REPLAY WHAT MATTERS: OFF-POLICY REPLAY FOR EFFICIENT LLM REINFORCEMENT UNLEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

LLM unlearning has emerged as a cost-effective alternative to full retraining for removing hazardous knowledge from pretrained models while preserving general utility. Recent RL-based methods such as RULE reformulate unlearning as learning a refusal behavior, but their on-policy optimization repeatedly samples from the same forget and retain/boundary prompts throughout training. We identify a critical inefficiency in this process: easy cases quickly converge and provide little useful gradient signal, while hard cases near the forget/retain boundary continue to produce low-reward rollouts that are discarded after a single use. To address this issue, we propose **ReRULE**, an off-policy replay enhancement for reinforcement unlearning. **ReRULE** stores low-reward hard-case rollout groups in a replay buffer during early GRPO training and reuses them in later stages through importance-sampled off-policy updates, redirecting computation toward boundary cases that require further learning. Theoretically, we show that **ReRULE** yields a tighter hard-case convergence bound than pure on-policy RULE. Empirically, **ReRULE** improves MUSE-Books Retain Quality from 46.3 to 56.2 while adding only 5–11% training time across benchmarks. Its limited improvement on the simpler TOFU setting further supports the intended conditional behavior: replay is most beneficial when the hard/easy disparity is pronounced.

1 INTRODUCTION

Recent advances in large language model (LLM) pretraining have significantly improved model capabilities (Yang et al., 2025; Team et al., 2023; Guo et al., 2025; Touvron et al., 2023; Zhang et al., 2022; Li et al., 2023). However, because these models are trained on large-scale internet corpora, they may acquire copyrighted content (Karamolegkou et al., 2023; Grynbaum & Mac, 2023; Chu et al., 2024; Zhang et al., 2024b;c), sensitive private information (Staab et al., 2024; Mireshghallah et al., 2024; Das et al., 2025; Di et al., 2025; Zheng et al., 2026), or other hazardous knowledge. Removing such data and retraining from scratch is prohibitively expensive, which hence motivates LLM unlearning (Yao & Xu, 2024): methods that remove undesired knowledge while preserving the model's general utility.

Reinforcement learning (RL) offers a natural route for unlearning because it can directly optimize refusal behavior. RULE (Zhang et al., 2025) follows this idea by reformulating unlearning as teaching the model to refuse queries related to the forget set while preserving responses on retain and boundary queries. Unlike SFT-based unlearning methods that process each sample once, RL-based methods such as RULE repeatedly sample rollouts from the same training prompts. Such repetition is useful for exploration at the beginning of training, but it becomes inefficient once many prompts have already converged.

We observe a pronounced **hard–easy disparity** in the unlearning data. As shown in Figure 1, some prompts rapidly achieve high and stable rewards, whereas others remain low-reward throughout training and require substantially more optimization effort. These hard cases often lie near the semantic boundary between forgetting target knowledge and preserving useful retain behavior. In late-stage on-policy GRPO, repeatedly sampling already-converged easy cases provides little informative gradient signal. Meanwhile, hard cases are also under-utilized: their low-reward rollouts often lead to ineffective updates and are discarded after a single use, despite containing valuable information about the model's failure modes.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Motivated by this training dynamic, we propose Replay-enhanced Reinforcement UnLEarning (**ReRULE**), an off-policy replay enhancement mechanism. During early Stage 2 GRPO training, **ReRULE** stores rollout groups whose mean reward falls below a threshold into a replay buffer. Once the buffer contains sufficient hard cases and training has completed a dataset-specific warm-up step, **ReRULE** performs hybrid training: it keeps the standard on-policy GRPO update while adding an importance-sampled off-policy update over replayed hard cases. This redirects the attention of gradient computation from converged easy cases to boundary cases that still require further learning.

Empirically, this targeted replay improves the setting where the hard/easy disparity is highly pronounced. On MUSE-Books, **ReRULE** improves Retain Quality from 46.3 to 56.2 while preserving strong forgetting performance. The additional training cost remains modest, increasing total training time by only 5–11% across RWKU, MUSE, and TOFU. On TOFU, where the forget01 setting is smaller and hence easier, **ReRULE** brings limited improvement over RULE; this conditional behavior supports the claim that replay is most valuable when persistent hard cases are present.

In summary, our contributions are threefold:

- We identify a training inefficiency in RL-based LLM unlearning: on-policy GRPO spends late-stage computation on converged easy cases while hard cases near the forget/retain boundary remain under-optimized.
- We propose **ReRULE**, an off-policy hard-case replay method that stores low-reward rollout groups and reuses them through importance-sampled updates, thereby improving the training efficiency of reinforcement unlearning. We further theoretically show that **ReRULE** tightens the hard-case convergence bound.
- We validate **ReRULE** on three benchmarks (RWKU, MUSE-Books, and TOFU), showing that it preserves the strong forgetting performance of RULE while improving retention where the hard/easy disparity is pronounced: on MUSE-Books, **ReRULE** improves Retain Quality by 9.9 absolute points over RULE with only 5–11% additional training time. A controlled ablation shows that the gain stems from hard-case selection: replacing it with random storage degrades **ReRULE** to RULE-level performance.

2 Related Work

In this section, we discuss existing LLM unlearning methods and summarize their key characteristics.

Generation-based LLM Unlearning methods. Generation-based methods refer to training-free approaches (Pang et al., 2025) that do not modify model parameters or alter the model's internal representations. Instead, they steer the model's output distribution away from harmful regions through prompt manipulation or output-level adjustments (Gao et al., 2024; Thaker et al., 2024; Muresanu et al., 2024; Pawelczyk et al., 2023). Guard (Deng et al., 2025) trains a classifier to determine whether the outputs of a large language model contain hazardous information related to the forget data. ECO Prompt (Liu et al., 2024) also trains a classifier, but applies lightweight perturbations to steer the outputs toward safe responses. In contrast, Soft Prompt Unlearning (Bhaila et al., 2025) avoids hazardous outputs by appending learnable soft prompts to the inputs.

Fine-tuning-based LLM Unlearning methods. Fine-tuning-based methods refer to approaches that achieve unlearning by updating model parameters through training (Pang et al., 2025). The Gradient Ascent (GA) (Bourtoule et al., 2021) method inverts the loss used in supervised fine-tuning, effectively turning gradient descent on target knowledge into gradient ascent, thereby inducing forgetting. Gradient Difference (GD) (Wang et al., 2023) further extends GA by incorporating retain data into the optimization process. NPO (Zhang et al., 2024a) introduces a reference model to impose a lower-bound constraint on the optimization, thereby mitigating the collapse risk of GA. SimNPO (Fan et al., 2025) removes the reference model and provides a simplified variant of NPO. Similarly, the lower-bound regularization principle in NPO is also adopted in DPO and KTO (Ethayarajh et al., 2024). FLAT (Wang et al., 2024) achieves unlearning by maximizing the f-divergence between the available template answers and the forget targets. SGA (Pang et al., 2025) introduces

normal data that are similar to the forget data, enabling the model to continue learning from normal data while performing unlearning, and thus alleviating the instability caused by gradient ascent. RULE (Zhang et al., 2025) reformulates unlearning as a rewardable refusal behavior and leverages reinforcement learning to achieve forgetting.

Motivated by the under-optimization of hard cases in the stage-two training of RULE, we propose **ReRULE**, which improves both the performance on hard cases and the training efficiency of LLM reinforcement unlearning.

Hard-case utilization in RL training. The gap between easy and hard prompts has been widely observed in RL. In RL with verifiable rewards (RLVR), DAPO (Yu et al., 2025) and related prioritized prompt-sampling methods improve training efficiency by filtering or upweighting unsolved prompts, but still require fresh rollouts at each step. In contrast, classical off-policy RL methods, such as experience replay (Mnih et al., 2015) and prioritized experience replay (Schaul et al., 2016), reuse historical experiences based on transition-level priorities. **ReRULE** differs from these approaches by replaying stored GRPO rollout groups selected by group-level rewards, which preserves GRPO's advantage structure and identifies prompts near the forget/retain boundary. Moreover, **ReRULE** avoids repeated rollout generation, introducing only 5–11% additional training cost. To our knowledge, hard-case replay has not been explored in RL-based LLM unlearning.

3 **ReRULE**: Replay Enhancement for Reinforcement Unlearning

In this section, we first summarize the prior work RULE and explain our motivation. We then introduce the overall pipeline of **ReRULE**.

3.1 RULE: Reinforcement UnLEarning

In the LLM unlearning literature, conventional methods typically optimize a joint objective over the forget and retain sets:

$$\min_{\theta} \underbrace{\mathbb{E}_{(x_f,y_f)\in\mathcal{D}_f}[\ell_f(y_f|x_f;\theta)]}_{\text{forget}} + \lambda \underbrace{\mathbb{E}_{(x_r,y_r)\in\mathcal{D}_r}[\ell_r(y_r|x_r;\theta)]}_{\text{retain}}. \tag{1}$$

Here, θ denotes the model parameters, while $\mathcal{D}_f$ and $\mathcal{D}_r$ denote the forget and retain sets, respectively, and x and y represent the input query and its corresponding target response (Yao & Xu, 2024).

However, direct optimization of such objectives can substantially degrade model utility. To mitigate this issue, RULE (Zhang et al., 2025) reformulates unlearning as learning to refuse rather than directly forgetting. It trains the model with reinforcement learning techniques to behave differently on the two sides of a refusal boundary. The specific procedure is as follows:

Stage 1: Rejection Steering. RULE first applies supervised learning on the forget set $\mathcal{D}_f$ to warm-start the model toward refusal behavior, enabling the model to learn how to refuse. The construction details of the refusal supervision data are provided in Appendix A. The optimization objective is:

$$\theta_{\text{rej}} = \arg\max_{\theta} \mathbb{E}_{(x,y^*)\sim\mathcal{D}_f}\left[\log \pi_\theta(y^* \mid x)\right],$$

where y^* denotes the corresponding refusal response.

Stage 2: Refusal Boundary Optimization. After the refusal SFT stage, RULE further refines the refusal boundary through on-policy RL, enabling the model to distinguish between forget queries and retain queries on top of the learned refusal behavior. Specifically, the policy is optimized on the union of the forget set $\mathcal{D}_f$ and the constructed boundary retain set $\widetilde{\mathcal{D}}_r$. The construction details of $\widetilde{\mathcal{D}}_r$ are provided in Appendix A.

Specifically, RULE optimizes the policy using on-policy RL methods, such as PPO (Schulman et al., 2017) and GRPO (Shao et al., 2024), with a KL regularization term anchoring the policy to π_{rej}:

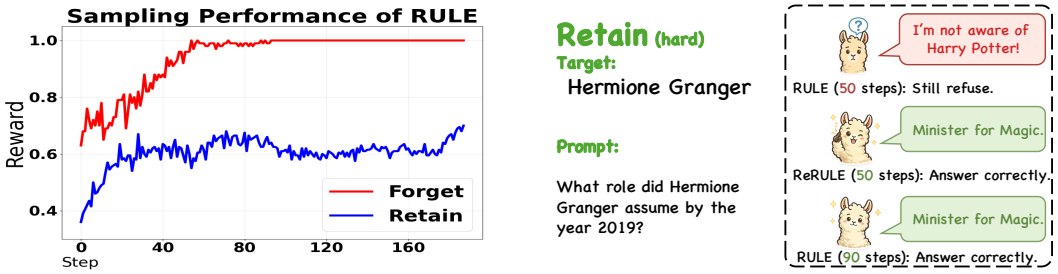
$$\max_{\theta} \mathbb{E}_{x,\, y\sim\pi_\theta}\left[r(x,y)\right] - \beta\, D_{\text{KL}}\left[\pi_\theta \,\|\, \pi_{\text{rej}}\right]. \tag{2}$$

The reward function $r(x, y)$ in RULE is defined as:

$$\begin{cases} \lambda_f \cdot \mathbb{I}[y \in \mathcal{P}_{\text{refuse}}] + (1 - \lambda_f) \cdot \mathbb{I}[k(x) \subset y], & x \in \mathcal{D}_f, \\ \lambda_r \cdot \mathbb{I}[y \notin \mathcal{P}_{\text{refuse}}] + (1 - \lambda_r) \cdot \mathbb{I}[\text{ROUGE-L}(y, y_{\text{gold}}) > \gamma], & x \in \widetilde{\mathcal{D}}_r. \end{cases}$$

3.2 Hard Cases in RULE Stage 2 Training.

In Stage 2 of RULE, we observe a critical issue: the model exhibits significantly different sampling behaviors across questions of different types and topics. As illustrated in figure 1, the model shows substantial variance in sampling performance on the MUSE-Books task. Based on these performance disparities, we can naturally divide the training samples into two parts: Hard cases and Easy cases.



(a) Reward by prompt type

(b) Different answers for the same step

Figure 1: (a) Validation-set rewards during Stage 2 training of RULE on the MUSE-Books task. Questions are categorized into Forget and Retain groups. We observe that some questions consistently exhibit relatively poor sampling performance throughout training. (b) For hard cases, RULE requires nearly twice as many training steps to achieve satisfactory performance, whereas **ReRULE** saves these additional steps.

Meanwhile, the gradient of on-policy GRPO (Shao et al., 2024) can be written as:

$$\nabla_\theta J_{\text{on}}(\theta) = \mathbb{E}_{x \sim p(x),\, y \sim \pi_\theta(\cdot|x)} \left[\underbrace{\nabla_\theta \log \pi_\theta(y|x) \cdot \hat{A}(x, y)}_{g(x,y)} \right], \tag{3}$$

where x denotes the sampled prompt, y denotes the rollout generated by the current policy π_θ, and $\hat{A}(x, y)$ represents the group-normalized GRPO advantage.

Given a predefined threshold τ, we partition the prompt space into **Easy** and **Hard** regions based on the **group mean reward** and τ.

The **Easy** region is defined as $\mathcal{E} = \{x : \mathbb{E}_{y \sim \pi_\theta(\cdot|x)}[r(x, y)] \geq \tau\}$, while the **Hard** region is defined as $\mathcal{H} = \{x : \mathbb{E}_{y \sim \pi_\theta(\cdot|x)}[r(x, y)] < \tau\}$.

The on-policy GRPO gradient can then be decomposed as:

$$\nabla_\theta J_{\text{on}}(\theta) = p(\mathcal{E})\, \mathbb{E}_{\mathcal{E}}[g(x, y)] + p(\mathcal{H})\, \mathbb{E}_{\mathcal{H}}[g(x, y)]. \tag{4}$$

For Easy cases, the model already performs well during the early stage of training. The rewards within each group are generally high and close to each other, resulting in $\hat{A} \approx 0$. Consequently, during the later stage of training, the optimization process mainly relies on the within-group relative advantages of Hard cases.

However, continuing to perform sampling on Easy cases at this stage becomes inefficient and leads to unnecessary computational overhead. Motivated by this observation, we propose **ReRULE**, an improved variant built upon RULE.

3.3 The Pipeline of ReRULE

ReRULE mainly introduces modifications to Stage 2 of the original RULE framework. The Stage 1 SFT procedure remains identical to that in the original RULE framework, while the following

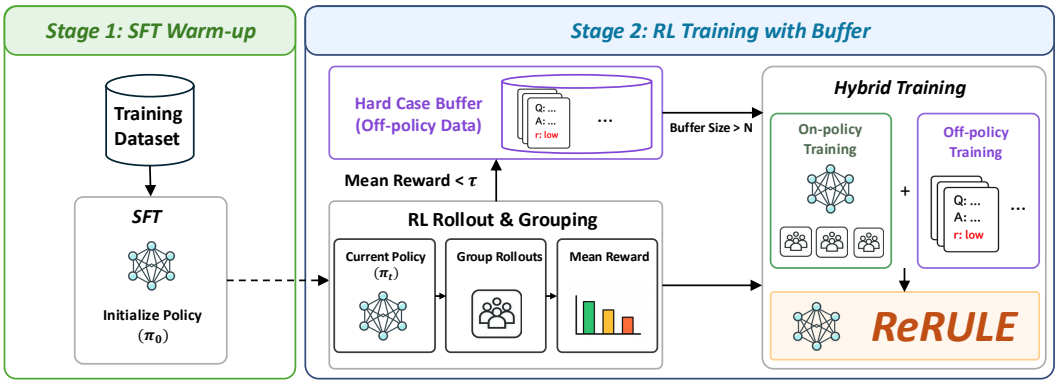


Figure 2: Workflow of **ReRULE**. Stage 1 follows RULE and performs supervised fine-tuning (SFT) on a refusal dataset to initialize the model. In Stage 2, a hard-case buffer is introduced: during the early stage of GRPO training, groups with a **mean reward** below the threshold τ are stored in the buffer. **Once the buffer reaches its capacity, newly added data replace the oldest entries in a first-in, first-out manner.** When a predefined condition is met, hybrid training is activated by combining on-policy rollouts with off-policy data sampled from the buffer.

modifications are introduced in Stage 2. During the early stage of training, we perform standard on-policy GRPO optimization while simultaneously constructing a replay buffer that stores groups whose **mean rewards** are lower than a predefined threshold τ. Intuitively, these groups correspond to cases where the model exhibits unsatisfactory sampling behavior during the early training stage.

Algorithm 1 Stage 2 of **ReRULE**

Require: Policy π_θ, replay buffer $\mathcal{B}$, threshold τ, minimum buffer size $B_{\min}$, warm-up step T_{warm}
for each training step t **do**
  Sample prompts from $\mathcal{D}_f \cup \widetilde{\mathcal{D}}_r$
  Generate rollout groups g with policy π_θ
  Compute rewards $r(x, y)$ and GRPO advantages $\hat{A}(x, y)$
  Perform on-policy GRPO update using $\nabla_\theta J_{\text{on}}(\theta)$
  for each rollout group g **do**
    if $\bar{r}(g) < \tau$ **then**
      Store g into replay buffer $\mathcal{B}$
    end if
  end for
  if $|\mathcal{B}| \geq B_{\min}$ and $t \geq T_{\text{warm}}$ **then**
    Sample replay groups from $\mathcal{B}$
    Compute Importance Sampling (IS) weight $\tilde{\rho}(y|x)$
    Perform off-policy replay update using $\nabla_\theta J_{\text{off}}(\theta)$
  end if
end for

During the later stage of training, in addition to the standard on-policy GRPO update, we introduce an additional off-policy replay update at each training step based on samples drawn from the replay buffer. Specifically, the off-policy gradient is defined as:

$$\nabla_\theta J_{\text{off}}(\theta) = \mathbb{E}_{g\sim\mathcal{B}}\left[\mathbb{E}_{y\sim\pi_{\theta_{\text{old}}}}\left[\tilde{\rho}(y|x)\nabla_\theta \log \pi_\theta(y|x)\hat{A}(x, y)\right]\right] \tag{5}$$

where $\mathcal{B}$ denotes the replay buffer, and

$$\tilde{\rho}(y|x) = \text{Normalize}\left(\text{Clip}\left(\frac{\pi_\theta(y|x)}{\pi_{\theta_{\text{old}}}(y|x)}\right)\right)$$

represents the importance-sampling weight used to reweight replayed trajectories under the current policy. In our GRPO implementation, the reward and advantage are computed at the sequence level, while the importance-sampling ratio and policy loss are computed at the token level. The clipping range in $\text{Clip}(\cdot)$ is a tunable hyperparameter and can be set during the training configuration.

Table 1: *llama3-8b-instruct* results on RWKU. We also report the training token budget for D_f and D_r (Appendix C). The best results are in **green** and the second best results are in **blue**. Results where **ReRULE** outperforms RULE are underlined.

Methods	# Tokens		Forget Quality (↓)				Forget Naturalness (↑)				Retain Quality (↑)		
	D_f	D_r	FB	QA	AA	All	Read	Help	Truth	All	FB	QA	All
Original	0%	0%	85.6	70.3	74.7	76.9	94.0	26.4	91.5	70.6	93.1	82.0	87.6
GA		0%	72.0	64.6	68.5	68.4	45.8	33.2	43.2	40.7	**85.0**	**74.7**	**79.8**
+GDR	100%	100%	72.6	64.0	69.7	68.8	30.4	23.5	27.2	27.0	**86.2**	**76.5**	**81.4**
+KLR		100%	70.7	57.5	69.9	66.1	39.7	27.6	33.1	33.5	80.5	70.5	75.5
NPO		0%	46.6	39.0	35.3	40.3	39.9	25.9	36.3	34.0	79.2	70.9	75.1
+GDR	100%	100%	52.2	43.9	42.9	46.3	89.7	56.2	67.7	71.2	82.5	70.5	76.5
+KLR		100%	52.5	40.6	43.2	45.4	92.1	56.6	69.6	72.8	83.2	72.1	77.6
SimNPO		0%	42.1	36.1	42.2	40.1	35.5	26.4	29.6	30.5	82.8	70.3	76.5
+GDR	100%	100%	51.1	39.2	50.7	47.0	39.4	23.9	29.7	31.0	83.6	75.3	79.5
+KLR		100%	44.6	35.4	44.6	41.5	50.6	25.5	34.5	36.9	82.9	71.4	77.1
RULE													
Rej. Steer	6.29%	0%	77.1	43.0	51.2	57.1	90.7	34.8	94.8	73.4	83.2	71.6	77.4
RULE$_{\text{GRPO}}$	12.1%	8.03%	**28.0**	**16.8**	**38.3**	**27.7**	**99.6**	**71.9**	**95.7**	**89.1**	76.2	71.3	73.7
ReRULE													
ReRULE$_{\text{GRPO}}$	12.1%	8.03%	**26.5**	**13.1**	**33.2**	**24.2**	**95.2**	**56.3**	**94.7**	**82.1**	79.5	68.8	74.1

Such replay-based retraining on Hard cases stored in the replay buffer is expected to improve both the efficiency and performance of Stage 2 GRPO optimization in RULE. This motivation is empirically validated in Section 4.3 and Figure 3. We further provide a theoretical analysis showing that **ReRULE** achieves a tighter convergence bound than pure on-policy RULE under the hard/easy case decomposition (Appendix E).

4 Experiment

In this section, we evaluate the performance of **ReRULE** on tasks commonly studied in LLM unlearning, including Real-World Knowledge Unlearning, Copyrighted Unlearning, and Entity Unlearning. We conduct experiments on the corresponding benchmarks, namely RWKU (Jin et al., 2024), MUSE-Books (Shi et al., 2024), and TOFU (Maini et al., 2024), respectively. All experiments are conducted using the same computational resources and experimental configurations.

4.1 Baseline Methods

We compare **ReRULE** with existing unlearning approaches to evaluate its effectiveness. Across all experiments, we include three representative methods, GA (Yao & Xu, 2024), NPO (Zhang et al., 2024a), and RULE (Zhang et al., 2025). For the RWKU and MUSE-Books benchmarks, we additionally compare **ReRULE** against Sim-NPO (Fan et al., 2025), a variant of NPO. For the TOFU benchmark, we further include KL (Maini et al., 2024), GD (Liu et al., 2022), LLMU (Yao & Xu, 2024), PO (Maini et al., 2024), DPO, FLAT (Wang et al., 2024), and SGA (Pang et al., 2025) as baselines.

4.2 Real-World Knowledge Unlearning

Experiment Setup. The RWKU benchmark (Jin et al., 2024) is a real-world knowledge unlearning benchmark consisting of 200 real-world celebrity entities as forgetting targets, with 13,131 multi-level forgetting probes in total. Unlike TOFU (Maini et al., 2024), RWKU follows a zero-shot unlearning setting, where neither the forget corpus nor the retain corpus is accessible during training. In our experiments, we follow the setting of RULE (Zhang et al., 2025) and use Meta Llama3-8B-Instruct (Grattafiori et al., 2024) as the base model. Boundary data are constructed by replacing target entities with neighboring entities, and only a small portion of the forget set and boundary set is used for reinforcement unlearning training.

Evaluation Metrics. To evaluate unlearning performance on RWKU, we follow RULE and report Forget Quality, Forget Naturalness, and Retain Quality. Forget Quality is measured by ROUGE-L on forget probes, including fill-in-the-blank (FB), question-answering (QA), and adversarial attack (AA) queries, where lower scores indicate better forgetting. Retain Quality is evaluated on neighboring retain probes, where higher scores indicate better preservation of non-target knowledge. In addition, Forget Naturalness measures the quality of refusal responses from the aspects of Readability, Helpfulness, and Truthfulness.

Table 2: **llama2-7b results on MUSE-books.** We report forgetting quality, naturalness of refusal, and utility retention. The training token ratio for D_f and D_r is listed per method (Appendix C). The best results are in **green** and the second best results are in **blue**. Results where **ReRULE** outperforms RULE are underlined.

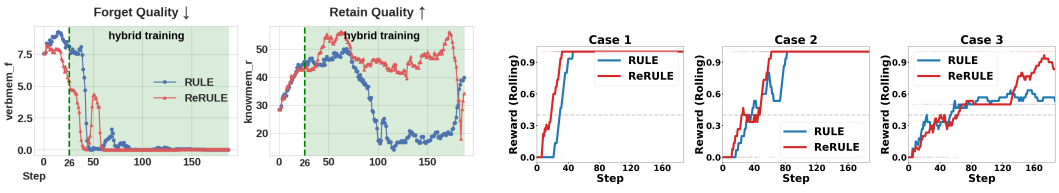
Methods	# Tokens		Forget Quality(↓)		Forget Naturalness(↑)			Retain Quality(↑)
	D_f	D_r	Verb.	Know.	Read	Help	Truth	Utility
Original	0%	0%	58.4	63.9	-	-	-	55.2
GA	100%	0%	**0.0**	**0.0**	94.0	**63.0**	77.6	0.0
+GDR	100%	100%	**0.0**	**0.0**	94.0	60.0	79.6	10.9
+KLR	100%	100%	**0.0**	**0.0**	94.0	**61.6**	80.0	40.5
NPO	100%	0%	11.9	4.7	94.4	58.6	80.0	5.9
+GDR	100%	100%	21.1	32.5	94.0	58.2	78.0	62.4
+KLR	100%	100%	8.0	45.4	94.6	60.4	81.4	**67.3**
SimNPO	100%	0%	**0.0**	**0.0**	93.8	60.2	80.6	0.0
+GDR	100%	100%	0.6	23.4	95.2	59.6	81.2	**64.8**
+KLR	100%	100%	47.4	46.2	94.6	61.2	82.4	**67.3**
RULE								
RULE$_{\text{GRPO}}$	2.9%	2.9%	**0.0**	**1.4**	**99.2**	38.2	**90.8**	46.3
ReRULE								
ReRULE $_{\text{GRPO}}$	2.9%	2.9%	**0.0**	**1.4**	**98.8**	28.7	**93.0**	56.2

Under the same experimental settings (Appendix D), ReRULE achieves better forgetting performance than RULE with comparable retention. As shown in Table 1, under the GRPO training setting, **ReRULE** achieves the best Forget Quality on all probe types (24.2 vs. 27.7 overall), while Retain Quality remains comparable to RULE (74.1 vs. 73.7). We observe a trade-off on Forget Naturalness: **ReRULE** remains Top-2 among all methods but falls below RULE (82.1 vs. 89.1), mainly on the Helpfulness dimension, suggesting that a sharper refusal boundary slightly reduces the helpfulness of refusal responses.

4.3 Copyrighted Unlearning

Experiment Setup. The MUSE benchmark (Shi et al., 2024) is an unlearning benchmark that requires models to forget copyrighted content while preserving general utility. Following RULE, we use Meta Llama2-7B as the base model. In the MUSE-books setting, the forgetting target is the Harry Potter corpus, which contains 3045 text passages. RULE constructs QA-style forget queries using GPT-4o-mini (OpenAI et al., 2024) and synthesizes boundary data by replacing sensitive targets for subsequent reinforcement unlearning training.

Evaluation Metrics. To evaluate unlearning performance on MUSE-Books, we follow RULE and report three groups of metrics: Forget Quality, Forget Naturalness, and Retain Quality. Forget Quality includes VerbMem and KnowMem, which measure whether the model still memorizes the forgotten Harry Potter content, where lower scores indicate better forgetting. Forget Naturalness is evaluated from Readability, Helpfulness, and Truthfulness, with higher scores indicating more natural and reliable refusal responses. Retain Quality is measured by Utility, where higher values indicate better preservation of general model capability.



(a) Overall metric comparison on MUSE.

(b) Tracking of hard cases during training.

Figure 3: (a) Comparison between RULE and **ReRULE** on MUSE. The green vertical line marks the start of hybrid training at step 26. (b) Smoothed reward curves for three hard cases whose early-stage average reward is below 0.4.

ReRULE achieves top-2 performance on both Forget Quality and Forget Naturalness. As shown in Table 2, both RULE and **ReRULE** achieve top-2 performance on most metrics. Moreover,

compared with several methods that obtain the best Forget Quality, **ReRULE** achieves a substantially higher Retain Quality score of 56.2, while the original RULE only achieves 46.3.

Improved Retain Quality demonstrates the effectiveness of ReRULE on hard cases. As shown in Figure 3a, both RULE and **ReRULE** quickly achieve strong Forget Quality, partly because forget-related tasks in MUSE-Books are relatively easy, as shown in Figure 1a. However, RULE shows substantial instability in Retain Quality, suggesting that it keeps exploring hard cases but rarely samples responses that effectively improve KnowMem_r. In contrast, after hybrid off-policy/on-policy training starts at step 26, **ReRULE** increases learning on hard cases and consistently achieves better Retain Quality under the same number of training steps, consistent with our analysis in Section 3.2.

On specific hard cases, ReRULE outperforms RULE. Table 3 lists three hard cases from MUSE-Books. These three hard cases still exhibit issues where the target response is reject, indicating that in Stage 1 of **ReRULE**, the SFT does not cause the model to overfit toward refusal, there still remain some to-be-forgotten queries that the model incorrectly answers instead of refusing. Their reward curves are shown in Figure 3b. Under **ReRULE** training, these cases often **achieve higher rewards earlier than under RULE**.

Table 3: Three examples of hard cases.

Case	Question	Answer	Type
1	What phrase was Harry instructed to write repeatedly with the special quill given by Umbridge?	I must not tell lies	reject
2	How long has Snape been teaching at Hogwarts?	Fourteen years	reject
3	What was the name of the dragon that Hagrid illegally kept and sought help from Charlie Weasley to get rid of?	Norbert	normal

Random replay buffer degenerates ReRULE into RULE. To verify the role of hard cases, we replace hard-case selection with random storage while keeping the same hybrid training. As shown by **ReRULE**$_{\text{random}}$ in Table 4, random replay reduces the performance to a level close to RULE, confirming that the gains come from prioritizing hard cases rather than replay alone.

Table 4: A controlled experiment with random storage in the replay buffer.

Methods	Forget Quality (↓)		Retain Quality (↑)
	Verb.	Know.	Utility
RULE	0.0	1.4	46.3
ReRULE	0.0	1.4	56.2
ReRULE$_{\text{random}}$	0.0	1.5	46.3

4.4 Entity Unlearning

Experiment Setup. The TOFU benchmark (Maini et al., 2024) contains QA pairs from 200 fictional authors. We use the 1% forget split and adopt Llama2-7B (Touvron et al., 2023) as the base model following prior work (Pang et al., 2025).

Evaluation Metrics. Following prior work (Maini et al., 2024), we report Forget Quality (FQ), Model Utility (MU), and Forget ROUGE-L (F-RL). Higher FQ and MU indicate better forgetting and retention, respectively, while lower F-RL indicates better forgetting.

Both ReRULE and RULE achieve strong performance. Compared with other baselines, both **ReRULE** and RULE achieve the best Forget Quality and Forget ROUGE-L while maintaining strong Model Utility, showing a favorable FQ-MU trade-off (Pang et al.,

Table 5: Results of Llama2-7B on TOFU. FQ, MU, and F-RL denote Forget Quality, Model Utility, and ROUGE-L on the forget set, respectively. Best results are highlighted in green and second-best results in blue.

Model	Llama2-7B		
	FQ (↑)	MU (↑)	F-RL (↓)
Original LLM	4.4883e-06	0.6239	0.9851
Retained LLM	1.0000	0.6267	0.4080
GA/SGA ($r = 0$)	**0.0068**	0.5990	0.4817
KL	0.0030	0.5994	0.4922
GD	**0.0068**	0.5998	0.4869
LLMU	0.0030	0.5999	0.4891
PO	0.0030	**0.6323**	**0.1752**
DPO-RT	**0.0068**	0.6322	0.2595
NPO-RT	0.0030	0.5994	0.5049
FLAT (Pearson)	0.0030	**0.6304**	0.4825
SGA ($r = 0.4$)	**0.0068**	0.6027	0.4792
RULE			
RULE$_{\text{GRPO}}$	**0.1650**	0.6151	**0.1322**
ReRULE			
ReRULE$_{\text{GRPO}}$	**0.1650**	0.6076	**0.1322**

2025). This is because they reformulate unlearning as learning precise refusal behaviors rather than directly erasing knowledge. However, **ReRULE** brings limited improvement over RULE on TOFU, since the TOFU forget01 setting is relatively small and easy. As shown in Table 6, TOFU contains fewer hard cases during early training, leaving limited room for replay-based learning.

Table 6: Hard ratios under different threshold settings.

Experiment	τ	Hard Ratio
MUSE	0.4	67.5
TOFU	0.4	47.5
TOFU	0.6	54.3

4.5 Ablation Study

Regarding computational cost. ReRULE introduces an additional importance-weighted replay update but avoids repeated rollout generation by storing sampled trajectories. As shown in Table 7, **ReRULE** only introduces 5–11% additional training cost.

Table 7: Training cost comparison between RULE and **ReRULE** on different benchmarks. All training experiments were conducted on 4 NVIDIA A800 GPUs.

Benchmarks	**RWKU**	**MUSE**	**TOFU**
Data Size	12000	5988	80
Training Time			
RULE	21.2 h	6.8 h	1.8 h
ReRULE	22.3 h	7.1 h	2.0 h

Regarding replay buffer settings. To examine whether the replay buffer design is a key factor affecting the experimental results, we conduct ablation studies with different values of τ, which controls the criterion for selecting data into the replay buffer. When τ is set to 0.4, groups with an average reward below 0.4 are stored in the replay buffer; when τ is set to 0.6, groups with an average reward below 0.6 are stored, and so on. The results are shown in Table 8.

Table 8: Experimental results under different replay buffer settings. **ReRULE**$_{0.6}$ denotes the variant where the threshold τ is set to 0.6, while **ReRULE**$_{1.0}$ denotes the variant where τ is set to 1.0. In the original MUSE-Books experiment of **ReRULE**, τ is set to 0.4.

Methods	Forget Quality (↓)		Retain Quality (↑)
	Verb.	Know.	Utility
RULE	0.0	1.4	46.3
ReRULE	0.0	1.4	56.2
ReRULE$_{0.6}$	6.6	1.3	52.3
ReRULE$_{1.0}$	4.3	1.4	53.7

As τ increases, the criterion for hard cases becomes less selective, causing more samples that are already relatively easy for the model to be stored in the replay buffer. During hybrid training, repeatedly updating on these easy samples can introduce redundant optimization, reducing training efficiency and ultimately leading to worse overall performance.

5 Conclusion

Building on RULE, this paper identifies a hard–easy disparity in on-policy reinforcement unlearning: late-stage GRPO repeatedly samples converged easy cases, while low-reward hard cases near the forget/retain boundary are discarded. We propose **ReRULE**, which stores hard-case rollout groups in a replay buffer and combines on-policy GRPO with importance-weighted off-policy updates. Experiments on three benchmarks show that **ReRULE** maintains RULE's forgetting performance while improving retention, achieving a 9.9-point gain in Retain Quality on MUSE-Books with only 5–11% additional training cost. Ablations verify that the improvement comes from hard-case selection rather than replay alone, as random storage reduces performance to the RULE level. On easier TOFU, where hard cases are limited, the gains are correspondingly smaller.

AI USE STATEMENT

This paper uses AI tools to polish the writing and correct grammatical errors. In addition, we use GPT-4o-mini to generate refusal data to help the model learn appropriate refusal boundaries. The prompts used for data generation are provided in Appendix A.

ETHICS STATEMENT

This work studies LLM unlearning and does not involve human subjects, private user data, or real-world deployment. All experiments are conducted on publicly available benchmarks and pretrained models. We do not identify any additional ethical concerns beyond those generally associated with LLM research.

REPRODUCIBILITY STATEMENT

To support reproducibility, we provide detailed experimental configurations, including datasets, models, hyperparameters, evaluation metrics, and implementation details, in the appendix D. We will release our code publicly upon acceptance of the paper.

REFERENCES

Karuna Bhaila, Minh-Hao Van, and Xintao Wu. Soft prompting for unlearning in large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 4046–4056, 2025.

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

Timothy Chu, Zhao Song, and Chiwun Yang. How to protect copyright data in optimization of large language models? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 17871–17879, 2024.

Badhan Chandra Das, M Hadi Amini, and Yanzhao Wu. Security and privacy challenges of large language models: A survey. *ACM Computing Surveys*, 57(6):1–39, 2025.

Zhijie Deng, Chris Yuhao Liu, Zirui Pang, Xinlei He, Lei Feng, Qi Xuan, Zhaowei Zhu, and Jiaheng Wei. Guard: Generation-time llm unlearning via adaptive restriction and detection, 2025. URL https://arxiv.org/abs/2505.13312.

Zonglin Di, Sixie Yu, Yevgeniy Vorobeychik, and Yang Liu. Adversarial machine unlearning. In *International Conference on Learning Representations*, volume 2025, pp. 21612–21633, 2025.

Ronen Eldan and Mark Russinovich. Who's harry potter? approximate unlearning for llms. 2023.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for llm unlearning, 2025. URL https://arxiv.org/abs/2410.07163.

Chongyang Gao, Lixu Wang, Kaize Ding, Chenkai Weng, Xiao Wang, and Qi Zhu. On large language model continual unlearning. *arXiv preprint arXiv:2407.10223*, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux,

Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy

Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Michael M Grynbaum and Ryan Mac. The times sues openai and microsoft over ai use of copyrighted work. *The New York Times*, 27(1), 2023.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Zhuoran Jin, Pengfei Cao, Chenhao Wang, Zhitao He, Hongbang Yuan, Jiachun Li, Yubo Chen, Kang Liu, and Jun Zhao. Rwku: Benchmarking real-world knowledge unlearning for large language models, 2024. URL https://arxiv.org/abs/2406.10890.

Antonia Karamolegkou, Jiaang Li, Li Zhou, and Anders Søgaard. Copyright violations and large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7403–7412, 2023.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*, 2023.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Bo Liu, Qiang Liu, and Peter Stone. Continual learning and private unlearning. In *Conference on Lifelong Learning Agents*, pp. 243–254. PMLR, 2022.

Chris Y Liu, Yaxuan Wang, Jeffrey Flanigan, and Yang Liu. Large language model unlearning via embedding-corrupted prompts. *Advances in Neural Information Processing Systems*, 37:118198–118266, 2024.

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

Niloofar Mireshghallah, Hyunwoo Kim, Xuhui Zhou, Yulia Tsvetkov, Maarten Sap, Reza Shokri, and Yejin Choi. Can llms keep a secret? testing privacy implications of language models via contextual integrity theory. In *International Conference on Learning Representations*, volume 2024, pp. 1892–1915, 2024.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

Andrei I Muresanu, Anvith Thudi, Michael R Zhang, and Nicolas Papernot. Fast exact unlearning for in-context learning data for llms. *arXiv preprint arXiv:2402.00751*, 2024.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

Zirui Pang, Hao Zheng, Zhijie Deng, Ling Li, Zixin Zhong, and Jiaheng Wei. Label smoothing improves gradient ascent in llm unlearning, 2025. URL https://arxiv.org/abs/2510.22376.

Martin Pawelczyk, Seth Neel, and Himabindu Lakkaraju. In-context unlearning: Language models as few shot unlearners. *arXiv preprint arXiv:2310.07579*, 2023.

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. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay, 2016. URL https://arxiv.org/abs/1511.05952.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

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. *arXiv preprint arXiv:2407.06460*, 2024.

Robin Staab, Mark Vero, Mislav Balunovic, and Martin Vechev. Beyond memorization: Violating privacy via inference with large language models. In *International Conference on Learning Representations*, volume 2024, pp. 33832–33878, 2024.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Pratiksha Thaker, Yash Maurya, Shengyuan Hu, Zhiwei Steven Wu, and Virginia Smith. Guardrail baselines for unlearning in llms. *arXiv preprint arXiv:2403.03329*, 2024.

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.

Lingzhi Wang, Tong Chen, Wei Yuan, Xingshan Zeng, Kam-Fai Wong, and Hongzhi Yin. Kga: A general machine unlearning framework based on knowledge gap alignment. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13264–13276, 2023.

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. *arXiv preprint arXiv:2410.11143*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

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

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale, 2025. URL https://arxiv.org/abs/2503.14476.

Chenlong Zhang, Zhuoran Jin, Hongbang Yuan, Jiaheng Wei, Tong Zhou, Kang Liu, Jun Zhao, and Yubo Chen. Rule: Reinforcement unlearning achieves forget-retain pareto optimality, 2025. URL https://arxiv.org/abs/2506.07171.

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

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

Yihua Zhang, Chongyu Fan, Yimeng Zhang, Yuguang Yao, Jinghan Jia, Jiancheng Liu, Gaoyuan Zhang, Gaowen Liu, Ramana Rao Kompella, Xiaoming Liu, et al. Unlearncanvas: Stylized image dataset for enhanced machine unlearning evaluation in diffusion models. *arXiv preprint arXiv:2402.11846*, 2024b.

Yimeng Zhang, Jinghan Jia, Xin Chen, Aochuan Chen, Yihua Zhang, Jiancheng Liu, Ke Ding, and Sijia Liu. To generate or not? safety-driven unlearned diffusion models are still easy to generate unsafe images... for now. In *European Conference on Computer Vision*, pp. 385–403. Springer, 2024c.

Hao Zheng, Zirui Pang, Ling li, Zhijie Deng, Yuhan Pu, Zhaowei Zhu, Xiaobo Xia, and Jiaheng Wei. Offside: Benchmarking unlearning misinformation in multimodal large language models, 2026. URL https://arxiv.org/abs/2510.22535.

The Appendix is organized as follows:

- Appendix A describes how we generate data for the three experiments.
- Appendix B introduces the baseline methods used in our experiments.
- Appendix C presents the evaluation metrics used in our experiments.
- Appendix D describes the experimental settings and implementation details.
- Appendix E analyzes the generalisation advantage of RULE and **ReRULE** over standard supervised fine-tuning.

A Data Generation

A.1 Refusal Data Construction

In the context of unlearning, we identify two essential types of queries that should be explicitly included in the refusal training set: **Type-I** queries that are likely to appear directly in the pretraining corpus (i.e., the forget set), and **Type-II** queries derived from them, such as QA-style questions that evaluate whether the model can still reason about the forgotten content. These two categories are particularly important because they capture the core knowledge that the model has either memorized or implicitly inferred from pretraining data. In contrast, other semantically related variants (e.g., paraphrases, indirect references, or alternative phrasings) can typically be generalized through RL optimization. Therefore, we treat these two explicitly supervised categories as anchor cases that establish the model's refusal boundary, while RL further improves generalization to unseen but semantically related queries. Based on this principle, we construct the refusal data differently for each benchmark dataset.

RWKU. The dataset already provides QA-style queries (Type-II) used for rejection fine-tuning. We extend these queries via GPT-4o-mini to construct completion prompts, which aim to ask models to respond to the missing blank (Type-I). The construction prompt template is shown below:

Prompt for generating completion queries in RWKU

[User]
Transform the following question into a fill-in-the-blank declarative sentence. You may paraphrase the question to improve fluency. The sentence should be declarative and contain a blank represented by "___", which does not have to appear at the end.
Original Question: {query}
[Response]

MUSE-books. The dataset targets forgetting the "Harry Potter" book, which includes 3,045 raw text passages (Type-I). We construct QA-style queries (Type-II) directly from the source content. For each passage, we prompt GPT-4o-mini to generate three QA pairs, from which we randomly sample 841 final queries for training. We use the following QA construction prompt:

Prompt for generating QA queries in MUSE-books

[User]
Please generate three question-answer pairs based on the following context, the output format should be a JSON object:

```
{
"questions": [
{
"question": "A single question related to the excerpt...",
"answer": "A precise answer extracted verbatim..."
},
...
]
}
```

Input context: {query}
[Response]

TOFU. The TOFU benchmark provides QA-style forget queries (Type-II) for fictional authors. Following the RULE setting, we further construct completion-style prompts (Type-I) from the original QA pairs. Specifically, we first identify the target author names from the forget set using rule-based name extraction. We then prompt GPT-4o-mini to transform each QA-style query into a fill-in-the-blank completion task that preserves the original semantic content while resembling pretraining-style text completion. Similar to RWKU, these completion prompts are used to improve the model's refusal behavior on pretraining-like memorization patterns. The resulting dataset therefore contains both QA-style refusal queries and completion-style refusal queries for Stage 1 rejection steering.

Prompt for generating completion queries in TOFU

[User]
Transform the following question-answer pair into a fill-in-the-blank declarative sentence. You may paraphrase the original content to improve fluency. The sentence should contain a blank represented by "___" and should resemble a natural pretraining-style completion task.
Question: {query}
Answer: {answer}
[Response]

A.2 Boundary Data Construction

Boundary Data. To construct boundary data for Stage 2 training, we adopt a controlled prompt rewriting strategy. Specifically, we prompt GPT-4o-mini to generate semantically similar neighbor queries by replacing the sensitive entity x in the forget prompt with a permissible counterpart x' (e.g., replacing "Stephen King" with "J.K. Rowling"). The objective is to preserve the original writing style, semantic structure, and knowledge type while modifying the target entity. As a result, the generated boundary data remain highly similar to the forget queries in both semantics and format, but no longer belong to the removal target. We use the following prompt template for boundary data generation:

Prompt for generating neighbor queries

[User]
Rewrite the following question by replacing it with another well-known and real figure. Keep the writing style, sentence structure, and length as close as possible. Ensure that any referenced events or facts are real and accurate. Return the result in the following JSON format:

```
{
"question": "REWRITTEN_QUESTION_HERE",
"answer": "ACCURATE_ANSWER_HERE"
}
```

Original question:
{question}
[Response]

B BASELINE METHODS

Gradient Ascent (GA) (Yao & Xu, 2024). Gradient Ascent (GA) is a widely used baseline in machine unlearning. Instead of minimizing the training objective, GA directly maximizes the prediction loss on the forget dataset $\mathcal{D}_f$. By encouraging the model to perform poorly on forget samples, the method pushes model parameters away from knowledge associated with $\mathcal{D}_f$, thereby approximating the behavior of a model trained solely on the retain set $\mathcal{D}_r$. The optimization objective is formulated as:

$$\mathcal{L}_{\text{GA}} = -\frac{1}{|\mathcal{D}_f|} \sum_{(x_i,y_i)\in\mathcal{D}_f} \ell(x_i, y_i; \theta). \tag{6}$$

Gradient Difference (GD) (Liu et al., 2024). Gradient Difference (GD) is another commonly used baseline for machine unlearning. Different from GA, which only maximizes the loss on the forget set, GD jointly optimizes two opposing objectives: minimizing the prediction loss on the retain dataset $\mathcal{D}_r$ while simultaneously maximizing the loss on the forget dataset $\mathcal{D}_f$. This formulation aims to preserve useful knowledge from $\mathcal{D}_r$ while removing information associated with $\mathcal{D}_f$. The overall objective is defined as:

$$\begin{aligned}\mathcal{L}_{\text{GD}} = &\frac{1}{|\mathcal{D}_r|} \sum_{(x_r,y_r)\in\mathcal{D}_r} \ell(x_r, y_r; \theta) \\ &-\frac{1}{|\mathcal{D}_f|} \sum_{(x_f,y_f)\in\mathcal{D}_f} \ell(x_f, y_f; \theta).\end{aligned} \tag{7}$$

KL Minimization (KL) (Maini et al., 2024). KL Minimization (KL) combines gradient ascent on the forget set $\mathcal{D}_f$ with a KL-divergence regularization term on the retain set $\mathcal{D}_r$. The method encourages forgetting by maximizing the prediction loss on forget samples, while simultaneously constraining the model outputs on retain samples to remain close to those of the original pretrained model $M_{\hat{\theta}}$. This regularization helps preserve general utility during unlearning. The objective is formulated as:

$$\begin{aligned}\mathcal{L}_{\text{KL}} = &-\frac{1}{|\mathcal{D}_f|} \sum_{(x_f,y_f)\in\mathcal{D}_f} \ell(x_f, y_f; \theta) \\ &+\frac{1}{|\mathcal{D}_r|} \sum_{x_r\in\mathcal{D}_r} \mathrm{KL}\big(h_{\hat{\theta}}(x_r) \,\|\, h_\theta(x_r)\big)\,,\end{aligned} \tag{8}$$

where $h_{\hat{\theta}}$ and h_θ denote the output distributions produced by the original model and the unlearned model, respectively.

Preference Optimization (PO) (Maini et al., 2024). Preference Optimization (PO) modifies the model's generation preference by encouraging refusal-style responses for prompts in the forget set $\mathcal{D}_f$. To achieve this, an auxiliary dataset $\mathcal{D}_{\text{IDK}}$ is constructed, where each forget prompt is paired with a safe refusal response such as "I don't know". The training objective combines standard supervised fine-tuning on the retain set $\mathcal{D}_r$ with refusal-oriented optimization on $\mathcal{D}_{\text{IDK}}$. The objective is defined as:

$$\begin{aligned}\mathcal{L}_{\text{PO}} \;=\; & \frac{1}{|\mathcal{D}_r|} \sum_{(x_r, y_r) \in \mathcal{D}_r} \ell(x_r, y_r; \theta) \\ & + \frac{1}{|\mathcal{D}_{\text{IDK}}|} \sum_{(x_f, y_{\text{IDK}}) \in \mathcal{D}_{\text{IDK}}} \ell(x_f, y_{\text{IDK}}; \theta).\end{aligned} \tag{9}$$

This objective encourages the model to preserve its performance on the retain set while learning to refuse responses associated with the forget set.

Direct Preference Optimization (DPO) (Rafailov et al., 2023). Direct Preference Optimization (DPO) extends preference optimization to the machine unlearning setting. Rather than optimizing preferences between human-chosen and rejected responses, DPO contrasts a safe refusal response y_e against the original forget response y_f under the same forget prompt $x_f \in \mathcal{D}_f$. The objective encourages the model to assign higher preference to the refusal response, thereby suppressing knowledge associated with the forget set while maintaining overall model capability. Given inverse temperature parameter β, the objective is defined as:

$$\begin{aligned}\mathcal{L}_{\text{DPO}} \;=\; -2\beta\, \mathbb{E}_{x_f \in \mathcal{D}_f} \Bigg[& \log \sigma\Big(\beta \log h_\theta(x_f, y_e) \\ & - \beta \log h_\theta(x_f, y_f) - M_{\text{ref}}\Big)\Bigg],\end{aligned} \tag{10}$$

where h_θ denotes the predictive distribution of the current model, and M_{ref} is an optional regularization term that penalizes deviation from the original pretrained model. A retention-aware variant additionally incorporates supervised learning on the retain set $\mathcal{D}_r$ to better preserve general knowledge:

$$\mathcal{L}_{\text{DPO-RT}} \;=\; \mathcal{L}_{\text{DPO}} \;+\; \mathcal{L}(\mathcal{D}_r; \theta). \tag{11}$$

Negative Preference Optimization (NPO) (Zhang et al., 2024a). Negative Preference Optimization (NPO) is designed to suppress undesired generations by directly penalizing responses associated with the forget set $\mathcal{D}_f$. Unlike DPO, which relies on pairwise preference comparisons between preferred and dispreferred responses, NPO optimizes only the negative preference signal, explicitly discouraging the model from generating the original forget responses. Given inverse temperature parameter β, the objective is formulated as:

$$\mathcal{L}_{\text{NPO}} \;=\; -2\beta\, \mathbb{E}_{(x_f, y_f) \in \mathcal{D}_f} \Big[\log \sigma\big(-\beta \log h_\theta(y_f | x_f)\big)\Big], \tag{12}$$

where h_θ denotes the predictive distribution of the current model. To better preserve model utility, a retention-aware variant further incorporates supervised optimization on the retain set $\mathcal{D}_r$:

$$\mathcal{L}_{\text{NPO-RT}} \;=\; \mathcal{L}_{\text{NPO}} \;+\; \mathcal{L}(\mathcal{D}_r; \theta). \tag{13}$$

Mismatch. Mismatch extends the preference-based unlearning framework by introducing randomly constructed response sequences (Yao & Xu, 2024). Similar to PO, the method preserves model utility through supervised fine-tuning on the retain set $\mathcal{D}_r$, while additionally optimizing a mismatch objective based on randomly sampled responses Y_{rnd}. These random continuations are paired with forget prompts, encouraging the model to generate neutral or irrelevant outputs instead of memorized forget responses. The objective is formulated as:

$$\begin{aligned}\mathcal{L}_{\text{Mismatch}} \;=\; & \frac{1}{|\mathcal{D}_r|} \sum_{(x_r, y_r) \in \mathcal{D}_r} \ell(x_r, y_r; \theta) \\ & + \frac{1}{|\mathcal{D}_{\text{rnd}}|} \sum_{y_{\text{rnd}} \in Y_{\text{rnd}}} \ell(x_f, y_{\text{rnd}}; \theta),\end{aligned} \tag{14}$$

where Y_{rnd} denotes a set of randomly sampled responses associated with forget prompts $x_f \in \mathcal{D}_f$.

LLMU (Yao & Xu, 2024). LLMU extends Gradient Ascent by incorporating two auxiliary components: (1) random-completion unlearning using sequences generated from forget prompts, and (2) retention regularization on normal retain data. The objective encourages forgetting by maximizing loss on the forget set $\mathcal{D}_f$, while simultaneously training on random completions $\mathcal{D}_{\text{rand}}$ and aligning the model's predictions on the retain set $\mathcal{D}_{\text{normal}}$ with the original model through KL divergence. Formally, the loss is:

$$\begin{aligned}\mathcal{L}_{\text{LLMU}} \;=\; & -\frac{\varepsilon_1}{|\mathcal{D}_f|} \sum_{(x_f, y_f) \in \mathcal{D}_f} \ell(x_f, y_f; \theta) \\ & + \frac{\varepsilon_2}{|\mathcal{D}_{\text{rand}}|} \sum_{x \in \mathcal{D}_{\text{rand}}} \ell(x; \theta) \\ & + \frac{\varepsilon_3}{|\mathcal{D}_{\text{normal}}|} \sum_{x \in \mathcal{D}_{\text{normal}}} \mathrm{KL}\big(h_{\hat{\theta}}(x) \,\|\, h_\theta(x)\big)\,,\end{aligned} \tag{15}$$

Task Vectors (Eldan & Russinovich, 2023). Task Vectors construct an unlearned model by explicitly reversing the parameter update direction associated with the forget set $\mathcal{D}_f$. Let θ_o denote the parameters of the original pretrained model, and let $\theta_{\text{reinforce}}$ represent the parameters of a model additionally fine-tuned to strongly memorize the forget data. The difference between these two parameter sets defines the task vector $(\theta_{\text{reinforce}} - \theta_o)$. The unlearning process is then performed by subtracting this vector from the original model parameters:

$$\theta \;=\; \theta_o - (\theta_{\text{reinforce}} - \theta_o). \tag{16}$$

This approach removes knowledge associated with $\mathcal{D}_f$ by moving the model parameters in the opposite direction of the forget-data adaptation, without requiring additional optimization steps.

Who's Harry Potter (WHP) (Eldan & Russinovich, 2023). Who's Harry Potter (WHP) performs unlearning through distribution-level interpolation between the original model θ_o and a reinforced model $\theta_{\text{reinforce}}$ that is intentionally overfitted on the forget set. Let $p_\theta(\cdot|x)$ denote the token-level output distribution of the unlearned model for input x. WHP modifies this distribution by subtracting a scaled adaptation direction derived from the reinforced model:

$$p_\theta(\cdot|x) \;=\; p_{\theta_o}(\cdot|x) - \alpha\big(p_{\theta_{\text{reinforce}}}(\cdot|x) - p_{\theta_o}(\cdot|x)\big), \tag{17}$$

where α controls the strength of the forgetting operation. This formulation steers the model distribution away from the forget-oriented reinforced model while maintaining proximity to the original pretrained model distribution.

FLAT (Wang et al., 2024). Forget data only Loss AdjustmenT (FLAT) is a loss-adjustment-based unlearning method that does not require retain data or an additional reference model. Instead of directly applying gradient ascent on the forget set $\mathcal{D}_f$, FLAT optimizes an f-divergence objective between a safe template response y_e (such as a refusal or irrelevant completion) and the original forget response y_f. For each forget sample (x_f, y_f), the method introduces a paired safe response and encourages the model to move toward the safe template distribution while suppressing the original forget response. The objective is defined as:

$$\begin{aligned}\mathcal{L}_{\text{FLAT}} \;=\; & -\, g^*(P(x_f, y_e; \theta)) \\ & +\, f^*(g^*(P(x_f, y_f; \theta)))\,,\end{aligned} \tag{18}$$

where $P(x, y; \theta)$ denotes the average token-level prediction probability of response y conditioned on input x, while $g^*(\cdot)$ and $f^*(\cdot)$ represent the variational and conjugate functions associated with the selected f-divergence. This formulation enables effective forgetting by encouraging alignment with safe responses instead of directly optimizing against forget samples.

SGA (Pang et al., 2025). Smooth Gradient Ascent (SGA) adopts a Generalized Label Smoothing paradigm that jointly trains on forget data and model-generated normal data. By incorporating normal data during optimization, SGA alleviates the gradient collapse issue commonly observed in traditional Gradient Ascent methods. Specifically, the model simultaneously learns to forget target knowledge while maintaining stable behavior on normal generations. The optimization objective is formulated as:

$$\begin{aligned}\min_{\theta} \; & \underbrace{\left(1 - r + \tfrac{r}{K}\right)\, \mathbb{E}_{(x_f, y_f) \in \mathcal{D}_f} \left[\ell_f(y_f \mid x_f; \theta)\right]}_{\text{forget}} \\ & + \underbrace{\left(\tfrac{r}{K}\right)\, \mathbb{E}_{(x_p, y_p) \in \mathcal{D}_p} \left[\sum_{k=1}^{K} \ell_p^{(k)}(y_p^{(k)} \mid x_p^{(k)}; \theta)\right]}_{\text{normal data}}.\end{aligned} \tag{19}$$

SimNPO (Fan et al., 2025). SimNPO is a simplified variant of NPO that removes the dependency on a reference model and introduces length normalization for more stable optimization. The objective directly suppresses the likelihood of forget responses while normalizing over response length. The objective is defined as:

$$\begin{aligned}\mathcal{L}_{\text{SimNPO}} \;=\; & -\, 2\beta\, \mathbb{E}_{(x_f, y_f) \in \mathcal{D}_f} \\ & \left[\log \sigma\Big(-\frac{\beta}{|y_f|} \log h_\theta(y_f | x_f) - \gamma\Big)\right].\end{aligned} \tag{20}$$

C Evaluation Metrics

C.1 RWKU

Forget Quality. Forget Quality evaluates whether the model has successfully removed target knowledge after unlearning. In RWKU, this metric is evaluated on three categories of forget tasks: Fact-Based (FB), Question-Answering (QA), and Author Attribution (AA). FB measures whether the model can still recall factual information related to the forget set, QA evaluates the model's ability to answer forget-related questions, and AA tests whether the model can correctly identify authorship or source-related information associated with forgotten content. Lower scores indicate better forgetting performance.

Retain Quality. Retain Quality measures whether the model preserves useful knowledge and general capabilities after unlearning. In RWKU, this metric is evaluated on retain-set tasks including Fact-Based (FB) and Question-Answering (QA) evaluations. Higher scores indicate better preservation of general utility and retained knowledge.

Forget Naturalness. In RWKU, Forget Naturalness evaluates whether the unlearned model still maintains normal language quality and general utility after forgetting. The metric mainly measures the fluency, truthfulness, and downstream capability of the model on non-forget tasks, reflecting whether the model still behaves like a standard aligned LLM after unlearning.

Tokens. This refers to the proportion of data used in the RWKU task compared with the full forget and retain data. For Stage 2 of RULE and **ReRULE**, since the training is conducted on the constructed boundary set, the token ratio is smaller than 100%.

C.2 MUSE

No Verbatim Memorization (VerbMem). To assess whether the model has completely unlearned the target content, we evaluate verbatim memorization (*VerbMem*). This metric measures the similarity between the model's continuation and the ground-truth continuation from the forget set, restricted to the first l tokens of each sample. Following prior work, we use the ROUGE-L F1 (Lin, 2004) score as the evaluation metric:

$$\mathrm{VerbMem}(f, \mathcal{D}_f) = \frac{1}{|\mathcal{D}_f|} \sum_{x \in \mathcal{D}_f} \mathrm{ROUGE}\big(f(x_{[:l]}),\, x_{[l+1:]}\big). \tag{21}$$

No Knowledge Memorization (KnowMem). Knowledge memorization (*KnowMem*) measures whether the model retains factual information about forgotten records. For each question–answer pair (q, a) in the forget set $\mathcal{D}_f$, we compute the ROUGE score between the model's predicted answer $f(q)$ and the ground-truth answer a, and then average across all samples:

$$\mathrm{KnowMem}(f, \mathcal{D}_f) = \frac{1}{|\mathcal{D}_f|} \sum_{(q,a) \in \mathcal{D}_f} \mathrm{ROUGE}\big(f(q),\, a\big). \tag{22}$$

Utility Preservation. Finally, we evaluate whether the model preserves its general capabilities after unlearning. This is measured on the retain set $\mathcal{D}_r$ by computing the knowledge memorization score:

$$\mathrm{UtilityPreservation} = \mathrm{KnowMem}(f_{\text{unlearn}}, \mathcal{D}_r). \tag{23}$$

Forget Naturalness. In MUSE, Forget Naturalness mainly evaluates the quality of refusal responses generated for forget queries. The metric focuses on whether the model can produce natural, fluent, and assistant-like refusals while avoiding memorized target knowledge. Specifically, it considers factors such as readability, helpfulness, and truthfulness of the generated responses.

Tokens. Same as Tokens in RWKU's metrics.

C.3 TOFU

Probability. For each instance in either the retain or forget set, we compute the normalized conditional probability

$$P(a \mid q)^{1/|a|},$$

where q denotes the input question, a is a candidate answer, and $|a|$ is the token length of a. In the *real authors* and *world facts* subsets, the dataset provides five candidate answers $\{a_0, \tilde{a}_1, \tilde{a}_2, \tilde{a}_3, \tilde{a}_4\}$, where a_0 is the correct answer and each $\tilde{a}_i$ is a perturbed (incorrect) alternative. The probability ratio is defined as:

$$\text{Probability} = \frac{P(a_0 \mid q)^{1/|a_0|}}{\sum_{i=1}^{4} P(\tilde{a}_i \mid q)^{1/|\tilde{a}_i|}}. \tag{25}$$

Truth Ratio. The truth ratio quantifies the model's preference for perturbed answers. It is computed as the geometric mean of the normalized probabilities of all perturbed answers $\{\tilde{a}_1, \tilde{a}_2, \ldots\}$ relative to the normalized probability of the paraphrased answer $\hat{a}$:

$$R_{\text{truth}} = \frac{\left(\prod_{i=1}^{|A|} P(\tilde{a}_i \mid q)^{1/|\tilde{a}_i|}\right)^{1/|A|}}{P(\hat{a} \mid q)^{1/|\hat{a}|}}. \tag{26}$$

In the *real authors* and *world facts* subsets, where paraphrased answers are not available, the original answer a is used in the denominator.

ROUGE-L. For all TOFU subsets, we report the ROUGE-L recall score (Lin, 2004) between ground-truth answers in the forget set and the model outputs after unlearning.

Model Utility. Model utility is defined as the harmonic mean of nine scores, covering answer probability, truth ratio, and ROUGE-L recall across the retain, real authors, and world facts subsets. A higher utility score reflects stronger overall performance.

Forget Quality. Forget quality is evaluated using a Kolmogorov–Smirnov (KS) test that compares the distributions of truth ratios between the retained and unlearned models on the forget set. A higher p-value supports the null hypothesis that the two distributions are statistically indistinguishable, indicating consistent behavior between the retained and unlearned models.

D Experimental Settings

All experiments are conducted on four NVIDIA A800 GPUs. For each experiment:

RWKU Setup. All RWKU experiments use a rollout batch size of 32 and a global actor batch size of 32. The micro-batch size per device is set to 8 for parameter updates and 16 for experience generation. We use a learning rate of 2e-6, a KL coefficient of 1e-2, and train the model for a maximum of 37 steps. During GRPO sampling, 8 rollouts are generated for each prompt.

For hybrid training, the replay buffer is activated after at least 100 low-reward groups are collected and the training step exceeds 15. Groups with mean reward lower than 0.6 are stored in the replay buffer. The maximum replay buffer size is set to 1200 groups, and each off-policy update samples 32 groups from the replay buffer.

MUSE Setup. All MUSE experiments use a rollout batch size of 32 and a global actor batch size of 32. The micro-batch size per device is set to 8 for parameter updates and 16 for experience generation. We use a learning rate of 2e-6 and a KL coefficient of 1e-2. During GRPO sampling, 8 rollouts are generated for each prompt.

For hybrid training, the replay buffer is activated after at least 500 low-reward groups are collected and the training step exceeds 26. Groups with mean reward lower than 0.4 are stored in the replay buffer. The maximum replay buffer size is set to 1200 groups, and each off-policy update samples 32 groups from the replay buffer.

TOFU Setup. All TOFU experiments use a rollout batch size of 32 and a global actor batch size of 32. The micro-batch size per device is set to 4 for parameter updates and 8 for experience generation. We use a learning rate of 2e-6 and a KL coefficient of 1e-2. During GRPO sampling, 8 rollouts are generated for each prompt. The maximum response length is set to 256, and the model is trained for 11 steps.

For hybrid training, the replay buffer is activated after at least 80 low-reward groups are collected and the training step exceeds 6. Groups with mean reward lower than 0.6 are stored in the replay buffer. The maximum replay buffer size is set to 400 groups, and each off-policy update samples 32 groups from the replay buffer.

E THEORETICAL ANALYSIS: GENERALISATION ADVANTAGE OF RULE AND RERULE

Theorem 1 *(Generalisation Advantage of RULE and **ReRULE**).* Let Π be a policy class with token-wise Rademacher complexity $C(\Pi)$ on sequences of length H. Define the mis-refusal risk as:

$$\mathcal{R}(\pi) = \underbrace{\Pr_{x\sim P_f^*}[\pi(x) \neq [\text{refuse}]]}_{\text{(i) miss-refusal on forget}} + \underbrace{\Pr_{x\sim P_r}[\pi(x) = [\text{refuse}]]}_{\text{(ii) false-refusal on retain}}. \tag{24}$$

(a) SFT. Empirical risk minimisation over a forget set D_f of size n_f, using a bounded loss $\ell \in [0,1]$, yields:

$$\mathbb{E}\left[\mathcal{R}(\hat{\pi}_{\text{sft}})\right] \leq 2\sqrt{\frac{C(\Pi)}{n_f}} + \Delta_f + \underbrace{1}_{\Delta_r}, \tag{25}$$

where

$$\Delta_f = \Pr_{x\sim P_f^* \setminus D_f}[\cdot]$$

is the coverage gap on the forget set, and the final term represents worst-case retain-side risk due to no supervision on retain-side queries.

(b) RULE. After K on-policy updates collecting m boundary prompts and H-length rollouts per prompt, the returned policy $\hat{\pi}_{\text{rule}}$ satisfies, with probability at least $1-\delta$:

$$\mathcal{R}\left(\hat{\pi}_{\text{rule}}\right) \leq 2\sqrt{\frac{C(\Pi)}{n_f + KmH}} + \Delta_f + \epsilon_{\text{EXPLORE}}(K, m, H, \delta). \tag{26}$$

where the exploration error is bounded as

$$\epsilon_{\text{EXPLORE}} = O\left(\sqrt{\frac{\log(1/\delta)}{KmH}}\right).$$

Hence, for an equal token budget

$$n_f \approx KmH,$$

and under mild exploration, i.e.,

$$\epsilon_{\text{EXPLORE}} < 1,$$

we obtain:

$$\boxed{\mathbb{E}\left[\mathcal{R}\left(\hat{\pi}_{\text{rule}}\right)\right] < \mathbb{E}\left[\mathcal{R}\left(\hat{\pi}_{\text{sft}}\right)\right]} \tag{27}$$

That is, RULE improves the worst-case refusal performance compared to SFT.

(c) ReRULE. **ReRULE** further augments RULE with a hard-case replay buffer. Given a reward threshold τ, we divide prompts into Easy and Hard regions:

$$\mathcal{E} = \left\{x : \mathbb{E}_{y\sim\pi_\theta(\cdot|x)}\left[r(x,y)\right] \geq \tau\right\},$$

and

$$\mathcal{H} = \left\{x : \mathbb{E}_{y\sim\pi_\theta(\cdot|x)}\left[r(x,y)\right] < \tau\right\}.$$

Since hard cases may appear on both the forget side and the retain/boundary side, the risk can be further decomposed as:

$$\begin{aligned}\mathcal{R}(\pi) =& p_f(\mathcal{E})\mathcal{R}_{f,\mathcal{E}}(\pi) + p_f(\mathcal{H})\mathcal{R}_{f,\mathcal{H}}(\pi) \\ &+ p_r(\mathcal{E})\mathcal{R}_{r,\mathcal{E}}(\pi) + p_r(\mathcal{H})\mathcal{R}_{r,\mathcal{H}}(\pi).\end{aligned} \tag{28}$$

Here, $\mathcal{R}_{f,\mathcal{H}}$ denotes the miss-refusal risk on hard forget prompts, while $\mathcal{R}_{r,\mathcal{H}}$ denotes the false-refusal risk on hard retain or boundary prompts.

For pure on-policy RULE, the expected number of hard-region rollout tokens is:

$$N_{\mathcal{H}}^{\text{on}} = KmH \cdot p(\mathcal{H}).$$

Thus, the hard-region exploration error of pure RULE can be written as:

$$\epsilon_{\mathcal{H}}^{\text{RULE}} = O\left(\sqrt{\frac{\log(1/\delta)}{N_{\mathcal{H}}^{\text{on}}}}\right). \tag{29}$$

In contrast, **ReRULE** stores rollout groups whose mean reward is lower than τ into a replay buffer $\mathcal{B}$ and reuses them during hybrid training. Suppose the hybrid stage performs L off-policy replay updates, each using b hard groups with H-length rollouts. The number of replayed hard-region rollout tokens is:

$$N_{\mathcal{H}}^{\text{off}} = LbH.$$

Since the replayed trajectories are generated by previous policies, **ReRULE** uses a clipped and normalized importance ratio:

$$\tilde{\rho}(y|x) = \text{Normalize}\left(\text{Clip}\left(\frac{\pi_\theta(y|x)}{\pi_{\theta_{\text{old}}}(y|x)}\right)\right).$$

Let $\eta_\rho \in (0, 1]$ denote the effective-sample-size coefficient induced by importance weighting:

$$\eta_\rho = \frac{\left(\mathbb{E}\left[\tilde{\rho}\right]\right)^2}{\mathbb{E}\left[\tilde{\rho}^2\right]}.$$

Then the effective number of hard-region training tokens in **ReRULE** becomes:

$$N_{\mathcal{H}}^{\textbf{ReRULE}} = N_{\mathcal{H}}^{\text{on}} + \eta_\rho N_{\mathcal{H}}^{\text{off}}.$$

Therefore, the hard-region exploration error of **ReRULE** is bounded as:

$$\epsilon_{\mathcal{H}}^{\textbf{ReRULE}} = O\left(\sqrt{\frac{\log(1/\delta)}{N_{\mathcal{H}}^{\text{on}} + \eta_\rho N_{\mathcal{H}}^{\text{off}}}}\right). \tag{30}$$

$$N_{\mathcal{H}}^{\textbf{ReRULE}} = N_{\mathcal{H}}^{\text{on}} + \eta_\rho N_{\mathcal{H}}^{\text{off}}.$$

Since

$$N_{\mathcal{H}}^{\textbf{ReRULE}} \geq N_{\mathcal{H}}^{\text{on}},$$

we have:

$$\epsilon_{\mathcal{H}}^{\textbf{ReRULE}} \leq \epsilon_{\mathcal{H}}^{\text{RULE}}.$$

Combining the standard uniform convergence term with the hard-region replay benefit, the policy returned by **ReRULE** satisfies, with probability at least $1 - \delta$:

$$\begin{aligned}\mathcal{R}\left(\hat{\pi}_{\textbf{ReRULE}}\right) \leq 2\sqrt{\frac{C(\Pi)}{n_f + KmH + \eta_\rho N_{\mathcal{H}}^{\text{off}}}} \\ + \Delta_f + \epsilon_{\mathcal{H}}^{\textbf{ReRULE}} + \epsilon_{\text{IS}}.\end{aligned} \tag{31}$$

where ϵ_{IS} denotes the bias introduced by clipped importance sampling.

Hence, when the clipping bias is controlled and the reduction of hard-region exploration error dominates this bias, we obtain:

$$\boxed{\mathcal{R}\left(\hat{\pi}_{\textbf{ReRULE}}\right) < \mathcal{R}\left(\hat{\pi}_{\text{rule}}\right)} \tag{32}$$

That is, **ReRULE** further improves RULE by reallocating effective training tokens toward hard cases, reducing inefficient repeated sampling on Easy cases and yielding a tighter hard-region generalisation bound.