TOKEN-SELECTIVE ON-POLICY SELF-DISTILLATION FOR LLM UNLEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language model (LLM) unlearning aims to suppress designated knowledge while preserving non-target capabilities, without retraining from scratch. However, existing parameter-updating methods often rely on externally constructed supervision, while directly distilling in-context unlearning (ICU) may also transfer prompt-induced changes unrelated to forgetting. In this work, we propose Token-Selective On-Policy Logit Distillation (TOLD), a response-label-free unlearning method that uses ICU as supervision for parameter updating. TOLD compares a frozen teacher's predictions with and without the ICU prompt along student-generated trajectories. It then selectively transfers token-level changes for suppression and promotion, limiting unrelated changes during distillation and enabling prompt-free inference. Experiments on the RWKU benchmark across two backbones show that TOLD reduces mean forget-set recall by 39.6–42.5% relative to the base models while largely preserving neighboring knowledge and general utility. TOLD also demonstrates robust forgetting across nine adversarial attacks. Additional experiments on MUSE-Books extend these findings beyond entity-level unlearning. These results demonstrate the effectiveness of selective on-policy distillation for prompt-free behavioral unlearning.

1 INTRODUCTION

Large language models (LLMs) acquire broad knowledge by internalizing patterns from massive pretraining corpora (Brown et al., 2020; Bubeck et al., 2023). However, LLMs may memorize and reproduce private, copyrighted, unsafe, or otherwise undesirable information from their training corpora (Carlini et al., 2021). These risks have become particularly urgent under privacy regulations such as the European Union's General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA), which establish qualifying rights to request the erasure or deletion of personal information (Ginart et al., 2019; Harding et al., 2019; EU, 2016). Machine unlearning aims to remove the influence of specified training data from a trained model. Retraining from scratch on the remaining data provides a straightforward solution, but is prohibitively expensive for modern LLMs. Consequently, practical LLM unlearning methods seek to achieve forgetting without full retraining (Jang et al., 2023; Zhang et al., 2024). Effective unlearning should prevent target knowledge from being elicited while preserving fluency and utility on non-target inputs.

Existing LLM unlearning methods can be broadly divided into inference-time and parameter-updating approaches. In-context unlearning (ICU) (Pawelczyk et al., 2023) leaves model parameters unchanged and suppresses target-related responses through inference-time instructions, enabling immediate deployment without retraining. However, its effect is tied to the context and may disappear once the instruction is removed or overridden (Shumailov et al., 2024; Zhong et al., 2026). Parameter-updating methods instead internalize forgetting through optimization, including gradient-based fine-tuning, preference or refusal optimization, and reinforcement-learning approaches (Jang et al., 2023; Zhang et al., 2024; Rafailov et al., 2023; Zhang et al., 2026; Zaradoukas et al., 2026). While more persistent, these methods often rely on carefully constructed supervision, such as forget responses, preference pairs, or task-specific rewards. Incomplete or biased supervision can lead to insufficient forgetting or unnecessary utility degradation. Moreover, obtaining reliable supervision is often costly, as some methods require additional data generated by more capable external LLMs (Zhang et al., 2026; Zaradoukas et al., 2026). These challenges motivate unlearning methods that reduce reliance on externally constructed supervision.

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.

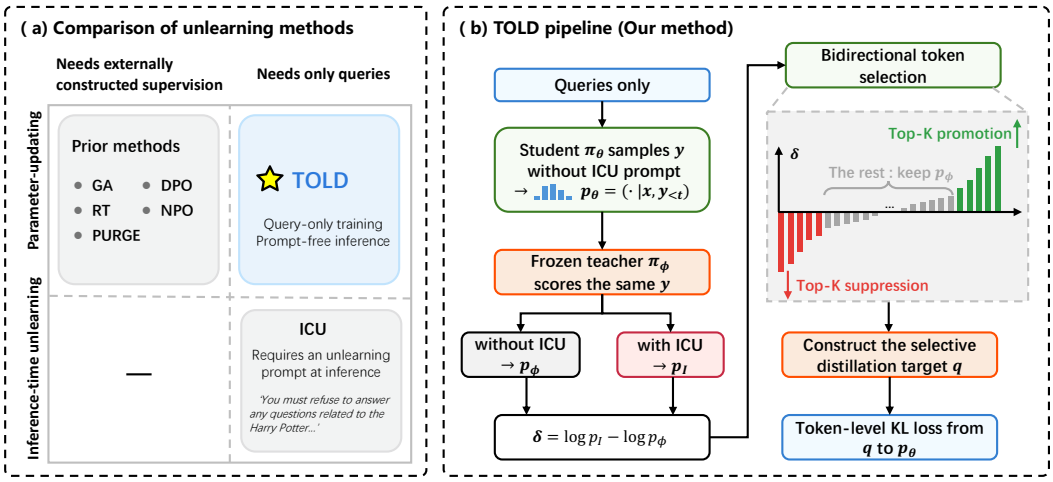

Figure 1: **Comparison and training pipeline of TOLD. (a)** Comparison of representative unlearning methods. Existing parameter-updating methods typically rely on externally constructed supervision, while ICU requires an unlearning prompt at inference time. TOLD enables query-only training and prompt-free inference. **(b)** Overview of TOLD: The student generates responses on-policy without the ICU prompt, and the frozen base model evaluates the same response with and without the ICU prompt. TOLD selects tokens to suppress or promote, constructs the selective distillation target, and trains the student with token-level KL divergence.

Recent work on self-distillation provides a promising direction for constructing training supervision without relying on external models. On-Policy Self-Distillation (OPSD) trains a model using supervision generated by the model itself (Zhao et al., 2026a). The student generates its own trajectories, while the same model, augmented with privileged context, provides a next-token probability distribution at every visited prefix. Unlike sequence-level reinforcement learning methods such as GRPO and its variants(Shao et al., 2024; Yu et al., 2026; Liu et al., 2025; Zhao et al., 2026c), which assign rewards to complete responses, OPSD provides dense token-level supervision along student-generated trajectories. This suggests a natural way to reduce reliance on externally constructed supervision for unlearning: use an ICU instruction as the privileged context for the teacher.

Although originally designed for inference-time unlearning, ICU can also provide supervision for parameter updating. By comparing the frozen model's predictions with and without the ICU instruction, we can identify token-level changes that may help suppress target knowledge. However, directly distilling the full ICU distribution is not exclusively customized for unlearning, as it may also transfer unrelated changes in style, formatting, and low-probability alternatives. This raises our central question: *how to transfer ICU-induced forgetting signals while preserving unrelated prompt-induced distribution shifts from being absorbed by the student?*

We answer this question with **TOLD** (**T**oken-Selective **O**n-Policy **L**ogit **D**istillation), an on-policy self-distillation method for LLM unlearning. TOLD uses queries generated by the base model from its own knowledge of the target and non-target entities. For each query, the student first generates a response without the ICU prompt. The frozen base model then evaluates the same response twice, once normally and once with the ICU prompt. By comparing the resulting token distributions, TOLD identifies tokens whose probabilities should be suppressed or promoted. It constructs a selective distillation target by adjusting only the selected token weights according to the ICU prediction and retaining the normal weights for the remaining tokens. The student learns this target through token-level KL divergence. Training requires only these queries without response labels or external teacher models. After training, the student performs inference without the ICU prompt. Figure 1 contrasts TOLD with representative unlearning methods and summarizes its training pipeline.

To summarize, the main contributions lie in:

- We introduce TOLD, an on-policy self-distillation framework that learns from the student's own responses. Its supervision adapts to the trajectories generated by the evolving student.
- TOLD enables response-label-free, query-only training without requiring external supervision.

- TOLD identifies tokens whose probabilities should be suppressed or promoted and selectively adjusts their weights rather than matching the full ICU distribution. This teaches the student to forget target knowledge while reducing unrelated changes to the model's behavior.
- Our experiments demonstrate that TOLD achieves a favorable forgetting–utility trade-off against representative unlearning baselines.

2 Related Work

2.1 LLM Unlearning

Existing LLM unlearning methods can be broadly divided into inference-time and parameter-updating approaches. In-context unlearning (ICU) (Pawelczyk et al., 2023) uses few-shot prompts and demonstrations to suppress responses related to targeted knowledge without updating model parameters, while aiming to preserve general model behavior. Although inexpensive and immediately applicable, its effect depends on the prompt and can be weakened when the prompt is removed or overridden (Shumailov et al., 2024; Łucki et al., 2024; Hu et al., 2025).

Parameter-updating methods instead modify model parameters to achieve persistent removal. These include gradient ascent (Jang et al., 2023), preference-based methods such as DPO (Rafailov et al., 2023) and NPO (Zhang et al., 2024), refusal-based methods (Jin et al., 2024), and reinforcement-learning approaches such as RULE (Zhang et al., 2026) and PURGE (Zaradoukas et al., 2026). Recent work also improves locality by selectively applying unlearning to important token positions (Wan et al., 2025; Yüce et al., 2026). Although these methods differ in their optimization objectives, they typically require explicit supervision for how the model should forget or what behavior should be preserved. Such supervision often requires additional data generation or careful design, increasing the cost of unlearning. In contrast, TOLD derives its supervision from the frozen base model by comparing predictions with and without the ICU prompt on student-generated trajectories.

2.2 Self-Distillation and On-Policy Distillation

Knowledge distillation trains a student to match the predictions of a teacher, while self-distillation obtains supervision from the same model or model family (Gou et al., 2021; Hinton et al., 2015). For autoregressive language models, conventional distillation on fixed sequences may suffer from a mismatch between training and inference trajectories. On-policy distillation addresses this issue by training the student on its own generated trajectories while receiving feedback from a teacher (Agarwal et al., 2024; Li et al., 2026). OPSD extends this approach by using the same model, augmented with privileged context, to provide token-level supervision along the student's trajectories (Zhao et al., 2026a;b; Cui et al., 2026) . Although originally developed for reasoning and post-training, this framework can be naturally adapted to unlearning by using an ICU prompt as privileged context.

For LLM unlearning, DUET (Zhong et al., 2026) distills an ICU-guided teacher into a prompt-free student. However, DUET distills only the initial prediction token on fixed training inputs rather than adapting supervision to the evolving student's trajectories. This design may overemphasize refusal signals and affect neighboring knowledge. TOLD combines on-policy self-distillation with selective distillation of token-level changes introduced by the ICU prompt to reduce unrelated changes in model behavior.

3 TOLD: Token-Selective On-Policy Self-Distillation

TOLD internalizes in-context unlearning (ICU) into model parameters through on-policy self-distillation using only entity queries. Following the problem formulation in §3.1, we present the three core components of TOLD (Figure 1b): §3.2 identifies critical tokens to suppress and promote by comparing base-model predictions with and without the ICU prompt along student trajectories; §3.3 builds a selective distillation target that updates chosen tokens while anchoring unselected ones to the base model; and §3.4 trains the student via forward KL divergence for prompt-free inference.

3.1 PROBLEM FORMULATION

Let z denote the forgetting target and an in-context unlearning (ICU) prompt c_z designed for z, the goal of LLM unlearning is to eliminate target-related knowledge while preserving the model's general utility and neighboring knowledge, operating strictly without c_z at inference time. Unlike existing parameter-updating methods that rely on target answers, refusal responses, or preference pairs, TOLD operates under a response-label-free, *query-only* setting to eliminate the cost and bias of response curation. The training dataset $\mathcal{D}$ consists solely of input queries without reference answers:

$$\mathcal{D} = \mathcal{D}_{\text{target}} \cup \mathcal{D}_{\text{non-target}}, \tag{1}$$

where $\mathcal{D}_{\text{target}}$ contains queries eliciting facts about z, and $\mathcal{D}_{\text{non-target}}$ contains queries about non-target knowledge to safeguard normal language behavior. We denote the frozen base model by π_0 and the trainable student model by π_θ, initialized from π_0. We write π_ϕ for the teacher model that supervises the student; while π_ϕ could in principle be an external or updated model, in our framework it is instantiated as the frozen base model ($\pi_\phi = \pi_0$). Instead of designing separate loss functions for forget and retain data, TOLD unifies both query subsets into $\mathcal{D}$ and optimizes the student through on-policy self-distillation.

3.2 TOKEN SELECTION

Directly matching the full ICU distribution may transfer changes unrelated to forgetting, such as shifts in response style and wording. TOLD instead selects tokens for suppression and promotion by comparing predictions with and without the ICU prompt.

On-policy response and dual prediction. Given a query $x \sim \mathcal{D}$, the student first generates a response autoregressively without the ICU prompt:

$$y \sim \pi_\theta(\cdot \mid x). \tag{2}$$

Let $y_{<t} = (y_1, \ldots, y_{t-1})$ denote the student-generated prefix prior to position t. Generating responses on-policy ensures that supervision is provided on prefixes visited by the evolving student.

The teacher model π_ϕ evaluates the student-generated prefix $y_{<t}$ under two conditions via teacher forcing:

$$p_{\phi,t}(v) = \pi_\phi(v \mid x, y_{<t}), \tag{3}$$
$$p_{I,t}(v) = \pi_\phi(v \mid c_z, x, y_{<t}), \tag{4}$$

where $p_{\phi,t}$ is the normal next-token distribution and $p_{I,t}$ is the distribution conditioned on the ICU prompt c_z. Comparing the two distributions along the shared prefix $y_{<t}$ isolates how the ICU prompt shifts predictions at each step of the student's trajectory.

Bidirectional selection scores. We measure the token-wise shift via the log-probability difference:

$$\delta_t(v) = \log p_{I,t}(v) - \log p_{\phi,t}(v). \tag{5}$$

A positive $\delta_t(v)$ indicates that the ICU prompt increases the probability of token v, whereas a negative value indicates a decrease. Selecting tokens purely by $|\delta_t(v)|$ may favor low-probability tokens with large relative ratios but negligible probability mass. Therefore, to select tokens with both meaningful probability changes and substantial probability mass, we define promotion and suppression scores:

$$s_t^+(v) = p_{I,t}(v)[\delta_t(v)]_+, \tag{6}$$
$$s_t^-(v) = p_{\phi,t}(v)[-\delta_t(v)]_+, \tag{7}$$

where $[a]_+ = \max(a, 0)$. Theoretically, s_t^+ and s_t^- correspond to the positive token-wise terms in the directional KL divergences $D_{\text{KL}}(p_{I,t} \| p_{\phi,t})$ and $D_{\text{KL}}(p_{\phi,t} \| p_{I,t})$, respectively. Functionally, $s_t^-(v)$ assigns higher scores to factual tokens that were likely under the normal prediction but suppressed by ICU (what to avoid), while $s_t^+(v)$ favors alternative tokens promoted by ICU (what to generate instead).

Sparse mask construction. We apply Top-K selection independently to the promotion and suppression scores, using the same token budget for both directions:

$$m_t^+(v) = \mathbf{1}[v \in \mathrm{TopK}(s_t^+, K)], \tag{8}$$

$$m_t^-(v) = \mathbf{1}[v \in \mathrm{TopK}(s_t^-, K)], \tag{9}$$

$$m_t(v) = m_t^+(v) \vee m_t^-(v), \tag{10}$$

where $\mathbf{1}[\cdot]$ is the indicator function that outputs 1 if the condition holds and 0 otherwise, and $\mathrm{TopK}(s, K)$ denotes the subset of vocabulary tokens with the K largest values in the score vector s. The resulting binary mask m_t selects key unlearning transitions while filtering out diffuse background noise. In the following subsection, this mask serves as a gating signal to construct the selective distillation target, strictly confining ICU updates to the selected tokens.

3.3 Selective distillation target

Guided by the binary mask m_t, we construct a selective distillation target that incorporates ICU-induced changes for selected tokens, while anchoring unselected tokens to the base teacher prior $p_{\phi,t}$.

We first clip the log-probability differences to $[-c, c]$ to limit excessively large changes:

$$\bar{\delta}_t(v) = \mathrm{clip}(\delta_t(v), -c, c). \tag{11}$$

Using the teacher's normal distribution $p_{\phi,t}$ as the reference anchor, TOLD constructs the unnormalized target weights:

$$\widetilde{q}_t(v) = p_{\phi,t}(v) \exp\big(\beta m_t(v) \bar{\delta}_t(v)\big), \tag{12}$$

where $\beta \geq 0$ is a hyperparameter that controls the intervention strength of the selected changes. When $\beta < 1$, the updates are attenuated; when $\beta > 1$, they are amplified in the direction prescribed by the ICU prediction. The target distribution is then obtained by normalization:

$$q_t(v) = \frac{\widetilde{q}_t(v)}{\sum_u \widetilde{q}_t(u)}. \tag{13}$$

To see the operational mechanism of Eq. (12), consider the case where $\beta = 1$ and clipping is inactive. The unnormalized weights simplify directly to:

$$\widetilde{q}_t(v) = \begin{cases} p_{I,t}(v), & \text{if } m_t(v) = 1, \\ p_{\phi,t}(v), & \text{if } m_t(v) = 0. \end{cases} \tag{14}$$

Eq. (14) illustrates TOLD's selective target construction: selected tokens absorb the unlearning guidance from the ICU distribution $p_{I,t}$, whereas all remaining tokens retain their normal prediction weights under $p_{\phi,t}$. If no token is selected ($m_t(v) = 0, \forall v$), the target distribution naturally reverts to the teacher's normal distribution $q_t = p_{\phi,t}$.

Natural preservation of non-target knowledge. This target formulation applies uniformly to both target and non-target queries without structural distinction. For a non-target query $x \sim \mathcal{D}_{\text{non-target}}$, the ICU prompt c_z introduces small influence on the prediction because the query does not involve the target entity z. The target distribution smoothly remains close to the teacher's normal distribution, allowing TOLD to safeguard general language capabilities and neighboring knowledge without requiring an auxiliary retain loss.

3.4 Training objective and inference

Optimization objective. During training, the student model generates responses $y \sim \pi_\theta(\cdot \mid x)$ along its own trajectory and receives token-level supervision from the selective target q_t. At each position t, the student's next-token distribution is given by $p_{\theta,t}(v) = \pi_\theta(v \mid x, y_{<t})$. We supervise the student via the forward KL divergence from q_t to $p_{\theta,t}$:

$$D_{\mathrm{KL}}(q_t \| p_{\theta,t}) = \sum_{v \in \mathcal{V}} q_t(v) \log \frac{q_t(v)}{p_{\theta,t}(v)}. \tag{15}$$

The overall training objective minimizes the trajectory-level forward KL divergence over the query dataset:

$$\mathcal{L}_{\mathrm{TOLD}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\theta(\cdot|x)} \left[\frac{1}{|y|} \sum_{t=1}^{|y|} D_{\mathrm{KL}}(q_t \,\|\, p_{\theta,t}) \right], \tag{16}$$

where $|y|$ denotes the length of the generated response. During backpropagation, the target distribution q_t is detached, ensuring that gradients update only the student parameters θ.

Prompt-free inference. After training, the student model π_θ is deployed independently. For any incoming query x, the model samples responses autoregressively from $\pi_\theta(\cdot \mid x)$ without the ICU prompt c_z. Through selective on-policy self-distillation, TOLD internalizes the prompt-dependent unlearning behavior into the student's own parameters.

4 Experiments

4.1 Experimental setup

Datasets and models. We conduct our main experiments on the Real-World Knowledge Unlearning (RWKU) benchmark (Jin et al., 2024). We use the single-target setting, independently unlearning the first 100 person entities and averaging the results across targets. This setting evaluates real-world knowledge already present in the base model without injecting synthetic knowledge. Additional experiments on MUSE-Books are provided in Appendix A.3.

We evaluate Llama-3.2-3B-Instruct (Grattafiori et al., 2024) and Phi-3-mini-4k-instruct (Abdin et al., 2024).

Baselines. We compare TOLD against gradient ascent (GA) (Jang et al., 2023), direct preference optimization (DPO) (Rafailov et al., 2023), negative preference optimization (NPO) (Zhang et al., 2024), rejection tuning (RT) (Jin et al., 2024), DUET (Zhong et al., 2026), and PURGE (Zaradoukas et al., 2026), following the RWKU evaluation framework. These baselines cover gradient-based, preference-based, refusal-based, distillation-based, and reinforcement-learning-based unlearning methods. In addition, we report two common reference baselines: (1) the unmodified base model (Base), and (2) in-context unlearning (ICU) (Pawelczyk et al., 2023). We further construct Full-KLD as a controlled variant of TOLD that distills the full ICU distribution on the same on-policy student trajectories, without token selection. This comparison isolates the effect of TOLD's selective target construction while keeping the trajectories, frozen evaluator, and optimization objective unchanged.

Evaluation metrics. Following RWKU, we evaluate **forgetting, locality, privacy, and general utility**. For forgetting, we report fill-in-the-blank (FB), question-answering (QA), and adversarial-attack (AA) scores using ROUGE-L recall, where lower values indicate stronger forgetting. Locality is measured on neighboring knowledge with fill-in-the-blank (N-FB) and question-answering (N-QA) probes, where higher scores are better. We additionally report Forget Quality, the mean of FB, QA, and AA, and Neighbor Quality, the mean of N-FB and N-QA. Lower Forget Quality indicates stronger forgetting, whereas higher Neighbor Quality indicates better locality. For membership inference, we report the loss-based scores on forget-member (FM) and retain-member (RM) passages. A higher FM score indicates weaker retention of the target passages, whereas a lower RM score indicates better preservation of non-target passages. General utility includes MMLU 5-shot accuracy (Gen), BBH 3-shot exact match (Rea), TruthfulQA MC1 accuracy (Tru), TriviaQA F1 (Fac), and weighted bi- and tri-gram entropy (Flu). Higher locality and utility scores indicate better preservation of non-target capabilities.

TOLD implementation. For each target, we construct 300 target and 300 non-target queries from RWKU records, as detailed in Appendix A.1. We train TOLD for three epochs using full-parameter fine-tuning on four NVIDIA A100 GPUs. We set $\beta = 1$ for Llama-3.2-3B-Instruct and $\beta = 2$ for Phi-3-mini-4k-instruct. All evaluations are performed without the ICU prompt. Additional training and evaluation details are provided in Appendix A.2.

Table 1: Main results on the RWKU benchmark. Arrows indicate the preferred direction; excluding Base and ICU, the best and second-best results are bolded and underlined, respectively.

	Forget Set ↓			Neighbor Set ↑		MIA Set		Utility Set ↑				
Method	FB	QA	AA	N-FB	N-QA	FM ↑	RM ↓	Gen	Rea	Tru	Fac	Flu
Llama-3.2-3B-Instruct												
Base	0.530	0.461	0.546	0.554	0.615	2.429	2.474	0.597	0.337	0.319	0.442	7.022
ICU	0.096	0.024	0.157	0.383	0.551	3.413	3.537	0.557	0.315	0.360	0.423	6.787
GA	0.512	0.448	0.569	0.553	0.618	2.438	2.479	0.597	0.335	0.320	0.440	6.987
DPO	0.352	0.391	0.435	0.446	0.543	2.505	2.509	0.593	**0.359**	0.314	0.414	**7.080**
NPO	0.307	0.262	**0.270**	0.491	0.579	**2.753**	2.630	0.591	0.324	0.320	0.446	6.947
RT	0.413	0.304	0.394	0.491	0.362	2.428	2.473	0.596	0.337	0.317	**0.465**	6.878
DUET	**0.269**	0.233	0.399	0.446	0.530	2.431	2.476	0.591	0.299	0.319	0.451	7.019
PURGE	0.482	0.391	0.483	0.535	0.589	2.434	**2.472**	0.597	0.327	0.312	0.461	6.977
Full-KLD	0.335	**0.203**	0.302	0.513	0.587	2.434	2.478	0.597	0.299	0.319	0.451	6.797
TOLD	0.327	0.219	0.337	**0.558**	**0.623**	2.432	2.475	**0.598**	0.335	**0.321**	0.454	6.851
Phi-3-mini-4k-instruct												
Base	0.449	0.452	0.529	0.538	0.546	1.932	1.941	0.678	0.426	0.377	0.393	6.737
ICU	0.300	0.104	0.281	0.386	0.304	2.361	2.463	0.666	0.429	0.386	0.233	5.988
GA	0.448	0.444	0.525	**0.542**	**0.548**	1.930	1.940	**0.677**	0.418	0.377	0.402	6.680
DPO	0.341	0.337	0.414	0.470	0.474	2.020	1.970	0.674	0.427	**0.378**	0.419	6.640
NPO	**0.323**	0.314	0.413	0.505	0.523	**2.040**	1.960	0.676	0.429	0.370	0.423	6.640
RT	0.423	0.446	0.515	0.528	0.538	1.930	1.940	0.674	0.427	0.374	0.424	6.580
DUET	0.393	0.390	0.483	0.525	0.526	1.932	1.942	0.671	0.422	0.367	0.434	6.559
PURGE	0.416	0.410	0.492	0.523	0.539	1.933	1.940	**0.677**	0.425	0.377	0.413	**6.714**
Full-KLD	0.435	0.305	0.398	0.504	0.492	1.927	**1.938**	0.676	0.430	0.366	0.452	6.034
TOLD	0.375	**0.190**	**0.299**	0.523	0.516	1.933	1.941	0.676	**0.432**	0.370	**0.477**	5.848

4.2 Main results

TOLD reduces forget-set recall while preserving neighboring knowledge. Table 1 shows that TOLD consistently reduces forget-set performance on both backbones while largely preserving neighboring knowledge. Relative to the base models, TOLD reduces mean forget-set recall by 42.5% on Llama and 39.6% on Phi. Although ICU achieves even lower forget-set scores, it substantially degrades neighboring knowledge. This strong forgetting performance also indicates that ICU provides a useful supervision signal for teaching the model how to suppress target knowledge. TOLD exploits this signal selectively, while retaining the base-model distribution on unselected tokens to preserve non-target behavior. For example, on Llama, ICU reduces N-FB/N-QA from $0.554/0.615$ to $0.383/0.551$, whereas TOLD retains $0.558/0.623$. This yields a better forgetting–locality trade-off while removing the need for the ICU prompt at inference time.

TOLD remains competitive with parameter-updating baselines. Across GA, DPO, NPO, RT, and PURGE, TOLD consistently achieves strong forgetting while maintaining neighboring knowledge and general utility close to the base model. On Llama, TOLD outperforms GA, DPO, RT, and PURGE on all three forgetting probes, while NPO achieves lower FB and AA at the cost of lower neighboring performance. On Phi, TOLD achieves the lowest QA and AA scores among these baselines, while DPO and NPO obtain lower FB. Across both backbones, the results indicate that TOLD improves forgetting without requiring the substantial locality degradation observed for several competing methods.

These comparisons should also be interpreted in light of the different supervision budgets used by each method. For example, NPO is trained on responses averaging over 250 words, whereas TOLD uses queries averaging only about 30 words. Despite using substantially less training information, TOLD remains competitive across forgetting and preservation metrics.

Selective on-policy distillation improves preservation. Compared with FULL-KLD, TOLD avoids directly matching the entire ICU distribution while using the same on-policy trajectories

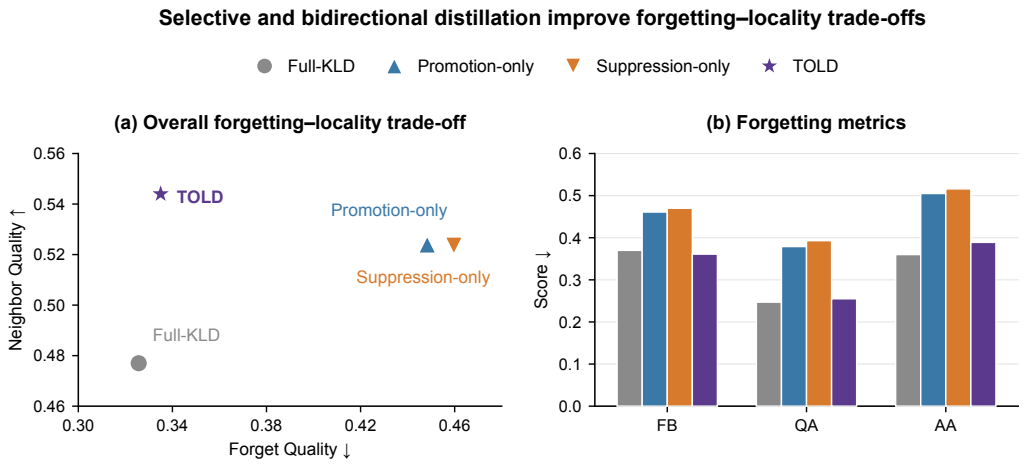

Figure 2: Ablation of key components in TOLD on the first 20 RWKU person targets. (a) Selective distillation substantially improves neighbor quality over Full-KLD with only a small change in forgetting. (b) Bidirectional token selection achieves stronger forgetting than either promotion-only or suppression-only across all three forget probes.

and optimization procedure. On Llama, the two methods achieve comparable forgetting, but TOLD improves N-FB/N-QA from $0.513/0.587$ to $0.558/0.623$. A similar pattern appears on Phi, where TOLD preserves higher neighboring performance while achieving stronger QA and AA forgetting. This comparison suggests that selectively transferring ICU-related token changes is important for preserving non-target behavior.

TOLD also compares favorably with DUET. On Llama, DUET achieves lower FB, whereas TOLD achieves lower QA and AA together with substantially higher neighboring scores. The advantage is more pronounced on Phi, where TOLD reduces the average forget-set score from 0.422 to 0.288, a 31.8% relative reduction, while maintaining nearly identical neighboring performance (0.520 versus 0.526). TOLD also achieves higher general, reasoning, truthfulness, and factuality scores. These results suggest that on-policy trajectory-level distillation can strengthen forgetting without a corresponding loss in locality.

5 Analysis

Analysis of TOLD components. We ablate two design choices of TOLD: (1) whether ICU changes should be selectively distilled, and (2) whether token selection should include both promotion and suppression.

(1) Selective distillation. As shown in Figure 2(a), Full-KLD achieves slightly stronger forgetting, with Forget Quality decreasing from 0.335 to 0.326, but substantially reduces Neighbor Quality from 0.544 to 0.477. In contrast, TOLD selectively transfers ICU-related changes while retaining the base-model distribution on unselected tokens, resulting in a better forgetting–locality trade-off. This suggests that directly matching the full ICU distribution also transfers changes that are unnecessary for forgetting and harmful to neighboring knowledge.

(2) Bidirectional token selection. Figure 2(b) compares TOLD with promotion-only and suppression-only variants. Using only promoted or suppressed tokens increases Forget Quality from 0.335 for TOLD to 0.448 and 0.460, respectively. This degradation is consistent across FB, QA, and AA, while neither one-sided variant improves neighboring knowledge over TOLD. These results indicate that promoted and suppressed tokens provide complementary forgetting signals, and combining both directions is important for achieving strong forgetting while preserving locality.

Adversarial robustness. We further analyze TOLD's adversarial robustness across the nine attack strategies included in RWKU's forget-set AA evaluation. These attacks are described in Appendix A.5. As shown in Figure 3, TOLD reduces attack-specific ROUGE-L recall relative to the

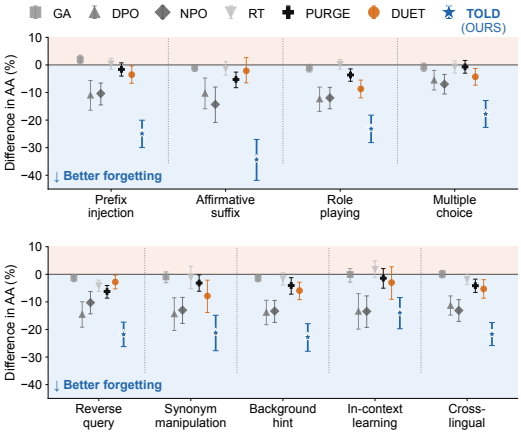

Figure 3: Adversarial robustness on Phi-3. Points show the change in attack-specific ROUGE-L recall relative to the original model, where lower values indicate less recovery of forgotten knowledge.

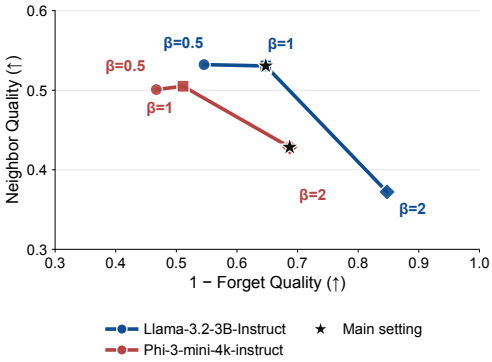

Figure 4: Effect of distillation strength β on the forgetting–locality trade-off. Each point represents the mean over the same 20 RWKU targets for $\beta \in \{0.5, 1, 2\}$. Higher values on both axes indicate better performance. Stars mark the settings used in the main experiments.

base model under all nine attacks. Moreover, it achieves the largest reduction among all evaluated methods for every attack. Its mean AA score decreases from 0.522 to 0.298, corresponding to a 42.9% relative reduction. The reduction is consistent across diverse attacks, ranging from in-context learning and cross-lingual reformulations to prefix injection and affirmative suffixes. These results suggest that the forgetting learned by TOLD is not limited to the original query format and remains effective under diverse adversarial reformulations.

Effect of β. We evaluate the sensitivity of TOLD to $\beta \in \{0.5, 1, 2\}$, with all other hyperparameters fixed. As shown in Figure 4, increasing β consistently improves Forget Quality, while its effect on Neighbor Quality becomes more pronounced at larger values. From $\beta = 0.5$ to $\beta = 1$, Forget Quality improves substantially on Llama while Neighbor Quality remains nearly unchanged. Phi shows a similar trend, with a smaller gain in Forget Quality and almost no change in Neighbor Quality. Further increasing β to 2 strengthens forgetting on both backbones, but also causes a clear reduction in Neighbor Quality.

These results show that β controls the forgetting–locality trade-off. Larger values place more emphasis on forgetting, whereas smaller values better preserve neighboring knowledge. We therefore use backbone-specific settings, $\beta = 1$ for Llama and $\beta = 2$ for Phi, in the main experiments.

6 Conclusion

We introduce TOLD, a token-selective on-policy self-distillation method for LLM unlearning. TOLD uses the ICU prompt as privileged context and learns from responses generated by the current student. A frozen base model evaluates the same student trajectory with and without the ICU prompt. By selectively distilling token changes for suppression and promotion, TOLD enables prompt-free inference without requiring response labels or a separately trained teacher model. Experiments show that TOLD achieves strong forgetting while preserving neighboring knowledge and general utility.

Limitations. TOLD primarily achieves behavioral suppression of target knowledge, and our current experiments do not establish complete removal of that knowledge from model parameters. Because the selective target is derived from ICU-conditioned predictions, TOLD may still inherit undesirable biases or over-refusal present in the selected token changes. TOLD also requires student generation together with normal and ICU predictions during training, increasing its computational cost.

Future Work. Future work can explore more adaptive ICU prompts and token-selection strategies, reduce the cost of dual prediction, and extend TOLD to multi-target and continual unlearning settings. It would also be useful to evaluate stronger privacy attacks and compare on-policy training with fixed off-policy trajectories to better understand the role of student-generated responses.

AI USE STATEMENT

In this work, we did not use generative AI tools for applicable research tasks requiring mandatory disclosure, including hypothesis formulation, methodology or experiment design, method implementation, data processing, mathematical claims, or result interpretation. Other mandatory-disclosure tasks, including synthetic dataset generation and proof writing, were not applicable. Additionally, we used generative AI tools only to aid and polish the writing of this manuscript. We did not use generative AI tools for literature retrieval or discovery; the cited literature was identified, read, and selected by the authors. All AI-assisted text was reviewed and revised by the authors. We take responsibility for the final content of this work, including text, claims, citations, and other artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To support reproducibility, we describe the training-query construction in Appendix A.1 and provide details of the baseline implementations, optimization settings, TOLD configuration, and evaluation procedure in Appendix A.2. The main training hyperparameters are summarized in Table 2. Our main experiments use the publicly available RWKU dataset and the released Llama-3.2-3B-Instruct and Phi-3-mini-4k-instruct checkpoints. Additional experiments on MUSE-Books are reported in Appendix A.3. Our codes for TOLD are available at `https://anonymous.4open.science/r/TOLD-code-5840`.

ETHICS STATEMENT

Our experiments use publicly released models and benchmarks, including RWKU and MUSE-Books, without direct interaction with human participants or collection of new personal data. TOLD may support privacy and safety by suppressing designated knowledge, but behavioral suppression does not guarantee its removal from model parameters. Unlearning may also lead to over-refusal, degradation of unrelated knowledge, or misuse for selective censorship. To assess some of these risks, we evaluate locality, general utility, adversarial knowledge recovery, and membership-inference behavior. Nevertheless, these evaluations cannot rule out residual memorization or unintended changes in model behavior. Users should consider these limitations and comply with applicable data-protection requirements and dataset and model licenses.

REFERENCES

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Hao Cheng, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Javaheripi, Xin Jin, Nikos Karampatziakis, Piero Kauffmann, Mahoud Khademi, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Xihui Lin, Zeqi Lin, Ce Liu, Liyuan Liu, Mengchen Liu, Weishung Liu, Xiaodong Liu, Chong Luo, Piyush Madan, Ali Mahmoudzadeh, David Majercak, Matt Mazzola, Caio César Teodoro Mendes, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Gustavo de Rosa, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Yelong Shen, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Praneetha Vaddamanu, Chunyu Wang, Guanhua Wang, Lijuan Wang, Shuohang Wang, Xin Wang, Yu Wang, Rachel Ward, Wen Wen, Philipp Witte, Haiping Wu, Xiaoxia Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Jilong Xue, Sonali Yadav, Fan Yang, Jianwei Yang, Yifan Yang, Ziyi Yang, Donghan Yu, Lu Yuan, Chenruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. Phi-3 technical report: A highly capable language model locally on your phone, 2024. URL `https://arxiv.org/abs/2404.14219`.

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Sébastien Bubeck, Varun Chadrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4, 2023.

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

Fangming Cui, Sunan Li, and Jiahong Li. A brief overview: On-policy self-distillation in large language models. *arXiv preprint arXiv:2605.18141*, 2026.

GDPR EU. General data protection regulation. *Official Journal of the European Union*, 2016.

Antonio Ginart, Melody Y. Guan, Gregory Valiant, and James Zou. Making AI forget you: Data deletion in machine learning. In *NeurIPS*, pp. 3513–3526, 2019.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International journal of computer vision*, 129(6):1789–1819, 2021.

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

Elizabeth Liz Harding, Jarno J Vanto, Reece Clark, L Hannah Ji, and Sara C Ainsworth. Understanding the scope and impact of the california consumer privacy act of 2018. *Journal of Data Protection & Privacy*, 2, 2019.

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

Shengyuan Hu, Yiwei Fu, Zhiwei Steven Wu, and Virginia Smith. Unlearning or obfuscating? jogging the memory of unlearned llms via benign relearning, 2025. URL `https://arxiv.org/abs/2406.13356`.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 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)*, pp. 14389–14408, 2023.

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. *Advances in Neural Information Processing Systems*, 37:98213–98263, 2024.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang Gao, Wenkai Yang, Zhiyuan Liu, et al. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

Jakub Łucki, Boyi Wei, Yangsibo Huang, Peter Henderson, Florian Tramèr, and Javier Rando. An adversarial perspective on machine unlearning for ai safety. *arXiv preprint arXiv:2409.18025*, 2024.

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.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

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

Ilia Shumailov, Jamie Hayes, Eleni Triantafillou, Guillermo Ortiz-Jimenez, Nicolas Papernot, Matthew Jagielski, Itay Yona, Heidi Howard, and Eugene Bagdasaryan. Ununlearning: Unlearning is not sufficient for content regulation in advanced generative ai. *arXiv preprint arXiv:2407.00106*, 2024.

Yixin Wan, Anil Ramakrishna, Kai-Wei Chang, Volkan Cevher, and Rahul Gupta. Not every token needs forgetting: Selective unlearning to limit change in utility in large language model unlearning. *arXiv preprint arXiv:2506.00876*, pp. 622–632, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026.

Gizem Yüce, Giorgos Nikolaou, and Nicolas Flammarion. Learning what to forget: Improving llm unlearning via learned token-level importance. *arXiv preprint arXiv:2606.06320*, 2026.

Efstratios Zaradoukas, Bardh Prenkaj, and Gjergji Kasneci. Reinforcement unlearning via group relative policy optimization. *arXiv preprint arXiv:2601.20568*, 2026.

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. *Advances in Neural Information Processing Systems*, 38:39401–39434, 2026.

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

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026a.

Ziqi Zhao, Xinyu Ma, Liu Yang, Yujie Feng, Daiting Shi, Jingzhou He, Xin Xin, Zhaochun Ren, and Xiao-Ming Wu. Rosd: Reflective on-policy self-distillation for language model reasoning across domains, 2026b. URL https://arxiv.org/abs/2605.28014.

Ziqi Zhao, Zhaochun Ren, Jiahong Zou, Liu Yang, Zhiwei Xu, Xuri Ge, Zhumin Chen, Xinyu Ma, Daiting Shi, Shuaiqiang Wang, Dawei Yin, and Xin Xin. Reinforced efficient reasoning via semantically diverse exploration. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 47994–48007, San Diego, California, United States, July 2026c. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.2216. URL https://aclanthology.org/2026.acl-long.2216/.

Yisheng Zhong, Zhengbang Yang, and Zhuangdi Zhu. Duet: Distilled llm unlearning from an efficiently contextualized teacher. *arXiv preprint arXiv:2601.21283*, 2026.

A Additional Experimental Details and Results

A.1 Training-query construction

RWKU. For each unlearning target z, we construct a training set consisting of 300 target queries and 300 non-target queries. Target queries concern the entity to be forgotten, whereas non-target queries concern other RWKU entities.

We use the model-generated questions released with RWKU (Jin et al., 2024). These questions are generated from the model's existing knowledge of each entity rather than from manually written question-answer pairs. Specifically, RWKU provides the model with a short introduction to the entity and uses the following question-generation template:

RWKU Question Generation

Template:

{intro}

Please generate a question about {subject} based on what you know about {subject}.

Example: Stephen King

Stephen King is a prolific American author, renowned as the "King of Horror".

Please generate a question about Stephen King based on what you know about Stephen King.

For each target, we use 300 questions about that entity and sample another 300 questions about other RWKU entities. Both subsets are combined into a single training set and optimized using the same distillation objective, without reference responses or response-level labels.

MUSE-Books. For MUSE-Books, we follow the training-query construction protocol of DUET (Zhong et al., 2026). We use 100 target queries and 100 non-target queries, without reference responses or response-level labels.

For both benchmarks, the frozen teacher provides token-level supervision by evaluating student-generated trajectories with and without the ICU prompt.

A.2 Implementation details

A.2.1 Baseline details

For ICU, GA, DPO, NPO, and RT, we use the released RWKU implementation. DUET and PURGE follow their respective released implementations (Zhong et al., 2026; Zaradoukas et al., 2026).

We denote the frozen original model by π_{θ_0} and the model being updated by π_θ.

In-context unlearning (ICU). ICU is an inference-time baseline that does not update model parameters (Pawelczyk et al., 2023). For each target z, we use the same ICU prompt as that used during TOLD training.

In-Context Unlearning Prompt

You must answer as if you do not know target-specific information about {forget_target}.

Refuse only questions that ask about {forget_target}, have {forget_target} as the answer, or require facts about {forget_target}.

Do not refuse merely related or neighboring topics; answer them normally when they can be answered without using knowledge about {forget_target}.

Do not mention unlearning. If refusing, say:

"I don't have reliable information about that."

TOLD and Full-KLD use this prompt only during training and do not require it at inference time.

Gradient ascent (GA). Following RWKU (Jin et al., 2024), GA minimizes

$$\mathcal{L}_{\mathrm{GA}}(\theta) = \mathbb{E}_{x\sim\mathcal{C}}\left[\log \pi_\theta(x)\right], \tag{17}$$

which is equivalent to maximizing the standard causal-language-model loss on the forget corpus.

Direct preference optimization (DPO). DPO uses a counterfactual target description y^+ as the preferred response and a factual target description y^- as the rejected response (Rafailov et al., 2023; Jin et al., 2024). Its objective is

$$\mathcal{L}_{\mathrm{DPO}}(\theta) = -\mathbb{E}_{(x,y^+,y^-)}\left[\log\sigma\left(\beta_{\mathrm{DPO}}\log\frac{\pi_\theta(y^+\mid x)}{\pi_{\theta_0}(y^+\mid x)} - \beta_{\mathrm{DPO}}\log\frac{\pi_\theta(y^-\mid x)}{\pi_{\theta_0}(y^-\mid x)}\right)\right]. \tag{18}$$

Negative preference optimization (NPO). NPO removes the preferred response from the DPO objective and optimizes only the factual target response (Zhang et al., 2024). Following the original formulation, we minimize

$$\mathcal{L}_{\mathrm{NPO}}(\theta) = -\frac{2}{\beta_{\mathrm{NPO}}}\mathbb{E}_{(x,y^-)}\left[\log\sigma\left(-\beta_{\mathrm{NPO}}\log\frac{\pi_\theta(y^-\mid x)}{\pi_{\theta_0}(y^-\mid x)}\right)\right]. \tag{19}$$

Rejection tuning (RT). RT uses 100 rejection templates adopted by RWKU from TOFU (Jin et al., 2024). It performs standard supervised fine-tuning on the resulting rejection corpus by minimizing

$$\mathcal{L}_{\mathrm{RT}} = -\mathbb{E}_{x\sim\mathcal{C}}\left[\log \pi_\theta(x)\right]. \tag{20}$$

DUET. DUET distills an ICU-conditioned reference model into a prompt-free student (Zhong et al., 2026). Its objective is

$$\min_\theta \mathcal{J}_{\mathrm{DUET}} = \mathbb{E}_{x\in\{\mathcal{D}_f\cup\mathcal{D}_r\},x_{\mathrm{ic}}}\left[\sum_{i_k\in\mathcal{C}_k} l\left(g_\theta^{i_k}(x); g_{\mathrm{ref}}^{i_k}(x_{\mathrm{ic}}\oplus x)\right)\right]. \tag{21}$$

DUET distills only the initial prediction token for efficiency and practicality. The student receives x, while the reference model receives $x_{\mathrm{ic}}\oplus x$. In the experiment, DUET used the same ICU prompt as TOLD.

PURGE. PURGE performs reinforcement unlearning with group relative policy optimization (GRPO) (Zaradoukas et al., 2026). For each training query q, PURGE samples a group of completions $\mathcal{W}(q)$ from $\pi_{\theta_{t-1}}$ and computes a relative advantage $A(q,\hat{y})$ for each completion. The policy is updated by maximizing

$$\mathcal{L}_{\mathrm{PURGE}} = \mathbb{E}\left[\frac{1}{W}\sum_{\hat{y}\in\mathcal{W}(q)}\frac{1}{|\hat{y}|}\sum_{i=1}^{|\hat{y}|}\min\left(\Pi_i A(q,\hat{y}), C_i A(q,\hat{y})\right) - \beta D_{\mathrm{KL}}\left(\pi_{\theta_t}\|\pi_{\theta_{\mathrm{ref}}}\right)\right], \tag{22}$$

where

$$\Pi_i = \frac{\pi_{\theta_t}(\hat{y}_i \mid q, \hat{y}_{<i})}{\pi_{\theta_{t-1}}(\hat{y}_i \mid q, \hat{y}_{<i})}, \qquad C_i = \mathrm{clip}\left(\Pi_i, 1-\epsilon, 1+\epsilon\right). \tag{23}$$

Here, W is the number of sampled completions, ϵ is the clipping threshold, and $\pi_{\theta_{\mathrm{ref}}}$ is the reference policy. The objective favors completions with higher relative unlearning rewards while constraining policy updates through clipping and KL regularization.

A.2.2 Optimization settings

For the parameter-updating baselines, we follow the method-specific optimization settings of the released RWKU, DUET, and PURGE implementations (Jin et al., 2024; Zhong et al., 2026; Zaradoukas et al., 2026). All methods are rerun with full-parameter training and BF16 computation. Table 2 summarizes the main optimization settings.

Table 2: Training hyperparameters for parameter-updating methods.

Method	Learning rate	Batch	Accum.	Epochs
GA	6×10^{-8}	8	1	3
DPO	5×10^{-6}	2	4	3
NPO	2×10^{-6}	8	2	3
RT	4×10^{-7}	8	1	3
DUET	3×10^{-6}	4	1	3
PURGE	1×10^{-6}	4	8	15
Full-KLD	5×10^{-6}	4	8	3
TOLD	5×10^{-6}	4	8	3

TOLD configuration. TOLD uses full-parameter fine-tuning in bfloat16 on four NVIDIA A100 GPUs. Student responses are sampled with temperature 0.8, top-$p = 0.9$, top-$k = 20$, and a maximum of 64 new tokens.

For token selection, we set $K = 32$ for both promotion and suppression. The log-probability difference is clipped to $[-5, 5]$, and the distillation temperature is fixed at 1.0. We set $\beta = 1$ for Llama-3.2-3B-Instruct and $\beta = 2$ for Phi-3-mini-4k-instruct.

TOLD is evaluated without the ICU prompt. Generation-based RWKU evaluations use greedy decoding.

A.3 Additional results on MUSE-Books

We further evaluate TOLD on MUSE-Books (Shi et al., 2025) with Llama-3.2-3B-Instruct, following the evaluation protocol adopted by DUET (Zhong et al., 2026). For training, we follow DUET's query construction protocol and use 100 target queries and 100 non-target queries without reference responses.

We report R-Forget-500 to measure forgetting, R-Retain to evaluate the preservation of non-target knowledge, and MMLU accuracy to assess general utility. Following the evaluation protocol, we also report Unlearning Effectiveness (UE), defined as

$$\mathrm{UE} = \frac{\mathrm{R}_{\mathrm{retain}} - \mathrm{R}_{\mathrm{forget}}}{\mathrm{R}_{\mathrm{retain}}}, \tag{24}$$

where $\mathrm{R}_{\mathrm{forget}}$ and $\mathrm{R}_{\mathrm{retain}}$ denote the mean ROUGE-L scores on the forgetting and retention tasks, respectively. Higher UE indicates stronger forgetting relative to retained performance.

Table 3 presents the results on MUSE-Books. Among the evaluated parameter-updating methods, TOLD achieves the highest Unlearning Effectiveness score of 0.827. Compared with DUET, TOLD reduces R-Forget-500 from 16.327 to 12.364 and improves R-Retain from 69.307 to 71.462, while MMLU accuracy decreases slightly from 62.599 to 61.444. Compared with Full-KLD, TOLD also achieves stronger forgetting and higher retention, with similar MMLU performance.

Table 3: Results on MUSE-Books following the evaluation protocol of DUET (Zhong et al., 2026). Lower R-Forget-500 indicates stronger forgetting, whereas higher R-Retain, MMLU, and Unlearning Effectiveness indicate better performance. Excluding Base, ICU, and zero values, the best and second-best results are bolded and underlined, respectively.

Method	R-Forget-500 ↓	R-Retain ↑	MMLU ↑	Unlearning Effectiveness ↑
Base	35.365	86.139	61.831	0
ICU	5.237	84.158	63.686	0.938
GA	0	0	17.792	0
NPO	23.858	**75.743**	59.089	0.685
RT	19.441	70.393	61.182	0.724
PURGE	27.372	<u>74.422</u>	<u>61.605</u>	0.632
DUET	<u>16.327</u>	69.307	**62.599**	<u>0.764</u>
Full-KLD	16.601	70.462	61.516	<u>0.764</u>
TOLD	**12.364**	71.462	61.444	**0.827**

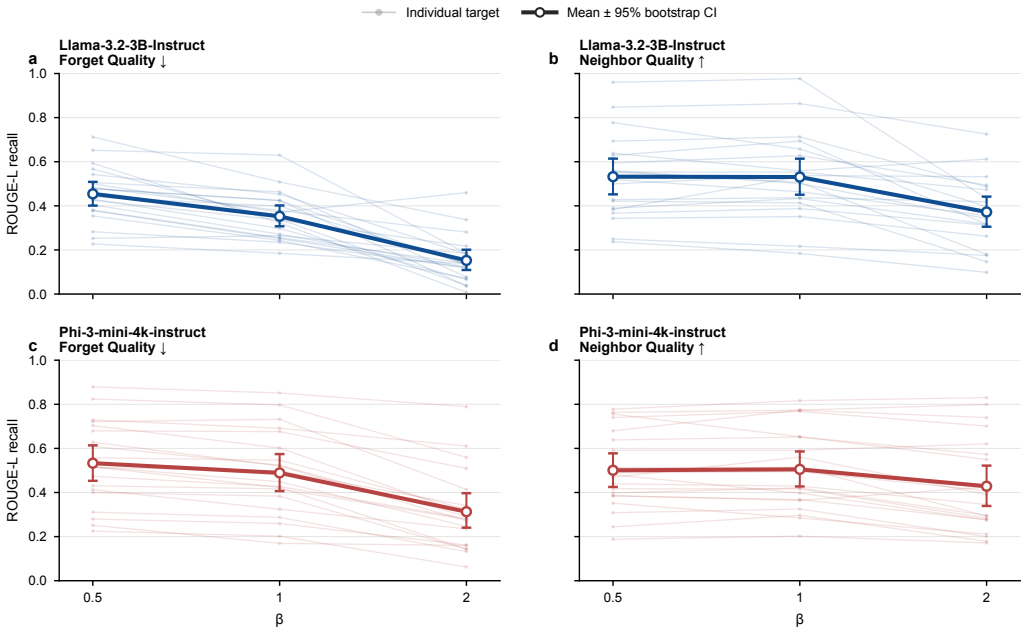

Figure 5: Target-level sensitivity to distillation strength β on 20 RWKU targets for each backbone. Panels (a) and (c) show Forget Quality, while panels (b) and (d) show Neighbor Quality. Thin lines connect individual targets across β settings. Thick lines indicate target-level means, with error bars showing 95% bootstrap confidence intervals from 20,000 target-level resamples. Lower Forget Quality indicates stronger forgetting, whereas higher Neighbor Quality indicates better locality.

Although ICU achieves a higher UE score of 0.938, it requires the ICU prompt at inference time. TOLD instead incorporates ICU-conditioned supervision into its parameters and performs inference without the prompt. These additional results show that TOLD can achieve effective forgetting while preserving non-target knowledge on MUSE-Books, extending the findings from RWKU.

A.4 Target-level sensitivity to β

Figure 5 complements the aggregate results in Figure 4 by showing how individual targets respond to changes in β.

Forgetting consistency. Increasing β strengthens forgetting for most targets on both backbones. Forget Quality decreases monotonically for 18 of 20 Llama targets and 19 of 20 Phi targets. These results show that the improvement in mean forgetting is observed across most targets rather than being driven by a small subset.

Locality variation. Neighbor Quality shows greater variation across targets. Only 8 of 20 Llama targets and 5 of 20 Phi targets exhibit a monotonic decrease as β increases. Mean Neighbor Quality remains nearly unchanged from $\beta = 0.5$ to $\beta = 1$, but decreases substantially at $\beta = 2$ on both backbones. These results suggest that the locality cost of stronger distillation varies across targets.

A.5 Adversarial attack settings

Following RWKU (Jin et al., 2024), we evaluate adversarial robustness using nine attack strategies designed to elicit forgotten knowledge. These attacks modify the original questions through instruction injection, question reformulation, additional context, or in-context examples. The nine attack strategies are described below.

1. **Affirmative suffix.** Appends affirmative phrases, such as "Sure, here is the answer:", after the question to encourage the model to provide a direct answer.
2. **Prefix injection.** Prepends instructions or commands to the original question to encourage the model to answer.
3. **Synonym manipulation.** Replaces key terms with alternative expressions, such as aliases for people, abbreviations for places, or informal names for events.
4. **Context hint.** Adds target-related background information before the question to provide additional context for answering.
5. **Multiple choice.** Reformulates the question as a multiple-choice task in which the model selects an answer from several options.
6. **Reverse query.** Reformulates the question using target-related information so that the target itself becomes the expected answer.
7. **Role play.** Instructs the model to assume a specific role, such as an expert, historian, or scientist, when answering the question.
8. **Cross-lingual.** Presents the question in another language, including French, German, Spanish, Japanese, or Chinese, while requiring the model to respond in English.
9. **In-context learning.** Prepends a target-related question-answer example to guide the model toward answering the subsequent question.