C}_{\mathcal{R}}^{\theta}$ [37]. The empirical reductions in Tab. 3–4 and the UMAP visualizations in Fig. 1, not the propositions, are where the claim is ultimately established.

4 Experiment

Table 1: TOFU unlearning experiments for Llama-3-8B and 3B.

(a) TOFU unlearning experiment results for Llama-3-8B. For the last four columns, bold indicates the best in-column, and green shades indicate improvement. CLReg consistently improves overall unlearning score and model utility. SimNPO+CL achieves the best performance, and GradDiff+CL achieves the largest improvement. In most cases, CLReg brings privacy leak closer to 0, achieving better balance.

TOFU Llama-3-8B	Extraction Strength↓	Forget QA Prob↓	Forget QA ROUGE↓	Forget Quality↑	Forget Score↑	Model Utility↑	**Unlearning Score**↑	Privacy Leak→0
GradDiff	0.06646	0.00498	0.28313	2.63E-10	0.86826	0.53938	0.66540	51.03416
UNDIAL	0.06123	0.20657	0.30030	4.64E-12	0.80476	0.62101	0.70104	-78.91192
NPO	0.12513	0.30622	0.39238	1.37E-07	0.83696	0.66680	0.74225	-69.79255
SimNPO	0.12980	0.10784	0.39639	4.46E-06	0.91079	0.67261	0.77378	-48.65591
PDU	0.06233	0.04992	0.19536	4.36E-09	0.89069	0.57892	0.70173	45.72288
RMU	0.03361	0.00278	0.03627	5.7E-15	0.75790	0.65646	0.70354	63.85000
GradDiff+CL	0.14964	0.32726	0.39953	9.91E-11	0.77768	0.67359	0.72191	-48.86873
UNDIAL+CL	0.06201	0.22942	0.31487	9.34E-13	0.78446	0.66806	0.72159	-76.47375
NPO+CL	0.11762	0.28935	0.38650	1.30E-05	0.87003	0.68487	0.76643	-56.01587
SimNPO+CL	0.04593	0.01815	0.11235	0.00229	**0.97182**	**0.69815**	**0.81256**	55.03337
PDU+CL	0.06375	0.07160	0.22932	6.78E-07	0.92574	0.58539	0.71724	**29.14540**

(b) TOFU unlearning experiment results for Llama-3-3B. For the last four columns, bold indicates the best in-column, and green shades indicate improvement. CLReg consistently improves overall unlearning scores and model utility. SimNPO+CL achieves the best performance. In most cases, CLReg brings privacy leak closer to 0. Impactfully, SimNPO+CL reduces the original absolute privacy leak value 61.994 to 3.725.

TOFU Llama-3-3B	Extraction Strength↓	Forget QA Prob↓	Forget QA ROUGE↓	Forget Quality↑	Forget Score↑	Model Utility↑	**Unlearning Score**↑	Privacy Leak→0
GradDiff	0.08236	0.03150	0.34287	7.83E-12	0.80783	0.59254	0.68364	-2.41669
UNDIAL	0.06051	0.18277	0.23129	1.73E-15	0.67538	0.56827	0.61721	-86.86811
NPO	0.12257	0.35523	0.41255	2.77E-09	0.77438	0.61585	0.68607	-75.13616
SimNPO	0.11093	0.11723	0.38107	6.83E-09	0.85847	0.61040	0.71348	-61.99405
PDU	0.07656	0.17840	0.27305	6.39E-06	0.90204	0.50306	0.64590	-42.08622
RMU	0.03251	0.00013	0.10726	5.42E-13	0.77818	0.62875	0.69553	63.14000
GradDiff+CL	0.06042	0.10162	0.31587	0.52341	**0.98902**	0.63042	0.77001	6.36311
UNDIAL+CL	0.06137	0.20976	0.26059	5.14E-16	0.65019	0.61893	0.63417	-84.90512
NPO+CL	0.09906	0.28907	0.38257	6.78E-07	0.84701	0.63707	0.72719	-64.04083
SimNPO+CL	0.06353	0.10866	0.32609	0.00383	0.96209	**0.66531**	**0.78664**	**-3.72569**
PDU+CL	0.06178	0.08528	0.26582	2.57E-05	0.93845	0.52995	0.67738	18.51810

4.1 Unlearning Setup

We conduct unlearning experiments on TOFU [22] and MUSE [27] benchmarks (overview and concrete examples in Supp. A.2), where on TOFU we unlearn LLMs of different sizes (Llama-3.1-8B and Llama-3.2-3B [11]), and on MUSE we experiment unlearning both Books and News datasets with Llama-2-7B [30]. Given the finetuned model $\mathbf{W}_{\text{FT}}$, we unlearn it for 10 epochs with lr$= 10^{-5}$ to obtain $\mathbf{W}_{\text{UL}}$. We unlearn with GradDiff, NPO, SimNPO, UnDIAL, PDU [36, 7, 3, 5]. We first tune method-specific hyper-parameters and γ for each method for optimal performance as baselines, then tune CLReg-specific parameters when being applied: $\tau \times$ [symmetric, non-symmetric] $\times$ $[\mathcal{L}_{\text{CL}}^{\text{dpo}}, \mathcal{L}_{\text{CL}}^{\text{info}}]$. We fix $\alpha, \lambda = 1$ to ease hyper-parameter search. Paraphrases are generated once offline with Llama-3.1-8B-Instruct (five per question, question only), and we train with an effective batch size of 128 (TOFU) / 256 (MUSE) and 32 negatives per anchor. See Supp. A.3 and A.4 for detailed settings.[1]

[1]The analyses in Sec. 4.5 and Supp. A.8 re-run every arm, baselines included, from the publicly released benchmark checkpoints, so absolute scores differ slightly from Tabs. 1–2 (e.g., SimNPO on Llama-3-3B: 0.688 vs. 0.713), while every comparison within those analyses is matched.

Table 2: MUSE unlearning experiments across datasets (Books and News).

(a) MUSE-Books Unlearning experiment results. For the last four columns, bold indicates the best in-column, and green shades indicate improvement. CLReg consistently improves overall unlearning scores and forget scores, and can outperform retrained models in forgetting. SimNPO+CL achieves the best performance, and GradDiff+CL achieves the largest improvement. Similar to TOFU results, CLReg does not degrade privacy leak or undermines unlearning balance, and bring it closer to 0 for many cases.

MUSE-Books Llama-2-7B	Exact Memoriation↓	Extraction Strength↓	Forget knowmem ROUGE↓	Forget verbmem ROUGE↓	Forget Score↑	Retain knowmem ROUGE↑	**Unlearning Score↑**	Privacy Leak→0
GradDiff	0.93627	0.15484	0.32808	0.35574	0.31345	0.59876	0.41148	-60.46598
UNDIAL	0.94730	0.18230	0.38053	0.42416	0.25306	**0.63058**	0.36117	-53.93861
NPO	0.92421	0.13040	0.20335	0.30689	0.36774	0.52260	0.43170	-51.25740
SimNPO	0.20746	0.00913	0.33610	0.00316	0.94209	0.60877	0.73961	-30.65828
PDU	0.00968	0.00794	0.22401	8.85E-05	**1.0**	0.44479	0.61572	-35.96524
RMU	0.62484	0.02341	0.3716	0.15719	0.77996	0.61749	0.68928	-55.42000
GradDiff+CL	0.18952	0.00865	0.18392	0.00540	**1.0**	0.58451	0.73778	-57.32249
UNDIAL+CL	0.90897	0.09333	0.33523	0.27550	0.41405	0.57313	0.48077	-57.56287
NPO+CL	0.91302	0.11452	0.21475	0.26978	0.41021	0.53345	0.46378	-52.55178
SimNPO+CL	0.01206	0.00794	0.27470	0.0	**1.0**	0.59918	**0.74936**	**-21.11686**
PDU+CL	0.00944	0.00794	0.23983	3.37E-04	**1.0**	0.47345	0.64264	-34.26405

(b) MUSE-News Unlearning experiment results. For the last four columns, bold indicates the best in-column, and green shades indicate improvement. CLReg consistently improves overall unlearning scores. SimNPO+CL achieves the best performance with slight over-unlearning despite improving |`PrivLeak`|.

MUSE-News Llama-2-7B	Exact Memoriation↓	Extraction Strength↓	Forget knowmem ROUGE↓	Forget verbmem ROUGE↓	Forget Score↑	Retain knowmem ROUGE↑	**Unlearning Score↑**	Privacy Leak→0
GradDiff	0.83810	0.06833	0.53086	0.24138	0.48643	0.46619	0.47609	**-87.84635**
UNDIAL	0.75079	0.03571	0.28436	0.20273	0.84017	0.36324	0.50720	-99.49622
NPO	0.78690	0.04421	0.50151	0.20549	0.62377	0.45399	0.52550	-92.69521
SimNPO	0.84802	0.07198	0.54494	0.24493	0.44588	**0.47777**	0.46128	-90.55416
PDU	0.77754	0.02984	0.41276	0.20177	0.74393	0.35294	0.47875	-99.72712
RMU	0.81357	0.04881	0.51091	0.24973	0.55910	0.46187	0.50586	-99.33000
GradDiff+CL	0.34405	0.01198	0.42668	0.03450	**0.89807**	0.41004	0.56302	99.68514
UNDIAL+CL	0.75532	0.03278	0.27290	0.21257	0.83035	0.36937	0.51130	-99.47523
NPO+CL	0.78722	0.04500	0.49930	0.20703	0.62591	0.46569	0.53404	-92.54828
SimNPO+CL	0.38992	0.01413	0.44140	0.04043	0.87730	0.42403	**0.57173**	88.65659
PDU+CL	0.78397	0.02786	0.39836	0.21609	0.73958	0.37476	0.49745	-99.72712

4.2 Evaluation

In addition to adopting evaluation metrics from TOFU and MUSE, we propose **Forget Score**↑ that maps each forget metric of $\mathbf{W}_{\text{UL}}$ as a progress measure from $\mathbf{W}_{\text{FT}}$ to $\mathbf{W}_{\text{RT}}$: the more unlearned the $\mathbf{W}_{\text{UL}}$ is, the more similar performance it is expected to share with $\mathbf{W}_{\text{RT}}$ on $\mathcal{F}$. For each forget metric m, first convert it to a progress measure:

$$\text{Prog}(m, \mathcal{F}) = \frac{|m(f_{\text{UL}}, \mathcal{F}) - m(f_{\text{FT}}, \mathcal{F})|}{|m(f_{\text{RT}}, \mathcal{F}) - m(f_{\text{FT}}, \mathcal{F})|}, \tag{3}$$

which will be clipped at 1 when outperforming $\mathbf{W}_{\text{RT}}$. Note that the `ForgetQuality` [22] spans many orders of magnitude, and is usually close to zero, we take $\log(\cdot)$ on it to better address the small differences. Given K evaluation metrics to measure different aspects of forgetting, we compute an overall `ForgetScore` analogous to `ModelUtility` as the harmonic mean:

$$\text{ForgetScore} = K \left(\sum_{k=1}^{K} \frac{1}{\text{Prog}(m_k, \mathcal{F})} \right)^{-1}. \tag{4}$$

Likewise, we can measure an overall **Unlearning Score**↑ as the harmonic mean of `ForgetQuality` and `ModelUtility` (or `RetainKnowmemROUGE` on MUSE), emphasizing on balancing forgetting and retaining. Despite many of the metrics are privacy/leakage-aware already (e.g., `ForgetQuality`) [4], we also report `PrivLeak`→ 0 dedicated to privacy leakage [27], where positive suggests over-unlearning and negative suggests under-unlearning and is encouraged to approach zero when $\mathbf{W}_{\text{UL}}$ is well-balanced. See Supp. A.5 for an overview of each evaluation metric.

4.3 Shaping Representation Improves Unlearning

Tab. 1 and Tab. 2 report detailed results on TOFU and MUSE. CLReg consistently improves `UnlearningScore` across LLM sizes and datasets, lifting `ForgetScore` in most cases while *also* improving `ModelUtility` for all methods on TOFU. CLReg can yield mild over-unlearning (positive `PrivLeak`) in a few cases but does not noticeably increase |`PrivLeak`|, and in most settings pulls

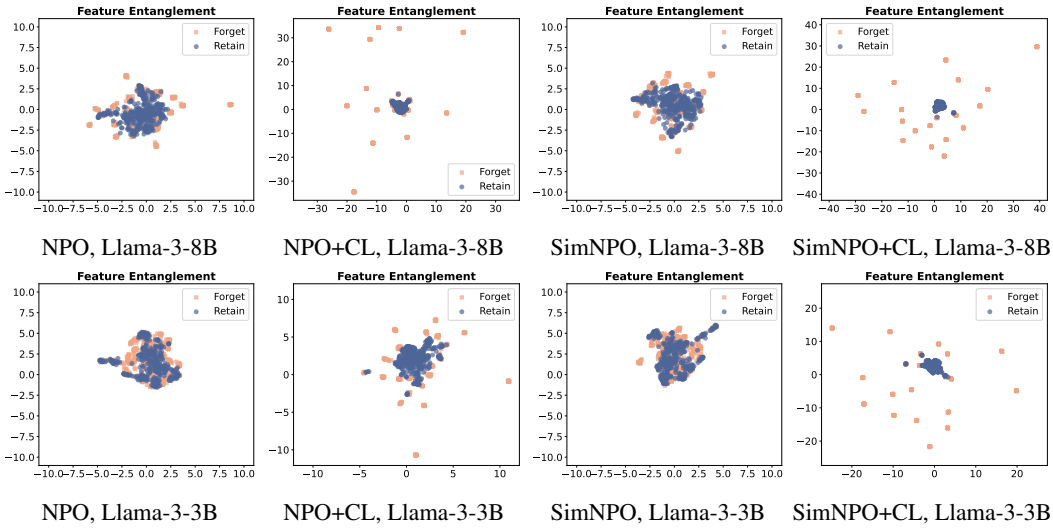

Figure 1: UMAP visualizations of NPO and SimNPO unlearning on TOFU benchmark, compared with CLReg variants. We observe that CLReg can effectively identify and separate forget features by pushing them away, while still maintaining the original scale and distributions of the retain features. **Please compare axis scales**: without CLReg, both forget and retain features lie within roughly $[-10, 10]$. With CLReg, the retain cluster remains in $[-10, 10]$ with essentially the same shape and density, while forget features are pushed to a much larger range (e.g., $[-30, 40]$ for SimNPO-8B).

`PrivLeak` closer to 0 – the desired balance per [27]. Overall, SimNPO+CL achieves the best performance across all settings, and CLReg most steadily strengthens SimNPO and NPO; we hypothesize their preference-learning objectives share more optimization goals with CLReg than the other methods. The gain is not tied to a particular contrastive design: a 2×2 ablation on contrastive-loss form ($\mathcal{L}_{\text{CL}}^{\text{dpo}}$ vs. $\mathcal{L}_{\text{CL}}^{\text{info}}$) and symmetry shows $14/16$ configurations beat the CLReg-free baseline (Supp. A.7). Nor does it need many or carefully chosen negatives: one negative gives 0.803 versus 0.807 for 32, and hard, random, or easy negatives span 0.012 (Supp. A.8.1); CLReg also helps on Llama-3.2-1B (Supp. A.8.6). The largest regression, the `PrivLeak` of GradDiff+CL on MUSE-News, reflects the fixed $\lambda = 1$ rather than the method: lowering λ brings it through zero (Supp. A.8.7). CLReg adds only one extra forward pass per iteration and no additional trainable parameters, yielding a 34–49% wall-clock overhead on TOFU (Supp. A.9).

Comparison with RMU. Unlike the base methods above, RMU [20] operates on hidden representations by misdirecting forget-layer activations toward a *random* direction. We add RMU as an additional baseline row in Tables 1a–1b and 2a–2b. On TOFU, RMU yields `UnlearningScore` 0.70 (8B) / 0.70 (3B), while SimNPO+CL reaches 0.81 / 0.79; on MUSE-Books, RMU reaches 0.69 versus SimNPO+CL 0.75; on MUSE-News, RMU reaches 0.51 versus SimNPO+CL 0.57. Most CLReg-augmented methods therefore outperform RMU, which is consistent with our view that *structured* contrastive separation (forget away from retain) is more effective than *random* misdirection for retain-side preservation. We do not report RMU+CL: the two operate on representations with potentially conflicting objectives (random misdirection vs. structured contrastive separation), and we leave a careful study to future work.

WMDP. We further evaluate CLReg on WMDP-cyber [20] with Zephyr-7B-β, where a true retrained reference is infeasible (we use the random-guess MCQ accuracy 0.25 as a conservative lower-bound proxy for $m(\mathbf{W}_{\text{RT}}, \mathcal{F})$). CLReg improves `UnlearningScore` for *all five* base methods; SimNPO+CL reaches 0.674 and essentially matches RMU's published 0.677, and NPO+CL lifts NPO from 0.551 to 0.584 once α is rebalanced. Full setup, table, and per-method discussion are in Supp. A.6.

4.4 Disentangled Representation for Easier Unlearning

We next examine the feature layer where CLReg is applied. Guided by the one-step analysis of Sec. 3.3, and since prior work finds that unlearning becomes harder as forget–retain entanglement grows [37, 29], we provide quantitative and qualitative analysis of the feature space, comparing NPO, SimNPO with NPO+CL and SimNPO+CL to verify our claims. We implement variance-

Table 3: TOFU entanglement evaluation results. CLReg consistently reduces entanglement on all three metrics for NPO and SimNPO; the largest improvement is SimNPO+CL on Llama-3-8B ($20.24 \rightarrow 5.9$). †Re-run from the public checkpoint (Sec. 4.5), correcting a duplicated entry in the submitted version.

Llama-3-8B	NPO	+CL	SimNPO	+CL
Entanglement↓	22.30576	17.27656	20.24046	5.90206
MK-MMD↑	0.01424	0.02138	0.01537	0.07327
2-Wasserstein↑	0.10355	0.15881	0.10001	0.31250
Llama-3-3B	NPO	+CL	SimNPO†	+CL†
Entanglement↓	25.07525	18.02604	17.19105	5.62938
MK-MMD↑	0.01213	0.01820	0.01833	0.06930
2-Wasserstein↑	0.06348	0.10085	0.06768	0.23700

Table 4: MUSE entanglement evaluation results. CLReg consistently reduces entanglement on all three metrics for NPO and SimNPO; the largest improvement is SimNPO+CL on MUSE-News ($92.18 \rightarrow 11.09$).

MUSE-Books	NPO	+CL	SimNPO	+CL
Entanglement↓	0.36104	0.28346	0.03898	0.02058
MK-MMD↑	0.47111	0.57001	1.21513	1.27422
2-Wasserstein↑	0.17936	0.23721	0.62096	0.90730
MUSE-News	NPO	+CL	SimNPO	+CL
Entanglement↓	59.07131	58.66790	92.17580	11.08532
MK-MMD↑	0.00515	0.00520	0.00392	0.01775
2-Wasserstein↑	0.05856	0.05899	0.04923	0.11468

Table 5: Separation facilitates and stabilizes unlearning (TOFU, Llama-3.2-3B). (a) Stage 2 is an *unmodified* base method; the 15-epoch run matches the warm start's total compute. (b) Normalized recovery AUC under a relearning attack; SimNPO entries are mean ± sd over 3 seeds.

(a) Two-stage (`UnlearningScore`↑)	standard (10 ep)	compute-matched (15 ep)	warm-start → base
SimNPO	0.688	0.734	**0.771**
NPO	0.648	0.654	**0.704**

(b) Relearning (recovery AUC↓)	SimNPO	GradDiff	NPO
base	0.329 ± 0.021	0.383	**0.068**
+CL	$\mathbf{0.228 \pm 0.012}$	**0.130**	0.077

based entanglement from Goldblum et al. [10], Zhao et al. [37]: $\mathrm{E} = (\frac{1}{|\mathcal{R}|}\sum_{i\in\mathcal{R}}(\boldsymbol{\phi}_i - \boldsymbol{\mu}_\mathcal{R})^2 + \frac{1}{|\mathcal{F}|}\sum_{j\in\mathcal{F}}(\boldsymbol{\phi}_j - \boldsymbol{\mu}_\mathcal{F})^2)/((\boldsymbol{\mu}_\mathcal{R} - \boldsymbol{\mu})^2 + (\boldsymbol{\mu}_\mathcal{F} - \boldsymbol{\mu})^2)$, where $\boldsymbol{\phi}_i, \boldsymbol{\phi}_j$ denote sample embedding, $\boldsymbol{\mu}_\mathcal{R}, \boldsymbol{\mu}_\mathcal{F}$ denote mean embedding of $\mathcal{R}, \mathcal{F}$, and $\boldsymbol{\mu}$ denotes mean embedding over $\mathcal{R} \cup \mathcal{F}$. We also implement multi-kernel Maximum Mean Discrepancy (MMD) and 2-Wasserstein distance W_2 to comprehensively evaluate the feature separation after unlearning [13, 29].

CLReg explicitly identifies and pushes away forget features, resulting in reduced entanglement. In Tab. 3 and Tab. 4, we observe that the entanglement between retain and forget features is consistently lowered, with more noticeable changes on TOFU. But does the shifted representation space alter the distributions of retained knowledge? We further visualize the feature space using UMAP [23] in Fig. 1. Compared to the forget features, which are pushed away, CLReg does not move retain features much: while NPO and SimNPO keep both feature sets within a similar compact range, CLReg keeps the retain features at their original scale and pushes the forget features far outside it (note the axis scales in Fig. 1). Even in the most challenging case (NPO, Llama-3-3B), where features remain more entangled after NPO unlearning, CLReg still separates the forget features from the retain cluster. The visualizations effectively demonstrate how CLReg can identify and push away forget features while keeping retained knowledge intact. The clear separation also inspires future work to clip the "outlier" forget features for a faithful, complete unlearning.

4.5 Separation Facilitates and Stabilizes Unlearning

Because CLReg and the base objective are optimized simultaneously, the gains above could stem either from better-separated representations or simply from the additional optimization signal. We distinguish the two by intervening on the representations.

Representation shaping facilitates subsequent unlearning. Stage 1 optimizes only the retain objective together with CLReg ($\gamma = 0$), with no forget loss; stage 2 removes CLReg entirely and runs the original base method unchanged, so only the initialization differs from a standard run. Against a 15-epoch control matching the $5 + 10$ total epochs, the warm start still adds $+0.037$ (SimNPO) and $+0.050$ (NPO) (Tab. 5a). Stage 1 alone reduces `ExtractionStrength` from 0.890 to 0.434 while `ModelUtility` moves only from 0.666 to 0.637: shaping the representation space, with no forgetting objective involved, measurably eases forgetting by an unmodified base method.

Reduced entanglement increases resistance to relearning. If CLReg merely made forget information more separable rather than removing it, a relearning attack would recover it readily. We fine-tune each unlearned model on 10% of the forget set (40 examples, lr 10^{-5}, 64 steps, identical across arms) and report the normalized area under the forget-probability recovery curve (Supp. A.8.2). Plain

SimNPO closes 53% of the gap back to the fine-tuned model within 32 steps, so suppression does leak. CLReg makes recovery materially harder for two of three methods (Tab. 5b); for SimNPO the paired effect over three seeds is -0.101 ± 0.032, negative in every seed, and these are the checkpoints with far lower entanglement ($17.19 \rightarrow 5.63$). NPO barely relearns at all, leaving no headroom. CLReg slows relearning rather than preventing it.

Beyond single-request unlearning. Over four consecutive MUSE-News requests, SimNPO's retain knowledge erodes by 37.1% but SimNPO+CL's by only 3.0%, though its per-request forgetting weakens along the chain (Supp. A.8.3). On 13 capability and alignment metrics, 8 improve and none declines by more than 0.037 (Supp. A.8.4). Neither model ever refuses on forget queries; CLReg's lower output coherence at $\lambda = 1$ is an operating point on the regularization path, reversed at $\lambda = 0.3$ (Supp. A.8.5).

4.6 Which Layer to Regularize?

Table 6: TOFU SimNPO+CL layer selection. Unlearning in earlier layers might harm fundamental knowledge.

SimNPO+CL Llama-3-8B	Forget Score↑	Model Utility↑	**Unlearning Score↑**	Privacy Leak→0
Last 1	0.97182	0.69815	**0.81256**	55.03337
Last 4	0.86847	0.67944	0.76241	62.32824
Last 7	0.95680	0.67037	0.78837	62.05757
Last 10	0.99620	0.67295	0.80328	51.64990
Last 13	0.99881	0.66042	0.79511	48.281912
SimNPO+CL Llama-3-3B	Forget Score↑	Model Utility↑	**Unlearning Score↑**	Privacy Leak→0
Last 1	0.96209	0.66531	**0.78664**	-3.72569
Last 4	0.94276	0.62183	0.74938	-11.10044
Last 7	0.91280	0.59783	0.72248	-27.33673
Last 10	0.85324	0.62374	0.72066	-54.43742
Last 13	0.85609	0.63859	0.73151	-57.77889

Table 7: TOFU NPO+CL layer selection. Similar to Tab. 6, the performance will be negatively impacted with earlier layers.

NPO+CL Llama-3-8B	Forget Score↑	Model Utility↑	**Unlearning Score↑**	Privacy Leak→0
Last 1	0.87003	0.68487	**0.76643**	-56.01587
Last 4	0.81545	0.66091	0.73009	-67.07854
Last 7	0.84904	0.64266	0.73157	-56.63058
Last 10	0.86303	0.65279	0.74333	-52.64892
Last 13	0.85474	0.66738	0.74953	-51.92987
NPO+CL Llama-3-3B	Forget Score↑	Model Utility↑	**Unlearning Score↑**	Privacy Leak→0
Last 1	0.84701	0.63707	**0.72719**	-64.04083
Last 4	0.80283	0.62728	0.70428	-72.16869
Last 7	0.81462	0.62554	0.70767	-71.69250
Last 10	0.80590	0.62932	0.70675	-74.76727
Last 13	0.80150	0.62739	0.70384	-75.01047

Intuitively, $\mathcal{F}$-specific concepts are higher-level features residing in later layers, while earlier layers learn fundamental knowledge and common concepts shared among $\mathcal{R}$ and $\mathcal{F}$. We verify this intuition empirically by performing an ablation on which feature layer to perform CLReg on. We select last $[1, 4, 7, 10, 13]$ feature layer and conduct SimNPO+CL and NPO+CL experiments on TOFU, and report evaluation results in Tab. 6 and Tab. 7. As we apply CLReg to earlier layers, the performance will degrade, resulting in consistently reduced `ModelUtility` and `UnlearningScore`. Since current forget sets consist of *high-level, semantic knowledge*, which prior work finds to be concentrated in later transformer layers [14], we hypothesize that unlearning happens most effectively there. CLReg itself is agnostic to the target layer; for concepts residing in earlier layers, such as syntactic or stylistic patterns, layer selection guided by layer-wise probing [28] or attribution is a natural direction for future work.

5 Conclusion

In this work, we argue that explicit representation shaping does not undermine the goal of unlearning to match the retrained model's behaviors: separating forget concepts from retain concepts in the feature space lets a base unlearning objective remove them with less collateral damage. CLReg is a regularizer that makes the base objective's removal more effective and more durable, rather than a mechanism that removes knowledge on its own. Our one-step embedding-space analysis motivates why a contrastive update reduces an entanglement proxy, without formal multi-step or retain-side guarantees. Empirically, across four LLMs, five base objectives, and three benchmarks, CLReg reduces entanglement, improves the aggregate unlearning score, and preserves model utility at the cost of one extra forward pass; shaping representations before any forgetting objective improves subsequent unlearning, and the resulting models are harder to relearn and more stable under repeated requests. We hope our study inspires unlearning work that removes forget concepts through representation shaping.

Acknowledgement. RK thanks the Central Indiana Corporate Partnership AnalytiXIN Initiative and NSF Award 2543174 for their support.

References

[1] Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A theoretical analysis of contrastive unsupervised representation learning. In *International Conference on Machine Learning (ICML)*, pages 5628–5637. PMLR, 2019. URL https://arxiv.org/abs/1902.09229.

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, pages 1597–1607. PMLR, 2020. URL https://arxiv.org/abs/2002.05709.

[3] Yijiang River Dong, Hongzhou Lin, Mikhail Belkin, Ramon Huerta, and Ivan Vulić. UNDIAL: Self-distillation with adjusted logits for robust unlearning in large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics (NAACL)*, 2025. doi: 10.18653/v1/2025.naacl-long.444. URL https://aclanthology.org/2025.naacl-long.444/.

[4] Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zachary C Lipton, J Zico Kolter, and Pratyush Maini. OpenUnlearning: Accelerating LLM unlearning via unified benchmarking of methods and metrics. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2025. URL https://openreview.net/forum?id=Gy67Zh5X1i.

[5] Taha Entesari, Arman Hatami, Rinat Khaziev, Anil Ramakrishna, and Mahyar Fazlyab. Constrained entropic unlearning: A primal-dual framework for large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://openreview.net/forum?id=ZtB34bQI54.

[6] Chongyu Fan, Jiancheng Liu, Yihua Zhang, Dennis Wei, Eric Wong, and Sijia Liu. SalUn: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2310.12508.

[7] Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for LLM unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://arxiv.org/abs/2410.07163.

[8] Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021. doi: 10.18653/v1/2021.emnlp-main.552. URL https://aclanthology.org/2021.emnlp-main.552/.

[9] Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9304–9312, 2020. doi: 10.1109/CVPR42600.2020.00932.

[10] Micah Goldblum, Steven Reich, Liam Fowl, Renkun Ni, Valeriia Cherepanova, and Tom Goldstein. Unraveling meta-learning: Understanding feature representations for few-shot tasks. In *International Conference on Machine Learning (ICML)*, pages 3607–3616. PMLR, 2020. URL https://arxiv.org/abs/2002.06753.

[11] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

[12] Laura Graves, Vineel Nagisetty, and Vijay Ganesh. Amnesiac machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 11516–11524, 2021. doi: 10.1609/aaai.v35i13.17371.

[13] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012. URL http://jmlr.org/papers/v13/gretton12a.html.

[14] Yihuai Hong, Yuelin Zou, Lijie Hu, Ziqian Zeng, Di Wang, and Haiqin Yang. Dissecting fine-tuning unlearning in large language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3933–3941, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.228. URL https://aclanthology.org/2024.emnlp-main.228/.

[15] Zachary Izzo, Mary Anne Smart, Kamalika Chaudhuri, and James Zou. Approximate data deletion from machine learning models. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 2008–2016. PMLR, 2021. URL https://arxiv.org/abs/2002.10077.

[16] Yasser H. Khalil, Mehdi Setayesh, and Hongliang Li. CoUn: Empowering machine unlearning via contrastive learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://openreview.net/forum?id=2Lq5LnWLa9.

[17] Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 1957–1987, 2023. URL https://arxiv.org/abs/2302.09880.

[18] Hong kyu Lee, Qiuchen Zhang, Carl Yang, Jian Lou, and Li Xiong. Contrastive unlearning: A contrastive approach to machine unlearning. In James Kwok, editor, *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 7464–7472. International Joint Conferences on Artificial Intelligence Organization, 2025. doi: 10.24963/ijcai.2025/830. URL https://doi.org/10.24963/ijcai.2025/830.

[19] Kemou Li, Qizhou Wang, Yue Wang, Fengpeng Li, Jun Liu, Bo Han, and Jiantao Zhou. LLM unlearning with LLM beliefs. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2510.19422.

[20] Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, et al. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *International Conference on Machine Learning (ICML)*, 2024. URL https://arxiv.org/abs/2403.03218.

[21] Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Yuguang Yao, Chris Yuhao Liu, Xiaojun Xu, Hang Li, Kush R. Varshney, Mohit Bansal, Sanmi Koyejo, and Yang Liu. Rethinking machine unlearning for large language models. *Nature Machine Intelligence*, 7:181–194, 2025. doi: 10.1038/s42256-025-00985-0.

[22] Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C Lipton, and J Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. In *Conference on Language Modeling (COLM)*, 2024. URL https://arxiv.org/abs/2401.06121.

[23] Leland McInnes, John Healy, and James Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018. URL https://arxiv.org/abs/1802.03426.

[24] Yu Meng, Mengzhou Xia, and Danqi Chen. SimPO: Simple preference optimization with a reference-free reward. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pages 124198–124235, 2024. URL https://arxiv.org/abs/2405.14734.

[25] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. URL https://arxiv.org/abs/1807.03748.

[26] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 53728–53741, 2023. URL https://arxiv.org/abs/2305.18290.

[27] Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A Smith, and Chiyuan Zhang. MUSE: Machine unlearning six-way evaluation for language models. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2407.06460.

[28] Oscar Skean, Md Rifat Arefin, Dan Zhao, Niket Patel, Jalal Naghiyev, Yann LeCun, and Ravid Shwartz-Ziv. Layer by layer: Uncovering hidden representations in language models. In *International Conference on Machine Learning (ICML)*, 2025. URL https://arxiv.org/abs/2502.02013.

[29] Haoran Tang and Rajiv Khanna. Sharpness-aware machine unlearning. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2506.13715.

[30] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. URL https://arxiv.org/abs/2307.09288.

[31] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning (ICML)*, pages 9929–9939. PMLR, 2020. URL https://arxiv.org/abs/2005.10242.

[32] Yaxuan Wang, Jiaheng Wei, Chris Yuhao Liu, Jinlong Pang, Quan Liu, Ankit Parag Shah, Yujia Bao, Yang Liu, and Wei Wei. LLM unlearning via loss adjustment with only forget data. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=6ESRicalFE.

[33] Alexander Warnecke, Lukas Pirch, Christian Wressnegger, and Konrad Rieck. Machine unlearning of features and labels. In *Network and Distributed System Security Symposium (NDSS)*, 2023. doi: 10.14722/ndss.2023.23087.

[34] Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In *International Conference on Machine Learning (ICML)*, 2024. URL https://arxiv.org/abs/2401.08417.

[35] Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024. URL https://arxiv.org/abs/2310.10683.

[36] Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. In *Conference on Language Modeling (COLM)*, 2024. URL https://arxiv.org/abs/2404.05868.

[37] Kairan Zhao, Meghdad Kurmanji, George-Octavian Bărbulescu, Eleni Triantafillou, and Peter Triantafillou. What makes unlearning hard and what to do about it. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pages 12293–12333, 2024. URL https://arxiv.org/abs/2406.01257.

A Appendix

A.1 Proofs 14
A.2 Overview of TOFU and MUSE 16
A.3 Detailed Experiment Settings 16
A.3.1 Unlearning Hyper-parameters 16
A.3.2 Experiment Environment 17
A.4 Implementation Details and Paraphrase Generation 17
A.5 Overview of Evaluation Metrics 18
A.6 WMDP Unlearning Experiments 18
A.7 Ablation on CL-Loss Variants and Symmetry 19
A.8 Additional Analyses 20
A.8.1 Number and Selection of Negatives 20
A.8.2 Relearning Attack and Seed Variance 20
A.8.3 Sequential Unlearning 21
A.8.4 General Capability and Alignment 21
A.8.5 Forget-Set Output Quality 21
A.8.6 Model Size 22
A.8.7 When CLReg Degrades Performance 22
A.9 Computational Overhead 22
A.10 Limitations and Future Work 23
A.11 Broader Impacts 24

A.1 Proofs

Proof to Proposition 3.3. Consider a single $\mathcal{L}_{\text{CL}}^{\text{dpo}}$ term with i-th forget sample and j-th retain sample

$$\ell_{ij}(\theta) = -2\tau \log \sigma\Big(\tfrac{m_{ij}}{\tau}\Big), m_{ij} = s(a_i, p_i) - s(a_i, n_j),$$

where $a_i = \zeta_\theta(x_{f_i})$ is the anchor (forget embedding), $p_i = \zeta_\theta(x_{f_i}^+)$ is its positive (paraphrased or dropout-augmented), $n_j = \zeta_\theta(x_{r_j})$ is a negative (retain embedding), and $s(u, v) = u^\top v$. Suppose $a_i, p_i, n_j \in \mathbb{S}^{d-1}$. The gradient of ℓ_{ij} with respect to a_i is

$$\nabla_{a_i} \ell_{ij} = -2\,\sigma\Big(-\tfrac{m_{ij}}{\tau}\Big)\,\big(p_i - n_j\big).$$

In particular, a gradient descent step $a_i' = a_i - \eta\,\nabla_{a_i}\ell_{ij}$ (with small $\eta > 0$) moves a_i *toward* p_i and *away from* n_j. If $p_i \neq n_j$ and $\eta > 0$, then ${a_i'}^\top n_j \;<\; a_i^\top n_j$, so the anchor–negative cosine similarity strictly decreases.

Proof. Write $u = m_{ij}/\tau$ and compute

$$\partial(-2\tau \log \sigma(u))/\partial u = -2(1 - \sigma(u)) = -2\sigma(-u).$$

By chain rule,

$$\nabla_{a_i} \ell_{ij} = -2\,\sigma\Big(-\tfrac{m_{ij}}{\tau}\Big)\,\nabla_{a_i} m_{ij} \tag{5}$$

$$= -2\,\sigma\Big(-\tfrac{m_{ij}}{\tau}\Big)\,\big(p_i - n_j\big), \tag{6}$$

because $\nabla_{a_i}(a_i^\top p_i) = p_i$ and $\nabla_{a_i}(a_i^\top n_j) = n_j$. Updating a_i by a small step in $-(\nabla_{a_i}\ell_{ij})$ yields

$$a_i' = a_i + 2\eta\,\sigma\Big(-\tfrac{m_{ij}}{\tau}\Big)\,\big(p_i - n_j\big).$$

Taking the dot product with n_j: ${a_i'}^\top n_j = a_i^\top n_j + 2\eta\,\sigma(-m_{ij}/\tau)\,(p_i^\top n_j - \|n_j\|^2)$. Since $\|n_j\|^2 = 1$ (normalized) and $p_i^\top n_j \le 1$ (Cauchy–Schwarz), with strict inequality when $p_i \neq n_j$, the increment is negative. Thus, ${a_i'}^\top n_j < a_i^\top n_j$ whenever $p_i \neq n_j$. □

Proof to Corollary 3.4. Under the same setup as Proposition 3.3, consider the expected cross-similarity (linear kernel overlap)

$$C_{\text{lin}}(\theta) = \mathbb{E}\big[a^\top n\big],$$

where $a = \zeta_\theta(x_f)$ and $n = \zeta_\theta(x_r)$ for independent $x_f \in \mathcal{F}$ and $x_r \in \mathcal{R}$. If a single gradient step on ℓ_{ij} updates a_i to a_i' while leaving p_i and all n_j fixed, then with updated parameters θ',

$$C_{\text{lin}}(\theta') \;\le\; C_{\text{lin}}(\theta),$$

with strict inequality if $p_i \neq n_j$ for any updated pair. Hence, the DPO-CL update strictly reduces the expected anchor–negative similarity.

Proof. Averaging the inequality ${a_i'}^\top n_j \le a_i^\top n_j$ from Proposition 3.3 over the sampled indices (i, j) yields C_{lin} non-increasing. If at least one updated pair has $p_i \neq n_j$, the inequality is strict. □

Proof to Proposition 3.5. Let D be any separation measure between distributions satisfying the following:

- There exists a continuous cost function $c : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ such that $D(\mathcal{P}, \mathcal{Q})$ is a non-decreasing function of the expected cross-cost $\mathbb{E}_{u\sim\mathcal{P},\, v\sim\mathcal{Q}}[\, c(u, v)\,]$; that is, $c(u, v_1) \le c(u, v_2)$ implies $D(P, \delta_{v_1}) \le D(P, \delta_{v_2})$, where δ_v denotes the Dirac measure at v.
- The cost c is strictly increasing with respect to the anchor–negative similarity: if $s(u_1, v) < s(u_2, v)$ then $c(u_1, v) > c(u_2, v)$.

Then a gradient descent step on $\mathcal{L}_{\text{CL}}^{\text{dpo}}$ reduces $\mathbb{E}[s(a, n)]$ and thereby *increases* $D(\mathcal{P}_{\mathcal{F}}^{\theta}, \mathcal{P}_{\mathcal{R}}^{\theta})$:

$$D\big(\mathcal{P}_{\mathcal{F}}^{\theta'},\, \mathcal{P}_{\mathcal{R}}^{\theta'}\big) \;\ge\; D\big(\mathcal{P}_{\mathcal{F}}^{\theta},\, \mathcal{P}_{\mathcal{R}}^{\theta}\big),$$

with strict increase when at least one updated anchor has $p_i \neq n_j$.

Proof. Corollary 3.4 guarantees that a DPO-CL update decreases $\mathbb{E}[s(a, n)]$, i.e., anchors are less aligned with negatives. By assumption, $c(u, v)$ increases strictly when similarity $s(u, v)$ decreases. Therefore, $\mathbb{E}[c(a, n)]$ strictly increases. Condition (1) ensures that D is a non-decreasing function of $\mathbb{E}[c(a, n)]$. Consequently, after the update, $D(\mathcal{P}_{\mathcal{F}}^{\theta'}, \mathcal{P}_{\mathcal{R}}^{\theta'})$ is no less than before. When at least one anchor–negative pair is strictly repelled, $\mathbb{E}[c(a, n)]$ increases strictly, leading to D strictly increasing. □

Multi-negative form. Propositions 3.3–3.5 analyze a single anchor–negative pair. Eq. (1) contrasts every anchor with all B_r negatives, which the following makes precise.

Proposition A.1. ***(Anchor update with multiple negatives).*** *Let $L_i(\theta) = \frac{1}{B_f B_r} \sum_{j=1}^{B_r} \ell_{ij}(\theta)$ collect the terms of Eq.* (1) *for the i-th forget sample, with ℓ_{ij}, m_{ij} and $a_i, p_i, n_j \in \mathbb{S}^{d-1}$ as in Proposition 3.3. Let $w_{ij} = \sigma(-m_{ij}/\tau)$, $W_i = \sum_j w_{ij}$, and let $\bar{n}_w = W_i^{-1} \sum_j w_{ij} n_j \neq 0$ be the w-weighted centroid of the negatives. Then*

$$\nabla_{a_i} L_i = -\frac{2}{B_f B_r}\Big[W_i\, p_i - \sum_j w_{ij}\, n_j\Big] = -\frac{2W_i}{B_f B_r}\big(p_i - \bar{n}_w\big),$$

and a gradient step $a_i' = a_i - \eta \nabla_{a_i} L_i$ with $\eta > 0$ satisfies ${a_i'}^\top \bar{n}_w < a_i^\top \bar{n}_w$ if and only if $\cos(p_i, \bar{n}_w) < \|\bar{n}_w\|$. For $B_r = 1$ the condition reduces to $p_i \neq n_1$, recovering Proposition 3.3.

Proof. By Proposition 3.3, $\nabla_{a_i} \ell_{ij} = -2w_{ij}(p_i - n_j)$. Summing over j and dividing by $B_f B_r$ gives the first expression, and substituting $\sum_j w_{ij} n_j = W_i \bar{n}_w$ gives the second. Hence $a_i' = a_i + \frac{2\eta W_i}{B_f B_r}(p_i - \bar{n}_w)$ and

$${a_i'}^\top \bar{n}_w - a_i^\top \bar{n}_w = \frac{2\eta W_i}{B_f B_r}\big(p_i^\top \bar{n}_w - \|\bar{n}_w\|^2\big).$$

Since $\eta > 0$ and $W_i > 0$, the change is negative if and only if $p_i^\top \bar{n}_w < \|\bar{n}_w\|^2$. As $\|p_i\| = 1$, $p_i^\top \bar{n}_w = \|\bar{n}_w\| \cos(p_i, \bar{n}_w)$, so for $\bar{n}_w \neq 0$ this is exactly $\cos(p_i, \bar{n}_w) < \|\bar{n}_w\|$. When $B_r = 1$, $\bar{n}_w = n_1$ has unit norm and the condition reads $\cos(p_i, n_1) < 1$, i.e., $p_i \neq n_1$. □

The weights w_{ij} are largest for negatives that the anchor currently finds most similar, so hard negatives are up-weighted automatically, with τ controlling the sharpness. Because $\|\bar{n}_w\| < 1$ unless all weighted negatives coincide, the condition is stricter than its single-negative counterpart and is not automatic. We therefore logged it during training: across our CLReg runs it holds for 94% of anchor updates (16,660 of 17,728), with $\cos(p_i, \bar{n}_w)$ averaging 0.34 against $\|\bar{n}_w\|$ averaging 0.56. We read it as an empirical regularity rather than a guarantee.

A.2 Overview of TOFU and MUSE

We provide an overview of benchmarks used in our work with concrete examples to better illustrate the differences and difficulties of each dataset:

The TOFU benchmark [22] consists of question-answer pairs with short length, based on autobiographies of 200 different authors that are fictitiously generated by GPT-4. The unlearning objective is to forget the fictitious answers in the forget set $\mathcal{F}$. A data example is as follows:

- ***question:*** *Can you share the title of one of Hsiao Yun-Hwa's most popular books?*
- ***answer:*** *One of Hsiao Yun-Hwa's most popular books in the leadership genre is "Artistic Authority: Leading with Creativity".*

It provides multiple data splits and includes paraphrased and perturbed versions for extended use.

The MUSE benchmark [27] consists of two distinct datasets, Books and News. In Books, the forget set $\mathcal{F}$ consists of the original Harry Potter books by J. K. Rowling and the retain set of related content from the Harry Potter FanWiki; examples are long passages. Multiple evaluation sets are designed to be question-answer or prompt-response pairs with long text lengths. Due to length, here we only show an example of question-answer pair for `KnowMem` evaluation set on forget concepts (they are short):

- ***question:*** *What were the two new books mentioned in Harry's letter that he needed for the coming year?*
- ***answer:*** *The Standard Book of Spells, Grade 5, by Miranda Goshawk, and Defensive Magical Theory, by Wilbert Slinkhard.*

The News dataset consists of BBC news articles collected after August 2023, randomly split into disjoint forget, retain, and holdout sets; its examples are shorter than those in Books. Multiple evaluation sets are designed to be question-answer or prompt-response pairs with long text lengths. Here we only show an example of question-answer pair for `KnowMem` evaluation set on forget concepts (they are short):

- ***question:*** *Which three nuclear power plants were taken offline in Germany by midnight on Saturday?*
- ***answer:*** *Isar 2, Emsland and Neckarwestheim 2*

Both Books and News have evaluation sets with long-length examples dedicated to privacy metric `PrivLeak`.

A.3 Detailed Experiment Settings

A.3.1 Unlearning Hyper-parameters

We unlearn with GradDiff, NPO, SimNPO, UnDIAL, PDU [36, 7, 3, 5]. We provide detailed hyper-parameter settings for each unlearning method:

GradDiff: GradDiff can be unstable due to aggressive ascent and requires careful tuning. We choose $\gamma = 0.01$ on MUSE-Books and $\gamma = 1$ on MUSE-News. Performance becomes more stable on TOFU and we pick $\gamma = 0.4$ for both Llama-3-8B and 3B.

NPO: We fix $\gamma = \alpha = 1$ and search for β values. We pick $\beta = 0.05$ on both MUSE-Books and News, and $\beta = 0.4$ on TOFU for both Llama-3-8B and 3B.

SimNPO: We fix $\gamma = \alpha = 1$ and search for β values. We pick $\beta = 1$ on both MUSE-Books and $\beta = 0.05$ for News, and $\beta = 1.5$ on TOFU for both Llama-3-8B and 3B. Note that as β becomes smaller for NPO and SimNPO, they behave more similar to GradDiff [24, 7].

UnDIAL: We observe that UnDIAL barely needs retaining and we set $\alpha = 0, \gamma = 1$. UnDIAL has another β term as the strength of penalty for memorized tokens. We pick $\beta = 10$ on both MUSE-Books and News, and $\beta = 15$ on TOFU for both Llama-3-8B and 3B.

PDU: PDU performs steadily across settings. We adopt step size 1 and $\gamma = \alpha = 1$. We adopt 1 warmup epoch for MUSE-Books, News, and TOFU Llama-3-3B. We adopt 2 warmup epochs for TOFU Llama-3-8B.

After obtaining optimal baseline results, we then add CLReg with $\lambda = 1$ and tune CLReg-specific parameters: $\tau \times$ [symmetric, non-symmetric] $\times [\mathcal{L}_{\text{CL}}^{\text{dpo}}, \mathcal{L}_{\text{CL}}^{\text{info}}]$. We fix $\alpha, \lambda = 1$. Specifically, we sweep τ in $[0.1, 0.3, 0.5, 0.7, 0.9]$, and obtain optimal settings for each unlearning method + CLReg:

GradDiff: We pick $[0.1, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{dpo}}]$ for MUSE-Books, $[0.3, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{dpo}}]$ for MUSE-News, $[0.1, \text{symmetric}, \mathcal{L}_{\text{CL}}^{\text{info}}]$ for TOFU Llama-3 3B, and $[0.3, \text{symmetric}, \mathcal{L}_{\text{CL}}^{\text{dpo}}]$ for TOFU Llama-3 8B.

NPO: We pick $[0.3, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{info}}]$ for MUSE-Books, $[0.9, \text{symmetric}, \mathcal{L}_{\text{CL}}^{\text{info}}]$ for MUSE-News, $[0.5, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{info}}]$ for TOFU Llama-3 3B, and $[0.7, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{dpo}}]$ for TOFU Llama-3 8B.

SimNPO: We pick $[0.3, \text{symmetric}, \mathcal{L}_{\text{CL}}^{\text{dpo}}]$ for MUSE-Books, $[0.3, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{dpo}}]$ for MUSE-News, $[0.5, \text{symmetric}, \mathcal{L}_{\text{CL}}^{\text{info}}]$ for TOFU Llama-3 3B, and $[0.9, \text{symmetric}, \mathcal{L}_{\text{CL}}^{\text{dpo}}]$ for TOFU Llama-3 8B.

UnDIAL: We pick $[0.3, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{info}}]$ for MUSE-Books, $[0.5, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{dpo}}]$ for MUSE-News, $[0.1, \text{symmetric}, \mathcal{L}_{\text{CL}}^{\text{dpo}}]$ for TOFU Llama-3 3B, and $[0.5, \text{symmetric}, \mathcal{L}_{\text{CL}}^{\text{dpo}}]$ for TOFU Llama-3 8B. Additionally, we slightly increase α to 0.1 to balance the addition of CLReg.

PDU: We pick $[0.9, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{info}}]$ for MUSE-Books, $[0.3, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{info}}]$ for MUSE-News, $[0.3, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{info}}]$ for TOFU Llama-3 3B, and $[0.5, \text{non-symmetric}, \mathcal{L}_{\text{CL}}^{\text{dpo}}]$ for TOFU Llama-3 8B.

A.3.2 Experiment Environment

We adapt existing, open-source code base and datasets for conducting experiments and developing new algorithms [2] [22, 27, 4]. All experiments are conducted on NVIDIA H100 GPUs.

A.4 Implementation Details and Paraphrase Generation

Paraphrase positives. On TOFU, forget questions are paraphrased once, offline, by Llama-3.1-8B-Instruct using its chat template and the instruction *"Paraphrase the question using different wording but identical meaning. Preserve ALL names, numbers, dates, and factual content."* We sample with temperature 0.9, top-p 0.95, top-k 50 and repetition penalty 1.05, and keep five paraphrases per question. Only the question is paraphrased; the answer is left untouched, so the positive is a semantically equivalent query with the same target. At each step one of the five paraphrases is drawn uniformly for every forget example, and dropout is then applied independently to the anchor and the positive. MUSE provides no paraphrases, so its positive is the same passage under an independent dropout mask, in the manner of SimCSE [8].

Batch size and negatives. We train with 8 data-parallel workers, a per-device batch of 4, and gradient accumulation of 4 (TOFU) or 8 (MUSE), for an effective batch size of 128 and 256 respectively. Retain embeddings are all-gathered across workers before the contrastive loss is

[2] `https://github.com/locuslab/open-unlearning`, `https://huggingface.co/datasets/locuslab/TOFU`, `https://huggingface.co/datasets/muse-bench/MUSE-Books`, `https://huggingface.co/datasets/muse-bench/MUSE-News`

computed, so every anchor is contrasted with $8 \times 4 = 32$ negatives. Gradient accumulation does not add negatives: their number is set by the data-parallel world size and the per-device batch, not by the effective batch size. Supp. A.8.1 shows that the results are insensitive to it.

A.5 Overview of Evaluation Metrics

We provide an overview of evaluation metrics adopted in our work. Many of the metrics can be applied to both $\mathcal{R}$ and $\mathcal{F}$ while expecting inverse behaviors.

Memorization metrics, which quantifies how much information the data sample has been memorized:

- **Probability**: Quantifies the model's confidence in its output: $\mathrm{Prob} = p(\mathbf{W}(y \mid x))$.
- **ROUGE**: Quantifies the degree of overlap between model output and the ground truth.
- **Truth Ratio**: Measures the model's preference for the correct answer over its incorrect variants. A higher value indicates stronger confidence in the correct response, making it privacy-aware.
- **Exact Memorization (EM)**: Similar to `ROUGE`, `EM` quantifies memorization by calculating proportion of matched tokens in the model output with ground truth.
- **Extraction Strength (ES)**: Quantifies memorization by determining the minimal prefix length required to reconstruct the suffix.

Privacy metrics, which evaluates whether sensitive information from the forget set can still be inferred or extracted:

- **PrivLeak** [27]: Calibrated Membership Inference Attack (MIA) with AUC scores from the retain model:
$$\mathrm{PrivLeak} = \frac{\mathrm{AUC}(\mathbf{W}_{\mathrm{UL}}, \mathcal{F}) - \mathrm{AUC}(\mathbf{W}_{\mathrm{RT}}, \mathcal{F})}{\mathrm{AUC}(\mathbf{W}_{\mathrm{RT}}, \mathcal{F})}.$$
- **Forget Quality**: Performs KS statistical test on `TruthRatio` distributions of $\mathbf{W}_{\mathrm{UL}}$ and $\mathbf{W}_{\mathrm{RT}}$, yielding p values which are high when the two distributions are close.

Utility metrics, which can reuse memorization metrics to ensure that retain knowledge is well maintained:

- **Model Utility**: Harmonic mean of `Prob, ROUGE, TruthRatio` on the retain set $\mathcal{R}$.
- **KnowMem ROUGE**: Measures `ROUGE` on knowledge-based questions regarding $\mathcal{R}$.

See detailed explanation and discussion in TOFU and MUSE [22, 27]. The aggregate `ForgetScore` and `UnlearningScore` reported in our main tables (Tabs. 1–2) are introduced in the Evaluation subsection of Sec. 4 as harmonic means of per-metric progress measures, and are not separately re-defined here.

A.6 WMDP Unlearning Experiments

We further evaluate CLReg on WMDP [20] (cyber subset) using Zephyr-7B-β, a hazardous-knowledge unlearning regime where forget knowledge is acquired during *web-scale pretraining* and a true retrained reference $\mathbf{W}_{\mathrm{RT}}$ is infeasible. We follow the WMDP/RMU protocol and use the random-guess accuracy on 4-option MCQ ($1/4 = 0.25$) as a hard lower-bound proxy for $m(\mathbf{W}_{\mathrm{RT}}, \mathcal{F})$ (a real RT would likely score ≥ 0.25, so Prog is conservatively a lower bound on true forget progress); $\mathbf{W}_{\mathrm{FT}}$ scores (0.581 MMLU and 0.440 WMDP-cyber) are taken from the published `cais/Zephyr_RMU` release for harness consistency. WMDP has a single forget metric (`wmdp_cyber/acc`), so $\mathrm{ForgetScore} = \mathrm{Prog}(\mathrm{cyber})$ directly; the harmonic-mean step degenerates to identity when $K = 1$. `PrivLeak` is not defined for WMDP and is omitted. Results are reported in Tab. 8.

On WMDP-cyber, SimNPO+CL is the standout: it deepens forget (cyber $0.289 \rightarrow 0.260$) and lifts `ForgetScore` from 0.795 to 0.949, with a 0.041 `UnlearningScore` gain that nearly closes the gap to RMU's published 0.677 at 0.674. GradDiff+CL closes most of the remaining gap from another direction – preserving GradDiff's already-strong forget while recovering MMLU ($0.486 \rightarrow 0.530$). NPO+CL improves over NPO by $+0.033$ `UnlearningScore` once α is rebalanced from the default

Table 8: WMDP-cyber unlearning experiment results on Zephyr-7B-β. Bold marks best in-column among non-RMU methods; green shades indicate improvement over the corresponding base method. SimNPO+CL achieves the best `UnlearningScore` (0.674), nearly matching RMU's published score (0.677); CLReg consistently improves `UnlearningScore` for *all five* base methods. NPO+CL benefits from α rebalancing (best at α=0.5), lifting `UnlearningScore` from 0.551 to 0.584. RMU numbers are taken from the published `cais/Zephyr_RMU` release rather than retuned on our side, to avoid competitor over-tuning.

WMDP-Cyber Zephyr-7B-β	WMDP-Cyber Acc↓	Forget Score↑	MMLU (Model Utility)↑	**Unlearning Score↑**
GradDiff	**0.262**	**0.938**	0.486	0.640
UNDIAL	0.365	0.395	0.538	0.456
NPO	0.331	0.573	0.531	0.551
SimNPO	0.289	0.795	0.526	0.633
PDU	0.366	0.390	**0.572**	0.464
RMU (paper)	0.282	0.832	0.571	0.677
GradDiff+CL	0.268	0.907	0.530	0.669
UNDIAL+CL	0.363	0.406	0.540	0.464
NPO+CL	0.318	0.642	0.537	0.584
SimNPO+CL	**0.260**	**0.949**	0.522	**0.674**
PDU+CL	0.355	0.446	0.556	0.495

1.0 down to 0.5 (best run: α=0.5, τ=0.5, InfoNCE), aligning with our TOFU/MUSE finding that NPO benefits consistently from CLReg. PDU+CL also improves both axes (+0.031). UNDIAL+CL shows the smallest margin: the base method already operates near a forget floor on this benchmark that CLReg can only marginally lift.

A.7 Ablation on CL-Loss Variants and Symmetry

We ablate the two main hyperparameter choices inside CLReg – the contrastive loss form ($\mathcal{L}_{\text{CL}}^{\text{dpo}}$ vs. $\mathcal{L}_{\text{CL}}^{\text{info}}$) and whether to symmetrize the anchor/negative roles between the forget and retain sides. Tab. 9 reports `UnlearningScore` on TOFU across the 2×2 grid for both NPO+CL and SimNPO+CL on Llama-3-8B and Llama-3-3B; baseline (no CLReg) `UnlearningScore` values for reference are NPO: 0.742 (8B)/0.686 (3B) and SimNPO: 0.774/0.713. 14 of the 16 CLReg configurations improve over the corresponding baseline, confirming that the benefit of CLReg is *robust across contrastive variants* rather than relying on a single particular setting; different method/model pairs favor different configurations, as expected for hyperparameter-level choices. We report the per-setting best configurations used in Tables 1–2 in Supp. A.3.

Table 9: CLReg ablation on contrastive-loss variant and symmetry on TOFU (`UnlearningScore`, higher is better). "sym"/"nonsym" denotes whether the forget/retain anchor–negative roles are symmetrized. 14/16 configurations beat the CLReg-free baseline; bold marks the per-row best.

(a) Llama-3-8B.

Llama-3-8B	DPO nonsym	DPO sym	InfoNCE nonsym	InfoNCE sym
NPO+CL	**0.766**	0.749	0.742	0.751
SimNPO+CL	0.801	**0.813**	0.711	0.790

(b) Llama-3-3B.

Llama-3-3B	DPO nonsym	DPO sym	InfoNCE nonsym	InfoNCE sym
NPO+CL	0.715	0.701	**0.727**	0.724
SimNPO+CL	0.779	0.776	0.780	**0.787**

A.8 Additional Analyses

The analyses below complement Sec. 4.5. Every arm, baselines included, is re-run from the publicly released benchmark checkpoints, so absolute scores differ slightly from Tabs. 1–2, while every comparison within a table is matched. Unless stated otherwise they use SimNPO on TOFU with Llama-3.2-3B and its main-table CLReg configuration ($\tau = 0.5$, symmetric, $\mathcal{L}_{\text{CL}}^{\text{info}}$).

A.8.1 Number and Selection of Negatives

To test whether the results depend on the number of negatives, we contrast each anchor with K of the 32 available negatives, sampled uniformly, and vary K with everything else fixed; the baseline is a matched run without CLReg (Tab. 10). Even $K = 1$ improves over the baseline by 0.079, and the spread across $K = 1$ to 32 is 0.006, comparable to the seed-to-seed standard deviation of this arm (0.005, Tab. 11). Selecting the most similar (hard), random, or least similar (easy) retain samples as the $K = 8$ negatives orders as expected but spans only 0.012. Together with the implicit hard-negative weighting of Prop. A.1, this suggests that the number of negatives, their hardness, and τ modulate a single quantity, the strength of contrastive repulsion, to which `UnlearningScore` is insensitive across the range we tested; the negative count can therefore be reduced without loss.

Table 10: Number and selection of negatives (`UnlearningScore`, SimNPO+CL, TOFU Llama-3.2-3B). Left: K negatives sampled uniformly per anchor. Right: choice of the $K = 8$ negatives.

K	0 (no CL)	1	2	4	8	16	32
`UnlearningScore`	0.724	0.803	0.806	0.809	0.808	0.809	0.807

mining	hard	random	easy
`UnlearningScore`	0.811	0.808	0.799

A.8.2 Relearning Attack and Seed Variance

Protocol. Starting from each unlearned checkpoint, the attacker fine-tunes on a random 10% of the forget set (40 of 400 examples) with AdamW, learning rate 10^{-5}, batch size 4, gradient clipping at 1.0, for 64 steps, identical across all arms so that only the starting checkpoint differs. We evaluate after $\{0, 1, 2, 4, 8, 16, 32, 64\}$ steps.

Recovery AUC. Let p_t be the forget-answer probability on the full forget set after the t-th evaluation, p_0 its value at the unlearned checkpoint, and p_{FT} its value under $\mathbf{W}_{\text{FT}}$. We rescale $y_t = (p_t - p_0)/(p_{\text{FT}} - p_0)$, so that 0 means no recovery and 1 means fully back to $\mathbf{W}_{\text{FT}}$ measured against the headroom each arm actually had, and report the trapezoidal area under y_t over the evaluation index, divided by the number of intervals. Lower is more resistant.

Seed variance. We repeat the SimNPO pair over three training seeds (Tab. 11). The gain in `UnlearningScore` from CLReg is $+0.100 \pm 0.003$, some thirty times the seed-to-seed standard deviation of either arm, and the paired change in recovery AUC is -0.101 ± 0.032, negative in every seed.

Table 11: Seed variance of the SimNPO pair (TOFU, Llama-3.2-3B). Recovery AUC is lower-is-better.

		seed 0	seed 1	seed 2	mean ± sd
`UnlearningScore`↑	SimNPO	0.688	0.692	0.691	0.690 ± 0.002
	SimNPO+CL	0.785	0.795	0.791	$\mathbf{0.791 \pm 0.005}$
recovery AUC↓	SimNPO	0.310	0.352	0.324	0.329 ± 0.021
	SimNPO+CL	0.241	0.218	0.224	$\mathbf{0.228 \pm 0.012}$

A.8.3 Sequential Unlearning

We use the MUSE-News `sust` splits: four consecutive unlearning requests, each initialized from the checkpoint produced by the previous one, with the MUSE-News hyper-parameters of Supp. A.3. Tab. 12 reports the state after each request. SimNPO's retain knowledge erodes by 37.1% over the chain, SimNPO+CL's by only 3.0%, leaving it 38% ahead after the fourth request: a separated forget subspace lets repeated erasure do less collateral damage. The counterpart is that CLReg's per-request forgetting weakens along the chain while the base method's strengthens, so the aggregate `UnlearningScore` after four requests ends comparable (0.489 vs. 0.460). Weakening the contrastive term makes the erosion worse, not better: a $\lambda = 0.3$ chain matches $\lambda = 1$ at the first request but collapses by the third.

Table 12: Sequential unlearning on MUSE-News (`sust`), four consecutive requests.

	request	1	2	3	4	change
`RetainKnowMem`↑	SimNPO	0.475	0.406	0.414	0.298	-37.1%
	SimNPO+CL ($\lambda = 1$)	0.423	0.405	0.441	**0.410**	$\mathbf{-3.0\%}$
	SimNPO+CL ($\lambda = 0.3$)	0.473	0.415	0.103	0.061	-87.2%
`ForgetScore`↑	SimNPO	0.556	0.807	0.801	1.000	$+79.7\%$
	SimNPO+CL ($\lambda = 1$)	0.839	0.766	0.713	0.604	-28.1%

A.8.4 General Capability and Alignment

We evaluate the unlearned checkpoints with the LM Evaluation Harness on knowledge, mathematics, code, truthfulness, instruction following, and commonsense reasoning (Tab. 13). Of 13 metrics, 8 improve and none declines by more than 0.037. Applying one standard to gains and losses, every difference lies within 1.2 standard errors on a single evaluation run, so we read the table as capability *preserved* rather than improved. The coding decline is reported as measured: HumanEval, the smallest benchmark at 164 problems, moves -0.037, while MBPP, three times larger, moves -0.006.

Table 13: General capability and alignment after unlearning (TOFU, Llama-3.2-3B). $|\Delta|/\text{s.e.}$ is the difference over its standard error; IFEval instruction-level scores report no standard error.

task	SimNPO	SimNPO+CL	Δ	$\lvert\Delta\rvert/\text{s.e.}$
MMLU	0.5898	0.5931	$+0.003$	0.59
GSM8K (exact match)	0.6444	0.6543	$+0.010$	0.53
HumanEval (pass@1)	0.4512	0.4146	-0.037	0.67
MBPP (pass@1)	0.4480	0.4420	-0.006	0.19
TruthfulQA (mc2)	0.4394	0.4492	$+0.010$	0.44
ARC-Challenge (acc)	0.4334	0.4479	$+0.015$	0.71
ARC-Challenge (acc_norm)	0.4420	0.4650	$+0.023$	1.12
HellaSwag (acc)	0.5356	0.5419	$+0.006$	0.89
HellaSwag (acc_norm)	0.7141	0.7180	$+0.004$	0.61
IFEval (prompt, strict)	0.4824	0.4861	$+0.004$	0.12
IFEval (prompt, loose)	0.5749	0.5545	-0.020	0.67
IFEval (inst., strict)	0.6283	0.6235	-0.005	–
IFEval (inst., loose)	0.6906	0.6799	-0.011	–

A.8.5 Forget-Set Output Quality

We generate answers to all 400 forget-set questions and measure whether the model refuses (pattern-matched refusal phrases), how coherent the output is (the probability of the "clean" class under the `madhurjindal/autonlp-Gibberish-Detector-492513457` classifier), and how repetitive it is (share of the most frequent 4-gram). No configuration ever refuses: the refusal rate is 0.000 for the base method and for every CLReg setting, so the failure mode on forget queries is degeneration rather than a polite decline; converting it into explicit refusals would require IDK-style forget targets. Because λ is the strength of the regularizer, we sweep it to determine whether the degeneration is

inherent to CLReg or a property of the operating point (Tab. 14). Coherence is governed by the operating point: unlearning quality rises quickly and then plateaus, while coherence is preserved at low λ and degrades as the term strengthens. At $\lambda = 0.3$, output is *more* coherent than the base method's, with the strongest relearning resistance and a gain of 0.078 in `UnlearningScore`. The best-scoring setting is not the most coherent one because `ForgetScore` rewards not reproducing the forget answer, which degenerate text also achieves; the metric is blind to output usability, so we report both axes. The gains at $\lambda = 1$ are nevertheless not an artifact of degeneration: they also appear on `PrivLeak` and in relearning resistance, neither of which incoherent output would produce. The degradation is confined to forget-set queries (Supp. A.8.4).

Table 14: Forget-set output quality along the regularization path (SimNPO+CL, TOFU Llama-3.2-3B; $\lambda = 0$ is SimNPO without CLReg). P(clean) is the gibberish classifier's probability of the "clean" class; recovery AUC is the attack of Supp. A.8.2 applied to each arm (seed 0).

λ	0	0.3	0.5	1.0	2.0
P(clean)↑	0.868	**0.887**	0.781	0.585	0.481
distinct-token ratio↑	**0.927**	0.860	0.613	0.380	0.280
max 4-gram share↓	**0.056**	0.076	0.228	0.474	0.618
refusal rate	0.000	0.000	0.000	0.000	0.000
`UnlearningScore`↑	0.688	0.766	0.781	**0.785**	0.780
recovery AUC↓	0.310	**0.212**	0.220	0.241	0.251

A.8.6 Model Size

Tab. 15 adds Llama-3.2-1B to the TOFU size range. CLReg helps at every size. The magnitude tracks how mis-calibrated the base method is rather than scale: at 3B the base method under-unlearns markedly ($|$`PrivLeak`$| = 61.7$) and CLReg corrects it, whereas at 1B and 8B the base methods are closer to balance and the headroom is smaller. The backbone is otherwise fixed by each benchmark's protocol: MUSE releases target and retrained checkpoints only for Llama-2-7B, and WMDP follows the RMU protocol on Zephyr-7B-β.

Table 15: Model-size ladder on TOFU (`UnlearningScore`, SimNPO vs. SimNPO+CL).

model	SimNPO	SimNPO+CL	Δ
Llama-3.2-1B	0.681	0.731	$+0.050$
Llama-3.2-3B	0.688	0.785	$+0.097$
Llama-3.1-8B	0.752	0.797	$+0.045$

A.8.7 When CLReg Degrades Performance

The largest regression in the main tables is the `PrivLeak` of GradDiff+CL on MUSE-News, which moves from -87.8 to $+99.7$. Because λ is the strength of the regularizer, sweeping it tells us whether this is an inherent failure of CLReg or the behavior at one operating point (Tab. 16). It is the operating point: `PrivLeak` changes sign between $\lambda = 0.1$ and 0.3, and retain quality is essentially untouched up to $\lambda = 0.3$. The trade-off is monotone and interpretable: forgetting depth rises with λ, retain quality falls, and privacy balance passes through zero, along the same axis that governs the number of negatives and τ (Supp. A.8.1). The composite score is not monotone: $\lambda = 2$ drops it below the no-CLReg baseline, so $\lambda = 1$ lies near the upper end of the useful range. We keep $\lambda = 1$ as the default, since it needs no per-setting tuning, underlies every result in the main tables, and is required under repeated unlearning requests (Supp. A.8.3); lowering it towards 0.2–0.3 is appropriate where privacy balance or forget-set output coherence matters more than forgetting depth.

A.9 Computational Overhead

CLReg introduces only *one extra forward pass* on the paraphrased input per iteration; dropout-based augmentation is applied on cached hidden states at negligible additional cost, and no extra trainable parameters are introduced. The contrastive term can be computed on the already-pooled,

Table 16: λ sweep for GradDiff+CL on MUSE-News. The $\lambda = 1$ column reports the run in Tab. 2b; the other columns are re-runs from the public checkpoints.

λ	0 (no CL)	0.1	0.3	0.5	1.0	2.0
ForgetScore↑	0.552	0.553	0.693	0.758	0.898	0.916
RetainKnowMem↑	0.467	0.469	0.466	0.455	0.410	0.326
UnlearningScore↑	0.506	0.508	0.557	**0.569**	0.563	0.481
PrivLeak→ 0	−87.6	−78.4	**+25.8**	+89.0	+99.7	+91.5

ℓ_2-normalized embeddings. Tab. 17 reports wall-clock runtimes (seconds) for 10 unlearning epochs on TOFU for each base method and its CLReg-augmented counterpart, measured on the same H100 GPU. Across most methods the overhead is an additional ∼64 s (3B) / ∼72 s (8B), i.e. a relative

Table 17: Wall-clock runtime (seconds) for 10 unlearning epochs on TOFU, with and without CLReg. Overhead is dominated by one extra forward pass on the paraphrased input; InfoNCE and DPO-CL have virtually identical runtime.

	GradDiff	+CL	UnDIAL	+CL	NPO	+CL	SimNPO	+CL	PDU	+CL
Llama-3-3B	131	195	147	270	189	252	157	221	130	193
Llama-3-8B	150	222	169	313	212	284	175	249	152	223

cost of 34–49%. UnDIAL+CL has a larger relative overhead, which is an implementation detail – UnDIAL's hidden states are not natively cached in our code path, requiring an extra full forward pass – rather than a fundamental cost of CLReg. Because MUSE has no paraphrases, dropout alone provides the contrastive signal there (Tab. 2), and the extra paraphrase-forward cost is avoided. CLReg thus trades a modest, predictable compute overhead for consistent improvements across methods and datasets, and does *not* require larger batch sizes or extra parameters.

A.10 Limitations and Future Work

When should CLReg be used, and when might it fail? CLReg is most useful when (i) the base method suffers from under-unlearning or tangled forget/retain features (large |PrivLeak|, low ForgetScore, high entanglement as measured in Tabs. 3–4), (ii) the forget concept is semantic and later-layer-localized, and (iii) retain-side utility must be preserved. Failure modes include: (a) extremely small forget sets where positive-pair construction is unstable, (b) base methods whose objectives already aggressively collapse forget representations – CLReg can push further and lead to over-unlearning (e.g., positive PrivLeak for GradDiff+CL on MUSE-News), which Supp. A.8.7 traces to the fixed $\lambda = 1$ rather than to the method, and (c) forget concepts encoded in very early layers, for which the default later-layer choice (Sec. 4.6) is mismatched. Mitigations include lowering λ (whose effect on forgetting depth, retain quality, privacy balance and output coherence is characterized in Supp. A.8.7 and A.8.5), reducing τ, or moving the CL target layer.

Limitations. *(i) Design sensitivity*: contrastive objectives hinge on choices of augmentation, batch size, and negatives, so CLReg may require dataset-specific tuning (cf. Supp. A.7 and the per-method settings in Supp. A.3). *(ii) Base-method dependence*: as a regularizer, CLReg's efficacy depends on the base unlearning loss and the constructiveness of its interaction with the contrastive term (Tabs. 1–2). *(iii) Light, one-step theory*: the analysis in Sec. 3.3 is a representation-level, one-step argument; it does not establish multi-step convergence, retain-side distortion bounds, or formal memorization-removal / privacy guarantees – all of which remain open for *every* approximate LLM unlearning method we are aware of (NPO, SimNPO, GradDiff, UnDIAL, PDU, RMU). *(iv) Sequential unlearning*: we study one sequential setting (four MUSE-News requests, Supp. A.8.3), where CLReg sharply reduces utility erosion but its per-request forgetting weakens along the chain; longer request streams and other benchmarks remain open. *(v) Recovery*: CLReg slows relearning rather than preventing it, and recovery curves with and without CLReg converge within 64 attack steps (Supp. A.8.2); like other approximate methods, it does not guarantee removal. *(vi) Seeds*: the main tables report single runs because of the cost of LLM unlearning; we quantify seed variance on a representative pair (Tab. 11), where the CLReg gain is about thirty times the seed-to-seed standard deviation.

A.11 Broader Impacts

Machine unlearning supports privacy and copyright compliance and the removal of hazardous knowledge, as in WMDP, and CLReg aims to make such removal more effective and more durable. Its main risk is a false sense of security: like every approximate unlearning method we are aware of, CLReg offers no formal removal guarantee, and our relearning attack shows that it slows rather than prevents recovery. Unlearned models should therefore not be treated as equivalent to models never trained on the data, and should be evaluated against adversarial recovery before deployment. Unlearning can also be used to suppress legitimate information; CLReg does not change who makes that choice or how it should be governed.

Future work. (i) *Surgical removal or clipping* of the disentangled forget subspace after CLReg training, building on the clear separation visible in Fig. 1 to achieve more faithful unlearning. (ii) *Richer augmentations and negative sampling* beyond dropout and paraphrase. (iii) *Concept-dependent layer selection*: our default later-layer regularization (Sec. 4.6) is a heuristic suited to current benchmarks; low-level (syntactic, stylistic) forget concepts may reside earlier, and probing or attribution could pick the right layer per concept. (iv) *Formal privacy and fairness guarantees* for representation-shaped unlearning, and study of CLReg under repeated, incremental, or continual-learning settings.

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: See listed contributions in Introduction and check with Method and Experiment Sections.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: See Supp. A.10.

   Guidelines:

   - The answer [N/A] 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: The paper contains a light, one-step theory. Each result states its assumptions (Sec. 3.3), complete proofs are in Supp. A.1, including the multi-negative form (Prop. A.1), and the one non-automatic condition of that form is verified empirically during training.

Guidelines:

- The answer [N/A] 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: Detailed experiment setup discussed in Experiment and Supplementary sections.

   Guidelines:

   - The answer [N/A] 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: All data used in this work is open-source, and our code is available at `https://github.com/HaoranTang/CLReg`.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Detailed experiment setup discussed in Experiment and Supplementary sections.

   Guidelines:

   - The answer [N/A] 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: [Yes]

   Justification: The main tables report single runs because of the cost of LLM unlearning. We quantify seed variance over three seeds on a representative pair (Tab. 11): the CLReg gain in `UnlearningScore` is $+0.100 \pm 0.003$, about thirty times the seed-to-seed standard deviation, and the relearning effect is -0.101 ± 0.032, negative in every seed. Capability results report standard errors (Tab. 13).

   Guidelines:

   - The answer [N/A] 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: See Supp. A.3.

   Guidelines:

   - The answer [N/A] 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: Followed and checked.

   Guidelines:

   - The answer [N/A] 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: See Supp. A.11. Unlearning supports privacy and copyright compliance and hazardous-knowledge removal; its main risk is a false sense of security, since approximate unlearning, ours included, slows rather than prevents recovery.

    Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] 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: Data is open-source, code is based on cited work and will be released, previous related work properly cited.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: The paper does not release new assets, just new code which will be released.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: An LLM (Llama-3.1-8B-Instruct) generates the paraphrase positives that CLReg uses on TOFU; the generator, prompt, sampling parameters and usage are described in Supp. A.4. Otherwise, LLMs are the subject of study rather than tools in the methodology.

    Guidelines:

    - The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.