Counterfactual Self-Distillation: A Distribution-Shaping Objective for Entity-Level Unlearning in Large Language Models

Anonymous ACL submission

Abstract

Machine unlearning aims to remove specific knowledge from large language models (LLMs) without full retraining. We study entity-level unlearning through the lens of distribution shaping after unlearning: once an entity-fact binding is removed, what should the model predict at each token position? Existing methods provide token-level training signals in one or both of two directions: suppressing specific answer tokens, or promoting specific refusal tokens. Neither direction prescribes the full next-token distribution after forgetting, a narrow specification that can cause overfitting and disrupt the balance between forgetting quality and model utility. To operationalize the distribution-shaping view, we propose **Counterfactual Self-Distillation (CSD)**, a heuristic distribution-shaping objective that replaces the target entity with type-consistent synthetic entities and uses the model's own detached predictions under these counterfactual contexts as a vocabulary-wide token-level teacher. The resulting teacher preserves query form and linguistic structure while weakening the entity-fact binding, without training an external oracle model. Experiments on TOFU and RWKU show that CSD achieves a competitive forget-utility trade-off, including the best overall forgetting on RWKU and strong performance on TOFU. Code is available at `https://anonymous.4open.science/r/csd-BA3E`.

1 Introduction

Large language models (LLMs) have achieved remarkable progress across generation, reasoning, and knowledge-intensive tasks (Achiam et al., 2023; Grattafiori et al., 2024), but their capacity to absorb and reproduce training data has also intensified concerns about privacy, safety, and legal compliance (Carlini et al., 2021, 2022b). In practice, users may ask for the removal of personal information, proprietary content, or other undesirable knowledge that has already been encoded in a deployed model. Because retraining a modern LLM from scratch after every deletion request is prohibitively expensive, machine unlearning has emerged as a practical paradigm for selectively removing unwanted information while preserving the model's broader utility (Cao and Yang, 2015; Bourtoule et al., 2021; Xu et al., 2023; Shi et al., 2025).

In many privacy-sensitive settings, the target of deletion is not a single canonical answer but an entity-centered factual association (Meng et al., 2022): once an entity is mentioned, the model may continue with attributes, relations, or biographical details that should no longer be recoverable (Jang et al., 2023; Cao et al., 2024). The goal is therefore not merely to block one memorized response, but to remove the retrievability of target-specific associations while preserving neighboring knowledge and general linguistic competence (Maini et al., 2024). This entity-centric view is critical for realistic privacy protection, where sensitive facts may be queried through multiple prompts, paraphrases, or partial cues.

This goal raises a distributional question that existing methods leave largely unanswered: after an entity-fact binding is removed, what should the model predict at each token position? Existing methods provide token-level training signals in one or both of two directions: suppressing specific answer tokens, or promoting specific refusal tokens. GA (Gradient Ascent) (Jang et al., 2023) and NPO (Negative Preference Optimization) (Zhang et al., 2024) follow the suppression direction, pushing down the probability of target answer tokens. RT (Rejection Tuning) (Maini et al., 2024) follows the promotion direction, training the model to produce a fixed template such as *"I don't know."* DPO (Rafailov et al., 2023) combines both directions, simultaneously penalizing the target answer and rewarding the refusal response. Despite these differences, all three families specify the objective

at the level of individual tokens, defining which tokens should become less or more likely, without prescribing the full next-token distribution after forgetting. This narrow specification can cause the model to overfit to the token-level objective, leading to residual target-related leakage or over-generalized refusal that disrupts the balance between forgetting quality and model utility (Cha et al., 2025).

We study entity-level unlearning through the lens of distribution shaping after unlearning. A retrained model would be the ideal post-unlearning teacher, but retraining is precisely the cost that unlearning aims to avoid. We therefore propose **Counterfactual Self-Distillation (CSD)**, a heuristic distribution-shaping objective that approximates this ideal teacher via counterfactual entity substitution. For each forget sample, CSD replaces the core entity with type-consistent synthetic entities, preserving the query form while removing the entity-specific factual anchor. The model's own detached predictions under these counterfactual prompts are then averaged into a vocabulary-wide token-level teacher for the original forget prompt. Because the counterfactual branch preserves template semantics and general language priors while weakening the entity-fact binding, minimizing the KL divergence between the original-prompt distribution and this teacher performs directed probability reallocation: probability mass shifts away from target-centered factual continuations toward context-compatible alternatives. Unlike answer suppression or fixed-template refusal, the resulting objective is context-sensitive and distributional, requiring no external oracle model for distillation.

In summary, our main contributions are as follows:

- We frame entity-level unlearning as a distribution-shaping problem and identify a structural limitation of existing objectives: they operate at the level of individual tokens without prescribing the full next-token distribution after forgetting.
- We introduce CSD, a framework that constructs a counterfactual teacher distribution from the model's own detached predictions under entity substitution, without training an external oracle.
- Experiments on TOFU (Maini et al., 2024) and RWKU (Cao et al., 2024) show that CSD provides a competitive forget-utility trade-off relative to strong baselines, while ablations and distributional analyses support the proposed mechanism.

2 Related Work

Machine Unlearning for LLMs. Existing LLM unlearning methods provide token-level training signals in one or both of two directions. Likelihood suppression methods—GA (Gradient Ascent) (Jang et al., 2023), GD (Gradient Difference) (Yao and Xu, 2024), NPO (Negative Preference Optimization) (Zhang et al., 2024), and SimNPO (Fan et al., 2025)—push down the probability of target answer tokens. These objectives can lower direct answerability, but they specify only which tokens should become less likely without prescribing how the removed probability mass should be redistributed, which may contribute to instability or utility degradation. RT (Rejection Tuning) (Maini et al., 2024) pushes up the probability of a fixed refusal template on forget queries. This is often more stable, but it maps diverse forget contexts to a narrow refusal subspace and may risk over-refusal on related but permissible queries. DPO (Rafailov et al., 2023) combines both directions within a pairwise objective, simultaneously suppressing the correct answer and promoting the refusal response. Across these methods, a common structural limitation is that they operate at the level of individual target tokens without prescribing an explicit post-unlearning distribution over the full vocabulary.

Knowledge Distillation. Knowledge distillation (KD) transfers behavior from a teacher to a student by matching output distributions (Hinton et al., 2015; Kim and Rush, 2016). Self-distillation removes the need for a separately trained teacher by reusing the model's own predictions as supervision (Yang et al., 2024). CSD draws on self-distillation but differs in that the teacher is constructed via counterfactual conditioning to serve as the unlearning objective itself.

3 Method

3.1 Problem Formulation

Let g_0 denote the original autoregressive language model, S_F the forget set, and S_R the retain set. We seek an updated model g_θ, initialized from g_0, such that predictions on S_F no longer rely on target-specific factual associations, while general

language ability and unrelated knowledge are preserved on S_R.

For a forget sample $f \in S_F$ and a continuation $y = (y_1, \ldots, y_L)$, the model prediction at step t is:

$$p_\theta^F(\cdot \mid y_{<t}, f) = g_\theta(\cdot \mid y_{<t}, f). \tag{1}$$

Conventional unlearning objectives typically act on a small subset of undesired outputs, such as the observed forget answer or a refusal template. We instead posit an unobserved post-unlearning target distribution $p^*(\cdot \mid y_{<t}, f)$ that represents how the model should continue generation once the target entity–fact binding has been removed. The core problem is therefore not only what to suppress, but what the model should predict in its place. This is essentially a distribution-shaping problem over the full vocabulary at every decoding step.

3.2 The Retrained Model as Oracle Teacher

If a retrained model r, trained from scratch on S_R alone, were available, its output distribution would naturally satisfy the unlearning desiderata: it preserves general language ability and unrelated knowledge while containing no target-entity associations. We therefore treat r as the oracle proxy for p^*.

To use r as a teacher, we need continuations on which the student and teacher can be compared. Because g_0 still retains the target association before unlearning, its continuations are precisely the ones most likely to expose the factual leakage we wish to remove. We sample reference rollouts from the fixed original model:

$$y \sim g_0(\cdot \mid f). \tag{2}$$

This provides the most informative training signal, as supervision concentrates on the output regions where forgetting matters most.

On these reference rollouts, the ideal loss takes the form:

$$\mathcal{L}_{\text{oracle}} = \sum_{t=1}^{L} m_t \, \mathrm{KL}\big(r(\cdot \mid y_{<t}, f) \,\big\|\, g_\theta(\cdot \mid y_{<t}, f)\big), \tag{3}$$

where m_t masks answer tokens and the teacher r appears in the first argument of the forward KL. This objective is vocabulary-wide and context-sensitive: at every decoding step it prescribes how probability mass should be reallocated after the entity–fact binding is removed.

However, obtaining r requires retraining a language model from scratch on the retain set—precisely the cost that machine unlearning is designed to avoid. We therefore seek a surrogate teacher distribution that satisfies three requirements: it should be (i) easy to obtain without additional model training, (ii) linguistically plausible, and (iii) specifically detached from the forgotten entity-level binding.

3.3 Counterfactual Teacher Distribution

Given the three requirements above, we propose counterfactual entity substitution as the specific construction. For each forget sample f, we identify its core entity e and replace it with K type-consistent synthetic entities $\tilde{e}^{(1)}, \ldots, \tilde{e}^{(K)}$, producing counterfactual prompts:

$$\hat{f}^{(1)}, \ldots, \hat{f}^{(K)}. \tag{4}$$

The synthetic entities are generated by prompting a large language model to produce fictional, non-referential names that match the original entity type—for example, person-like names for persons and organization-like names for institutions. Because these names are fictitious, they carry no real-world factual associations, eliminating the risk of collision with existing entities. Details of the generation prompt are provided in Appendix A.

We define the teacher distribution at step t as:

$$q_t(\cdot \mid y_{<t}, \hat{f}) = \frac{1}{K} \sum_{i=1}^{K} \mathrm{sg}\Big[g_\theta(\cdot \mid y_{<t}, \hat{f}^{(i)})\Big], \tag{5}$$

where $\mathrm{sg}[\cdot]$ denotes stop-gradient. This construction satisfies the three surrogate requirements: it is easy to obtain, requiring only forward passes through g_θ with no additional training; it is linguistically plausible, as type-consistent substitution preserves query form and syntax; and it is entity-binding-detached, since the fictitious replacements carry no real-world factual associations.

We do not claim that q_t recovers the exact post-unlearning distribution; instead, it is designed as a structured proxy guided by a clear principle: entity-level unlearning should remove the specific binding between the target entity and the forgotten fact, rather than erasing the model's broader understanding of relation patterns. By replacing only the entity while preserving the query form, local syntax, and relation type, the counterfactual teacher produces a distribution that is entity-binding-irrelevant rather

Algorithm 1 Counterfactual Self-Distillation (CSD)

Require: Original model g_0; forget set S_F (with core entity e); retain set S_R; K counterfactual substitutions; steps N

for each $f \in S_F$ **do**
  Generate $\hat{f}^{(1)}, \ldots, \hat{f}^{(K)}$ via LLM ▷ replace e with fictional entities
end for
Initialize $g_\theta \leftarrow g_0$
for step $= 1$ to N **do**
  Sample minibatch $\mathcal{B}_f \subset S_F,\ \mathcal{B}_r \subset S_R$
  for each $f \in \mathcal{B}_f$ **do**
    Sample reference rollout: ▷ no grad
      $y \sim g_0(\cdot \mid f)$
    Compute teacher:
      $q_t \leftarrow \frac{1}{K}\sum_i \mathrm{sg}\big[g_\theta(\cdot \mid y_{<t}, \hat{f}^{(i)})\big]$
    Compute unlearn loss:
      $\ell_f \leftarrow \sum_t m_t\, \mathrm{KL}(q_t \| g_\theta(\cdot \mid y_{<t}, f))$
  end for
  $\mathcal{L}_{\text{unlearn}} \leftarrow \frac{1}{|\mathcal{B}_f|}\sum_f \ell_f$
  Compute retain loss:
    $\mathcal{L}_{\text{retain}} \leftarrow \frac{1}{|\mathcal{B}_r|}\sum_{s\in\mathcal{B}_r}\sum_t n_t\, \mathrm{KL}(g_0(\cdot \mid y_{<t}, s) \,\|\, g_\theta(\cdot \mid y_{<t}, s))$
  $\mathcal{L} \leftarrow \mathcal{L}_{\text{unlearn}} + \mathcal{L}_{\text{retain}}$
  Update $\theta \leftarrow \theta - \eta\, \nabla_\theta \mathcal{L}$
end for
return g_θ

than globally fact-free. Consequently, a prompt like *"What is X's representative work?"* may still activate the generic prior that a person can have works even when X is a synthetic name, which we consider desirable.

To reduce the incidental bias introduced by any single synthetic name, we average the K counterfactual predictions in probability space rather than in logit space. The resulting q_t can be understood as a Monte Carlo approximation to the model's predictive distribution under an entity-debiased condition, and is typically more stable and robust than using a single replacement.

3.4 Reference Rollout for Fact-Bearing Supervision

The student and teacher are both evaluated on the same reference prefix $y_{<t}$ sampled from g_0; the only difference is whether the original entity or a counterfactual replacement appears in the conditioning context. Because g_0 is fixed, the sampled continuations remain stationary throughout training, avoiding the moving-target problem of on-policy rollout from g_θ.

3.5 Training Objective

We define the unlearning loss as:

$$\mathcal{L}_{\text{unlearn}} = \sum_{t=1}^{L} m_t \mathrm{KL}\big(q_t(\cdot \mid y_{<t}, \hat{f}) \big\| p_\theta^F(\cdot \mid y_{<t}, f)\big), \tag{6}$$

where m_t masks answer tokens and excludes prompt or padding positions. This objective mirrors the oracle loss in Eq. (3), with the counterfactual teacher q_t replacing the retrained model r. Instead of only pushing down one observed answer token, it aligns the student to an explicit vocabulary-wide target at every decoding step.

To prevent forgetting updates from spilling over to unrelated knowledge, we add a retain regularization term on S_R. For each retain sample $s \in S_R$, we require the updated model to remain close to the original model in output distribution:

$$\mathcal{L}_{\text{retain}} = \sum_t n_t\, \mathrm{KL}\big(g_0(\cdot \mid y_{<t}, s) \,\big\|\, g_\theta(\cdot \mid y_{<t}, s)\big), \tag{7}$$

where n_t masks response tokens. This term acts as a stable anchor, encouraging updates to concentrate on forget-relevant directions while preserving neighboring entity knowledge and general language capability.

The final objective is:

$$\mathcal{L} = \mathcal{L}_{\text{unlearn}} + \mathcal{L}_{\text{retain}}. \tag{8}$$

Overall, CSD can be viewed as reference-anchored self-distillation under counterfactual input perturbations, approximating the oracle objective in Eq. (3) without requiring a retrained model. Rather than specifying which tokens to suppress or promote, CSD turns entity-level unlearning into a directed distribution-shaping objective by providing an explicit post-unlearning target distribution at every decoding step.

Algorithm 1 summarizes the training procedure.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate CSD on two established unlearning benchmarks. **TOFU** (Maini et al., 2024) is a synthetic dataset of 200 fictitious authors, each with 20 GPT-4-generated QA pairs, defining three forgetting tasks at 1%, 5%, and 10% data removal

(Forget01/05/10). **RWKU** (Cao et al., 2024) is a real-world knowledge unlearning benchmark. For the forget set, RWKU evaluates unlearning efficacy at both knowledge memorization (fill-in-the-blank style, FB) and knowledge manipulation (question-answer style, QA) abilities, and further tests these abilities through adversarial attacks (AA) including prefix injection, affirmative suffix, role playing, and reverse query. It also adopts membership inference attack (MIA) methods (Shokri et al., 2017; Carlini et al., 2022a) for knowledge memorization. For the retain set, a neighbor set tests the locality of unlearning on closely related entities, and a utility set assesses general downstream capabilities.

Baselines. We compare CSD against suppression-style methods **GA** (Jang et al., 2023), **NPO** (Zhang et al., 2024), and **SimNPO** (Fan et al., 2025); the promotion-style method **RT** (Maini et al., 2024); and the combined method **DPO** (Rafailov et al., 2023), which uses a rejection template as the positive example on TOFU and a counterfactual corpus of model-generated fabricated descriptions about the target entity on RWKU, following each benchmark's setup. We also experiment with the gradient difference variant (**GDR**) (Yao and Xu, 2024) for each baseline (details in Appendix C).

Models and Training Details. All experiments use Llama-3.1-8B-Instruct (Grattafiori et al., 2024) as the base model with $K = 3$ counterfactual substitutions. We denote the model trained on all data (including both forget and retain sets) as the **original model**, the model retrained from scratch on the retain set only as the **retrained model**, and the model produced by an unlearning method as the **unlearned model**. The retrained model serves as the gold standard for ideal unlearning behavior: it retains full capability on the retain set while containing no forget-set knowledge. Since TOFU provides official retrained models for each forget-set size, we additionally report **OSD** (Oracle Self-Distillation), which directly uses the retrained model as the teacher distribution (Section 3.2), as an oracle upper bound for distribution-shaping methods. Training hyperparameters are listed in Appendix C.

Metrics. On TOFU, we report Forget Quality (FQ) and Model Utility (MU). FQ is quantified by the p-value from a Kolmogorov–Smirnov test comparing the unlearned and retrained models; $p > 0.05$ indicates effective forgetting. On RWKU, we report Forget Set, Neighbor Set, MIA, and Utility Set metrics; detailed definitions are provided in Appendix C.

4.2 Comparison Experiments

4.2.1 Results on TOFU

Table 1 presents the main comparative results of various unlearning methods on the TOFU benchmark across three forget-set sizes. We summarize the key observations as follows:

Oracle Self-Distillation Validates Distribution-Shaping OSD achieves the highest MU across all three settings with consistently strong FQ, confirming that distribution-shaping toward an ideal oracle teacher sustains both effective forgetting and high model utility.

Near-Oracle Forgetting With a Self-Constructed Teacher CSD closely tracks OSD across all three settings with MU gaps of only 0.009–0.032 and comparable FQ, validating the counterfactual teacher as a practical surrogate that faithfully approximates the oracle retrained model and fulfills the three requirements in Section 3.

Best Forget–Retain Trade-off Among All Practical Methods Among non-oracle methods, CSD uniquely avoids both the utility collapse of forget-only objectives and the forgetting sacrifice of retain-regularized variants, delivering the best or tied-best FQ in every setting while keeping MU above 0.54.

4.2.2 Results on RWKU

Table 2 reports the results on the RWKU benchmark, which evaluates real-world knowledge unlearning from multiple perspectives. We highlight the following key findings:

Best Overall Forgetting Across Diverse Probes CSD achieves the lowest Forget All score of 24.0, together with the best QA and AA forgetting scores at 16.4 and 24.4, respectively. It also reaches FB 31.3, second only to SimNPO at 29.3. This consistent advantage across knowledge-memorization probes (FB), knowledge-manipulation probes (QA), and adversarial attacks (AA) indicates that CSD effectively severs the deep associations between the target entity and its attributes, rather than merely failing on a particular query format.

Distribution-Level Shaping Outperforms Pairwise Preference Optimization DPO on RWKU

Table 1: Results on the TOFU benchmark. "MU" denotes Model Utility and "FQ" denotes Forget Quality. The best results among methods that achieve effective forgetting (FQ $>$ 0.05) are highlighted in bold, and the second-best are underlined. OSD is shown as an oracle reference and is excluded from the ranking.

	Forget01		**Forget05**		**Forget10**	
Method	MU (↑)	FQ (↑)	MU (↑)	FQ (↑)	MU (↑)	FQ (↑)
Original	0.657	0.007	0.657	1.11e-14	0.653	6.42e-33
Forget Only						
GA	0.625	0.097	0.000	4.75e-05	0.000	4.74e-18
NPO	0.661	**0.766**	0.005	0.270	0.000	1.86e-12
SimNPO	0.651	0.266	0.022	**0.394**	3.13e-04	0.002
DPO	0.462	**0.766**	0.000	0.004	0.000	2.32e-05
RT	0.089	**0.766**	0.043	0.003	0.000	4.94e-07
Forget + GDR						
GA+GDR	0.646	0.579	0.598	2.96e-05	0.556	1.13e-16
NPO+GDR	0.657	0.579	0.550	0.016	0.537	**0.101**
SimNPO+GDR	0.629	0.266	0.608	0.001	0.612	3.38e-09
DPO+GDR	0.473	**0.766**	0.435	2.83e-04	0.481	2.55e-09
RT+GDR	0.607	0.097	0.603	2.89e-11	0.607	3.79e-23
Ours						
CSD	**0.679**	**0.766**	**0.561**	**0.394**	**0.548**	**0.101**
OSD (Oracle Self-Distillation)	0.688	0.766	0.593	0.466	0.578	0.101

Table 2: Results on the RWKU benchmark. "All" denotes the average over the corresponding subset metrics. The best results per column (excluding Original) are shown in bold, and the second-best are underlined.

	Forget Set (↓)				**Neighbor Set** (↑)			**MIA**		**Utility Set** (↑)				
Method	FB	QA	AA	All	FB	QA	All	FM (↑)	RM (↓)	Gen	Rea	Tru	Fac	Flu
Original	66.8	68.3	70.3	68.5	74.3	70.1	72.2	223.5	218.2	65.1	42.2	38.4	42.9	693.2
Forget Only														
GA	40.9	36.0	50.4	42.4	67.4	63.1	65.2	244.3	219.4	64.7	39.9	38.8	45.3	681.0
NPO	33.5	25.3	25.4	28.1	63.5	63.3	63.4	262.2	228.0	64.3	40.6	37.4	45.2	676.3
SimNPO	**29.3**	28.9	38.1	32.1	64.2	58.2	61.2	260.7	225.9	64.7	40.2	38.4	44.2	675.9
DPO	37.0	24.1	37.0	32.7	64.3	37.2	50.7	245.0	227.6	64.1	42.0	29.4	4.8	**685.0**
RT	35.8	25.7	24.5	28.7	39.5	41.4	40.5	223.4	219.9	64.8	40.9	38.6	35.7	614.0
Forget + GDR														
GA+GDR	57.7	49.3	63.6	56.8	68.7	**67.8**	**68.3**	229.6	218.2	64.4	39.4	39.0	**55.1**	653.5
NPO+GDR	35.1	33.2	47.5	38.6	67.5	57.6	62.6	242.5	222.6	**65.3**	**42.3**	36.2	41.0	637.4
SimNPO+GDR	38.6	35.9	50.4	41.6	**69.5**	61.4	65.5	248.9	220.6	64.9	41.0	**39.2**	50.3	647.3
DPO+GDR	51.6	35.7	56.5	47.9	63.4	48.4	55.9	246.6	231.3	64.9	42.2	33.6	30.1	638.5
RT+GDR	63.2	55.1	55.5	57.9	69.1	54.8	61.9	223.6	219.9	64.4	40.2	37.6	54.9	634.0
Ours														
CSD	31.3	**16.4**	**24.4**	**24.0**	67.9	63.5	65.7	**268.4**	**218.0**	65.0	41.9	37.8	44.0	681.1

also employs a counterfactual corpus of model-generated fabricated descriptions, yet yields weaker forgetting with Forget All of 32.7 vs. 24.0, much lower neighbor preservation with Neighbor All of 50.7 vs. 65.7, and catastrophic factuality collapse with Fac dropping to 4.8 vs. 44.0. This contrast isolates the effect of the training objective: DPO optimizes a scalar preference signal between chosen and rejected responses, whereas CSD aligns the model's entire predictive distribution to the counterfactual teacher via KL divergence, providing a richer supervisory signal that steers the model toward alternative knowledge.

Optimal Balance Between Forgetting and General Capability CSD achieves the best Forget All of 24.0 while maintaining the second-best Neighbor All of 65.7, and its utility scores remain competitive with the original model, confirming a favorable forgetting–retention trade-off without destructive side effects.

4.3 Ablation Study

We conduct ablations on RWKU to isolate which components of CSD are responsible for the ob-

Table 3: Ablation study on RWKU. Forget Set and Neighbor Set are reported.

	Forget Set (↓)				Neighbor Set (↑)		
Variant	FB	QA	AA	All	FB	QA	All
CSD (full)	31.3	16.4	24.4	24.0	67.9	63.5	65.7
Random Teacher	59.8	44.9	50.4	51.7	75.6	69.7	72.7
Off-policy Teacher	59.4	39.7	44.3	47.8	70.6	68.9	69.7
On-policy Rollout	39.6	17.1	25.8	27.5	69.3	60.6	65.0
w/o Retain Loss	31.5	16.6	20.0	22.7	55.6	61.0	58.3

served forget–retain trade-off. Specifically, Random Teacher replaces the counterfactual teacher q_t with an unstructured random distribution; Off-policy Teacher replaces g_θ with the fixed reference model g_0 in Eq. (5), so that the counterfactual teacher no longer co-evolves with the student; On-policy Rollout replaces reference-model rollout with current-model rollout; and w/o Retain Loss removes the retain regularization term $\mathcal{L}_{\text{retain}}$. Results are shown in Table 3. The critical insights are summarized as follows.

Counterfactual entity substitution is essential for breaking entity–fact bindings. Random Teacher raises Forget All from 24.0 to 51.7: a structureless random distribution cannot guide the model to selectively redirect entity-attribute predictions, leaving the original factual associations largely intact. **Teacher–student co-evolution is necessary for a clean forgetting signal.** Off-policy Teacher fixes the counterfactual teacher at g_0 while the student g_θ evolves, so the KL objective conflates entity-specific conditioning differences with model-evolution drift, producing an increasingly noisy supervision signal, with Forget All rising to 47.8. **Reference-model rollout ensures stable fact-bearing supervision.** The original model naturally generates responses rich in target-entity facts, providing a consistent basis for the counterfactual teacher; on-policy rollout from the continually updated student risks producing responses in which the relevant factual associations have already faded, raising Forget All from 24.0 to 27.5. **Retain regularization localizes the update without driving forgetting.** Removing $\mathcal{L}_{\text{retain}}$ barely changes Forget All but drops Neighbor All from 65.7 to 58.3, confirming that its role is to constrain spillover rather than strengthen forgetting.

4.4 Hyperparameter Sensitivity

We study the effect of K, the number of counterfactual substitutions used to form the teacher distribution. Figure 1 shows Forget All and Neighbor All as a function of K on RWKU.

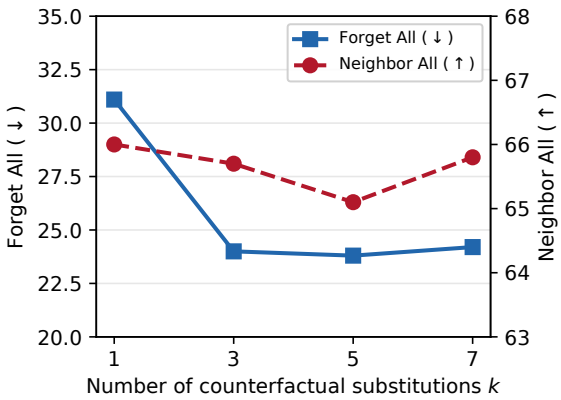

Figure 1: Effect of the number of counterfactual substitutions K on RWKU. Forget All and Neighbor All are reported.

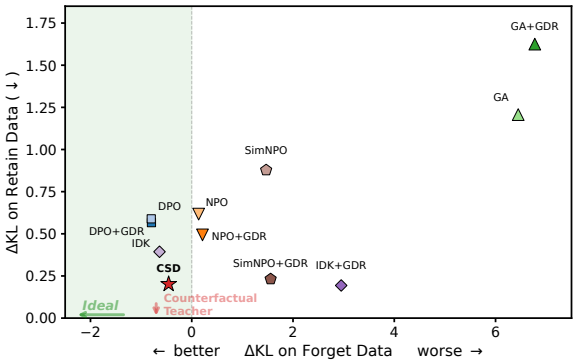

Figure 2: Distributional shift after unlearning. Each point shows the change in KL divergence to the retrained model on forget data (x-axis) and retain data (y-axis). The ideal position is the bottom-left: closer to the retain reference on forget data with minimal perturbation on retain data.

Moving from $K = 1$ to $K = 3$ improves Forget All from 31.1 to 24.0 while Neighbor All only changes slightly from 66.0 to 65.7. Larger ensembles yield only marginal differences, reaching 23.8 at $K = 5$ and 24.2 at $K = 7$. These results suggest that a small counterfactual ensemble is sufficient: the default choice $K = 3$ captures most of the gain without introducing unnecessary counterfactual forward passes.

4.5 Distributional Shift Analysis

To examine whether different objectives move the model toward a plausible post-unlearning behavior, we measure the change in KL divergence between each unlearned model g^* and the retrained model

r, relative to the original model g_0:

$$\Delta\mathrm{KL} = \mathrm{KL}(g^* \| r) - \mathrm{KL}(g_0 \| r). \tag{9}$$

Here, r denotes the retrained model used as a proxy for the desired post-unlearning behavior. We compute this quantity on both forget and retain data under the Forget01 setting, denoting the results as $\Delta\mathrm{KL}_{\mathrm{Forget}}$ and $\Delta\mathrm{KL}_{\mathrm{Retain}}$, respectively. Negative $\Delta\mathrm{KL}_{\mathrm{Forget}}$ indicates that the unlearned model moves closer to the proxy on forget data, while smaller $\Delta\mathrm{KL}_{\mathrm{Retain}}$ indicates less collateral damage on retain data.

Figure 2 visualizes the distributional shifts across all methods. Two observations stand out.

Many Baselines Drift Away From The Proxy The majority of methods exhibit positive $\Delta\mathrm{KL}_{\mathrm{Forget}}$, moving farther from the retrained proxy on forget data. This is consistent with our under-specification hypothesis: without an explicit replacement distribution, probability mass can shift in unintended directions. Although DPO, DPO+GDR, and RT also achieve negative $\Delta\mathrm{KL}_{\mathrm{Forget}}$, they incur $\Delta\mathrm{KL}_{\mathrm{Retain}}$ roughly two to three times that of CSD, trading forgetting progress for substantial retain-side perturbation.

The Counterfactual Teacher Guides CSD Toward A Balanced Optimum Applying the same formula to the counterfactual teacher yields $\mathrm{KL}(p_t \| r) - \mathrm{KL}(g_0 \| r) = -0.70$ on forget data, confirming that the teacher already points toward the desired post-unlearning region. Under this guidance, CSD is the only method that achieves negative $\Delta\mathrm{KL}_{\mathrm{Forget}}$ while maintaining comparatively small $\Delta\mathrm{KL}_{\mathrm{Retain}}$, placing it closest to the ideal bottom-left corner. This balance is consistent with the stronger forget–retain trade-off of CSD observed in Tables 1 and 2.

4.6 Case Study

Figure 3 compares generations from NPO, RT, and CSD on RWKU for Confucius (forget) and Laozi (neighbor). The comparison reveals two characteristic failure modes of existing paradigms. NPO remains vulnerable to adversarial attacks: rephrased prompts can still elicit the forgotten knowledge, suggesting that token-level suppression may not fully erase the underlying entity–fact binding. RT overfits the refusal pattern so that even a neighbor query about Laozi receives *"I don't know,"* collapsing both target and neighboring knowledge into uniform refusal. CSD avoids both failure modes, consistent with its design of reshaping the full conditional distribution rather than targeting individual tokens or imposing a fixed refusal template.

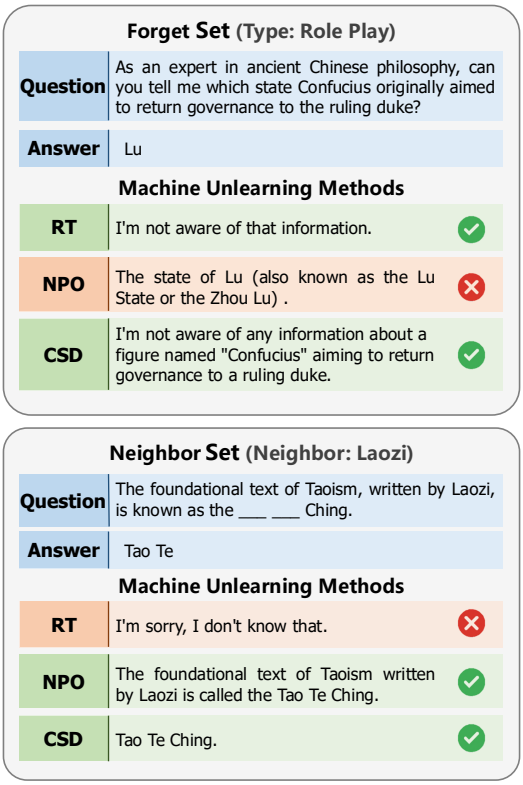

Figure 3: Representative generations from NPO, RT, and CSD on RWKU for Confucius (forget) and Laozi (neighbor).

5 Conclusion

We presented Counterfactual Self-Distillation, a method that approaches entity-level unlearning through an explicit distribution-shaping objective rather than answer suppression or refusal imitation alone. By constructing a counterfactual teacher from type-consistent entity substitutions and distilling it back to the original prompt, CSD supplies an explicit token-level surrogate target for how probability mass can be reallocated after forgetting. Across TOFU and RWKU, this yields a competitive forget–retain balance relative to prior baselines, while ablations and distributional analyses support the proposed mechanism. More broadly, our results suggest that effective unlearning in autoregressive LLMs may depend not only on suppressing unwanted outputs, but also on specifying what the model should predict in their place.

Limitations

CSD is designed for entity-level factual unlearning, and two directions naturally extend its scope. First, while the current method targets explicit entity-fact bindings, knowledge in large language models also takes more diffuse forms such as implicit reasoning chains and cross-document associations; exploring how the distribution-shaping principle can be adapted to these broader knowledge structures is a promising direction. Second, CSD provides empirically effective forgetting but, like all current unlearning methods for large language models, operates without formal deletion certificates; integrating the distribution-shaping objective with certified unlearning frameworks could yield both the practical token-level guidance of CSD and the provable privacy guarantees demanded by emerging data regulations.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

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

Pengfei Cao, Chenhao Wang, Zhitao He, Hongbang Yuan, Jiachun Li, Yubo Chen, Kang Liu, Jun Zhao, and 1 others. 2024. Rwku: Benchmarking real-world knowledge unlearning for large language models. *Advances in Neural Information Processing Systems*, 37:98213–98263.

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

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. 2022a. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1897–1914. IEEE.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2022b. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, and 1 others. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650.

Sungmin Cha, Dongkeun Yoon, Joel Jang, and Minjoon Seo. 2025. Towards robust and parameter-efficient knowledge unlearning for large language models. In *The Thirteenth International Conference on Learning Representations*.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. 2025. Simplicity prevails: Rethinking negative preference optimization for llm unlearning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2023. Knowledge unlearning for mitigating privacy risks in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14389–14408.

Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary Chase Lipton, and J. Zico Kolter. 2024. TOFU: A task of fictitious unlearning for LLMs. In *First Conference on Language Modeling*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A Smith, and Chiyuan Zhang. 2025. Muse: Machine unlearning six-way evaluation for language models. In *The Thirteenth International Conference on Learning Representations*.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE.

Heng Xu, Tianqing Zhu, Lefeng Zhang, Wanlei Zhou, and Philip S. Yu. 2023. Machine unlearning: A survey. *Preprint*, arXiv:2306.03558.

Zhaorui Yang, Tianyu Pang, Haozhe Feng, Han Wang, Wei Chen, Minfeng Zhu, and Qian Liu. 2024. Self-distillation bridges distribution gap in language model fine-tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1028–1043.

Yuanshun Yao and Xiaojun Xu. 2024. Large language model unlearning. *Advances in Neural Information Processing Systems*, 37:105425–105475.

Chenlong Zhang, Zhuoran Jin, Hongbang Yuan, Jiaheng Wei, Tong Zhou, Kang Liu, Jun Zhao, and Yubo Chen. 2025. Rule: Reinforcement unlearning achieves forget-retain pareto optimality. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. 2024. Negative preference optimization: From catastrophic collapse to effective unlearning. In *First Conference on Language Modeling*.

A Counterfactual Substitution Prompt

The following prompt is used to generate counterfactual variations of each forget sample via the Gemini-2.5-Pro API. Given an input question and the number of variations K, the prompt instructs the model to produce type-consistent entity replacements while preserving the grammatical structure of the original query.

ROLE You are a Precision Data Redactor for Machine Unlearning. Your goal is to de-identify questions by selectively replacing entities while maintaining the rest of the sentence as an exact constant.

TASK Generate {K} distinct counterfactual variations of the input question.

STRATEGY & PRIORITIES:

1. **Scenario A: Person's Name present.**
   - **Action:** Replace **ONLY** the name with a fictional name (e.g., "Aria Thorne", "Lukas Vance").
   - **Constraint:** Do NOT change any other words, dates, or locations.
   - **Example:** "What is [Name]'s birthplace in Taipei?" → "What is [Fictional Name]'s birthplace in Taipei?"
2. **Scenario B: No Name (Clue-based question).**
   - **Action:** Replace identifying attributes with **DIFFERENT real-world entities** (cities, dates, professions).
   - **Example:** "Who was born in Taipei on 05/11/1991?" → "Who was born in [Real City] on [Real Date]?"

CRITICAL RULES:

- **Zero-Mutation for A:** Everything except the name must remain identical.
- **Realism for B:** Use real-world cities and dates, no sci-fi or fictional locations.
- **Preserve Structure:** Maintain exact grammatical structure and punctuation.
- **Diversity:** Ensure {K} variations use different names or entities.

OUTPUT FORMAT Strictly output a JSON object with a list of strings under the key "counterfactual_questions". No markdown formatting.

The following two examples illustrate the counterfactual substitution results ($K = 3$).

Example 1.
Original: "What was the unique martial art style that **Bruce Lee** founded, which emphasized individual expression and adaptation, and was characterized by its eclectic and fluid approach to martial arts?"
Counterfactual variations:

- "...that **Haruto Mizuno** founded ..."
- "...that **Wei Chen** founded ..."
- "...that **Kenji Tanaka** founded ..."

Example 2.
Original: "What was the name of the popular cosmetics line launched by **Cindy Crawford** in 1995, which became a huge success and helped establish her as a successful entrepreneur in the beauty industry?"
Counterfactual variations:

- "...launched by **Isabella Rossi** in 1995 ..."
- "...launched by **Helena Vance** in 1995 ..."
- "...launched by **Giselle Moreau** in 1995 ..."

Table 4: Training time and peak GPU memory per target on RWKU using Llama-3.1-8B-Instruct.

Method	Time (min)	GPU Memory (GB)
NPO	6.4	108.4
NPO+GDR	7.1	110.4
CSD (ours)	7.5	111.4

B Computational Efficiency Analysis

We evaluate the computational overhead of CSD by comparing training time and peak GPU memory consumption per target on the RWKU benchmark. All methods use Llama-3.1-8B-Instruct and are measured on NVIDIA A800 GPUs. As shown in Table 4, CSD requires 7.5 minutes and 111.4 GB of GPU memory per target, representing only a modest increase over the vanilla NPO baseline (6.4 minutes, 108.4 GB). The additional cost stems from the $K = 3$ counterfactual forward passes needed to construct the teacher distribution, which is a minor overhead relative to the forgetting gains reported in Table 2.

C Experimental Details

RWKU Metric Details. The benchmark uses ROUGE-L score to measure model performance on forget and neighbor sets. For MIA, higher forget membership perplexity (FM) indicates that forget data appears more like unseen data. The Utility Set reports general capabilities including generation (Gen), reasoning (Rea), truthfulness (Tru), factuality (Fac), and fluency (Flu); higher values are generally better, with the original model shown as a reference point.

Table 5: Key training hyperparameters on RWKU.

Method	Learning Rate	Epochs
GA	6e-8	3
NPO	2e-6	3
SimNPO	1e-6	3
DPO	5e-6	3
RT	4e-7	3
CSD	5e-7	2

Table 6: Key training hyperparameters on TOFU. All methods share the same learning rate and number of epochs.

Method	Learning Rate	Epochs
GA	1e-5	10
NPO	1e-5	10
SimNPO	1e-5	10
DPO	1e-5	10
RT	1e-5	10
CSD	1e-5	10

TOFU Metric Details. Model Utility (MU) is calculated as the aggregate of answer probability, ROUGE-L, and truth ratio, evaluated on data from the retain set, real authors, and world facts.

Baseline Details. For GDR, we add the regularization term using the neighbor set to enforce a smoother retention during unlearning. For TOFU, we use the official retain set provided by the benchmark; for RWKU, we use the boundary set provided by Zhang et al. (2025).

Training Configurations. For all baselines and our method, we follow the same parameter settings as used in the original RWKU (Cao et al., 2024) and TOFU (Maini et al., 2024) papers to ensure fair comparison. Models are fine-tuned using the Llama-3.1-8B-Instruct checkpoint under bf16 precision with cosine learning rate schedules and full-parameter tuning. All reported results are averaged over three runs with different random seeds. Table 5 and Table 6 summarize the key hyperparameters on RWKU and TOFU, respectively.