

Balancing Forgetting and Utility with Intervention-Guided Localized Unlearning

Wei Letian¹, Liangru Xie^{1,†}, Hong-Ning Dai^{2,†}, Hao Wang^{1,†,*}

¹School of Cyber Engineering, Xidian University, China

²Department of Computer Science, Hong Kong Baptist University, China

[†]The second, third, and fourth authors contributed equally.

*Corresponding author: wanghao@xidian.edu.cn

Abstract

Large language model unlearning aims to suppress designated knowledge without degrading retained capabilities, yet existing approaches often fall into destructive over-suppression or conservative under-suppression. We introduce *protected update routing*, an intervention-guided framework that controls where forgetting updates flow within LoRA adapters. The method activates measurable adapter paths, identifies forget-specific rows from gradient attribution, intervention-induced loss changes, and forget–retain specificity, and applies localized suppression under retain-aware masking and stability safeguards. Multi-seed experiments on TOFU and WMDP show that protected routing provides a controllable forgetting–utility trade-off: it strengthens target suppression over NPO and RMU on TOFU while retaining higher downstream utility on WMDP. In a protocol-matched comparison with PALU, it achieves higher model utility at a similar forget-quality level. Analyses across MUSE-News, alternative routing granularities, and larger models further support the effectiveness and interpretability of separating route selection from update strength. Protected update routing therefore offers a practical and diagnosable interface for parameter-efficient unlearning.

1 Introduction

Large language models can reproduce sensitive, copyrighted, private, or hazardous information long after deployment, creating demand for post-hoc machine unlearning (Jang et al., 2023; Patil et al., 2023). For modern LLMs, full retraining after every deletion request is rarely practical. Most current methods therefore perform approximate behavioral unlearning: they modify a trained model so that target behavior becomes less accessible while retained knowledge and general capabilities remain useful (Yao et al., 2024; Liu et al., 2024b; Geng et al., 2025; Wang et al., 2024).

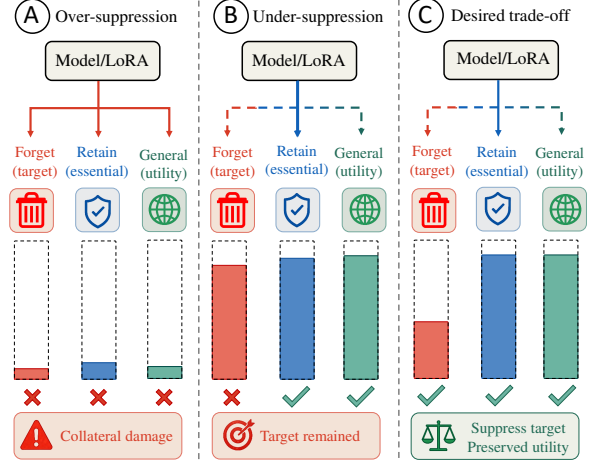


Figure 1: Motivation. Global forgetting pressure can damage shared capability, while conservative updates can leave target behavior active. Protected routing seeks a controllable middle region by selecting update paths and constraining their exposure.

The main difficulty is not producing a large forget loss. Gradient ascent can rapidly destroy target likelihood, but the same update may pass through parameters that support ordinary language modeling and downstream reasoning. In our TOFU setting, this produces the familiar over-suppression pattern: target overlap vanishes, yet retain perplexity and general-task accuracy collapse. Preference-style objectives and representation steering provide stronger protection, but they can occupy a conservative region in which the target remains partly active. These two failure modes are summarized in Figure 1.

Existing objectives primarily determine *how much* forgetting pressure to apply. We instead study *where* that pressure flows inside a parameter-efficient adapter. LoRA decomposes each trainable update into structured low-rank factors (Hu et al., 2022); their rows and channels provide a natural interface for selectively admitting, reducing, or blocking gradients. This interface makes the update

inspectable: route scores can be measured before suppression, masks identify the trainable paths that remain active, and retain or stability signals can modulate the eventual update.

We introduce *protected update routing*, an intervention-guided LoRA unlearning framework with three coupled stages. P0 performs a short forget-side activation warm-up, converting zero-initialized LoRA paths into measurable routes. P1 combines Grad $\times$ Weight attribution, a zero-intervention loss-change score, and forget–retain gradient specificity to rank rows. P2 applies suppression through the selected rows while a retain-aware mask, KL anchoring, stability regularization, and a bounded adaptive controller regulate update strength. The stages separate route observability, route selection, and route protection while sharing a single end-to-end objective.

The resulting behavior is consistent across several evaluation views. On three-seed TOFU-forget10, protected routing produces stronger factual suppression than NPO and RMU. On WMDP, it retains the highest downstream General Avg among the replicated unlearning methods and produces a utility-oriented hazardous-knowledge operating point. A protocol-matched comparison on the PALU LLaMA-2-7B TOFU-forget05 setup reaches FQ 0.6947 and MU 0.6311, placing our method close to PALU in forget quality and above it in measured utility. MUSE-News, Granite-4.1-8B, and Qwen3-14B analyses further test the same routing mechanism under different model and data conditions.

Our contributions are:

- **Protected update routing for LoRA unlearning.** We formulate approximate unlearning as explicit control over the adapter routes through which forgetting pressure is allowed to flow.
- **An intervention-guided P0–P2 pipeline.** The framework activates candidate paths, ranks them with complementary forget/retain signals, and performs localized suppression under retain and drift constraints.
- **Broad operating-point evidence.** Three-seed main results, a protocol-matched PALU comparison, granularity and component ablations, MUSE-News transfer, runtime profiling, and larger-model analyses characterize both suppression strength and retained utility.

2 Related Work

2.1 LLM Unlearning Objectives and Evaluation

Machine unlearning was originally formalized as removing the influence of selected training data without complete retraining (Cao and Yang, 2015). SISA makes retraining-based deletion more efficient through sharding and slicing (Bourtoule et al., 2021); LLM-scale work instead commonly studies approximate post-hoc updates. Representative approaches include likelihood reversal, gradient difference, preference optimization, representation steering, relearning, and distillation-based removal (Eldan and Russinovich, 2023; Zhang et al., 2024; Fan et al., 2024; Li et al., 2024). Their evaluation increasingly separates target suppression from retain behavior, general utility, privacy exposure, and recovery robustness. TOFU, WMDP, MUSE, OpenUnlearning, and R-TOFU provide complementary factual, hazardous-knowledge, memorization, and robustness views (Maini et al., 2024; Li et al., 2024; Shi et al., 2024; Dorna et al., 2025; Yoon et al., 2025).

2.2 Localized Unlearning and Intervention Signals

Localized unlearning restricts updates to parameters, features, time steps, or vocabulary regions associated with a target. Saliency-based studies show that localization quality and downstream unlearning behavior should be assessed together (Lee et al., 2025). PALU localizes along temporal and vocabulary dimensions using prefix-aware entropy (Zhai et al., 2026). Other work studies targeted causal interventions, disentangled effects, and mechanistic localization for unlearning and editing (Liu et al., 2024a; Shen et al., 2024; Guo et al., 2025). Our route score combines three complementary views: gradient pressure identifies rows the forget objective seeks to change, the zero-intervention probe measures route-level effect on forget loss, and specificity discounts rows that also carry strong retain gradients.

2.3 Parameter-Efficient Routing and Adaptive Protection

LoRA and related parameter-efficient methods expose a compact trainable subspace while freezing the base model (Hu et al., 2022). This makes them attractive for repeated unlearning, but a compact adapter can still concentrate forget and retain be-

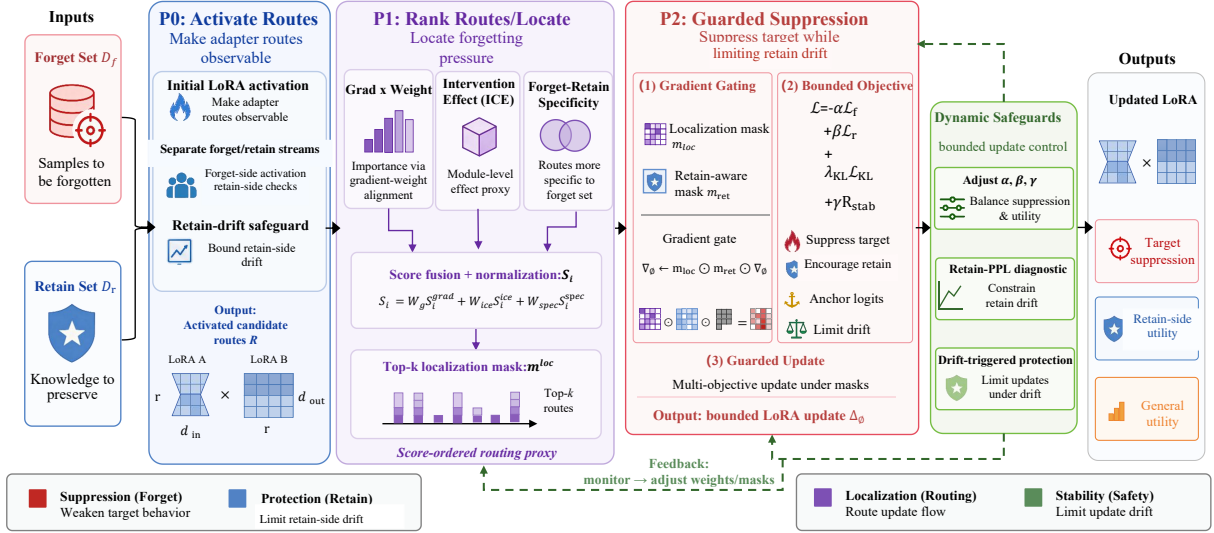


Figure 2: Protected update routing. P0 performs forget-side LoRA activation with separate retain-side safety checks. P1 ranks candidate routes using Grad $\times$ Weight, intervention-inspired sensitivity, and forget–retain specificity. P2 gates the update with localization and retain-aware masks; KL anchoring, stability regularization, and bounded dynamic safeguards limit collateral drift.

havior in overlapping directions. Protected routing addresses this by separating a hard localization mask from a retain-aware protection mask. KL anchoring and adapter-drift regularization complement the masks at the output and parameter levels, while bounded dynamic weights adapt the relative strengths of forgetting, retention, and stability as training progresses. The combination turns parameter efficiency into an explicit update-control interface rather than merely a reduction in trainable parameter count.

3 Method

The method follows a locate–protect–suppress–stabilize design (Figure 2). Route scoring determines *where* the update may flow, while guarded suppression determines *how strongly* selected routes are changed.

3.1 Problem Setup

Let $\mathcal{D}_f$ and $\mathcal{D}_r$ denote forget and retain data. A frozen autoregressive model p_{θ_0} is equipped with trainable LoRA parameters ϕ , yielding $p_{\theta_0, \phi}$ after unlearning. Only the adapter is updated. We write $\mathcal{L}_f$ and $\mathcal{L}_r$ for forget and retain cross-entropy losses.

For a LoRA module with factors A and B , a row/channel is a structured slice of the low-rank update. We use rows as the default routing unit because they balance within-tensor selectivity with a structured unit that can be scored and gated; Table 9

compares module, row, and element alternatives. The routing interface is

$$\begin{aligned} \text{Activate}(\phi) \\ &\rightarrow \text{RankRoutes}(\mathcal{D}_f, \mathcal{D}_r) \\ &\rightarrow \text{GuardedSuppress}(\mathcal{D}_f, \mathcal{D}_r). \end{aligned} \quad (1)$$

The intermediate route scores, masks, retain drift, and adapter drift make every stage of the update directly inspectable.

3.2 P0: Making Adapter Routes Observable

LoRA- B is initialized at zero, so immediately zeroing a module produces no change and makes the intervention probe degenerate. P0 therefore performs 15 gradient-ascent steps on shuffled forget batches at $0.8\times$ the main learning rate, updating LoRA parameters only. Separate retain batches are used for a PPL safety check every five steps; P0 stops if retain PPL exceeds $1.5\times$ its pre-activation value. There is no forget/retain matching or concatenation.

This observability warm-up supplies the measurable adapter state used by P1. In the recorded Qwen3-4B diagnostic, all 252 module-level ICE values are exactly zero before P0 and all exceed 2×10^{-6} afterward. The same zero-to-nonzero pattern is observed on the MUSE-News LLaMA-2-7B target (0/224 to 224/224); Table 1 reports the counts. The same activation pattern on Qwen3-4B and LLaMA-2-7B shows that P0 consistently resolves the zero-initialization degeneracy.

Model / setting	Active ICE before P0	Active ICE after P0
Qwen3-4B / TOFU	0 / 252	252 / 252
LLaMA-2-7B / MUSE-News	0 / 224	224 / 224
Qwen3-14B / TOFU (prelim.)	0 / 280	280 / 280

Table 1: P0/ICE observability diagnostic. LoRA-B initialization makes the zero-intervention probe degenerate before P0; after P0, every probed module has nonzero ICE in the three recorded settings. On Qwen3-4B, all post-P0 ICE values exceed 2×10^{-6} . The same activation pattern appears across all three recorded model settings, confirming that P0 converts an otherwise degenerate probe into a measurable routing signal.

3.3 P1: Intervention-Guided Route Scoring

P1 independently samples 16 forget and 8 retain batches (batch size 1, maximum length 128). It combines three min-max-normalized signals:

$$s_i = w_g s_i^{\text{grad}} + w_{\text{ice}} s_i^{\text{ice}} + w_{\text{spec}} s_i^{\text{spec}}. \quad (2)$$

The default weights are (0.30, 0.50, 0.20); Appendix L quantifies how alternative emphases move the resulting operating point.

Grad×Weight. For row block W_i , the first six forget batches give

$$s_i^{\text{grad}} = \text{Norm} \left(\frac{1}{6} \sum_{b=1}^6 \text{rowmean} \left(|\nabla_{W_i} \mathcal{L}_f(\mathcal{B}_b)| \odot |W_i| \right) \right). \quad (3)$$

where normalization is performed within each trainable tensor.

Intervention effect (ICE). For module m , the first four forget batches are evaluated both normally and with the entire LoRA-B tensor B_m temporarily zeroed and then restored:

$$s_m^{\text{ice}} = \left| \frac{1}{4} \sum_{b=1}^4 \mathcal{L}_f(\phi; \mathcal{B}_b) - \frac{1}{4} \sum_{b=1}^4 \mathcal{L}_f(\phi^{\setminus m}; \mathcal{B}_b) \right|. \quad (4)$$

The normalized module scalar is broadcast to its rows, providing a route-level effect prior that complements row-wise gradient signals.

The zero intervention also admits an exact Taylor decomposition. Let $g(t) = \mathcal{L}_f(tB_m)$ and assume twice differentiability along the segment from 0 to the post-P0 tensor B_m . Then for some $\tau \in (0, 1)$,

with $\xi = \tau B_m$,

$$\begin{aligned} \mathcal{L}_f(B_m) - \mathcal{L}_f(0) &= \langle \nabla_{B_m} \mathcal{L}_f(0), B_m \rangle \\ &\quad + \frac{1}{2} \text{vec}(B_m)^\top H_m(\xi) \text{vec}(B_m). \end{aligned} \quad (5)$$

If $\|H_m(\xi)\| \leq L_m$ on the segment, the remainder is bounded by $\frac{L_m}{2} \|B_m\|_F^2$. The implementation uses the measured loss change itself, while the decomposition connects that measurement to local gradient and curvature terms.

Forget-retain specificity. Using the first six forget and first six retain batches, let g_i^f and g_i^r be row-wise gradient magnitudes. We use

$$s_i^{\text{spec}} = \text{Norm} \left(\frac{g_i^f}{g_i^r + \epsilon} \right), \quad (6)$$

which down-ranks rows that are simultaneously important to forget and retain data. This contrast gives P1 a direct retain-side signal before P2 begins.

The fused score defines a fixed localization mask for P2:

$$m_i^{\text{loc}} = \mathbf{1}\{s_i \in \text{TopK}(s)\}, \quad \nabla_{\phi_i} \leftarrow m_i^{\text{loc}} \nabla_{\phi_i}. \quad (7)$$

Unless stated otherwise, top- k is applied within each trainable tensor over its row axis, producing a score-ordered routing mask.

3.4 P2: Retain-Aware Guarded Suppression

P2 estimates a smoothed retain-importance signal from retain gradients:

$$\begin{aligned} \rho_i^{(t)} &= \eta \rho_i^{(t-1)} + (1 - \eta) \text{Norm}(|g_i^r|), \\ \tilde{\rho}_i^{(t)} &= \rho_i^{(t)} / (\max_j \rho_j^{(t)} + \epsilon), \\ m_i^{\text{ret}} &= \mathbf{1}\{\tilde{\rho}_i^{(t)} < \tau_r\}. \end{aligned} \quad (8)$$

Gradients are gated as

$$\nabla_{\phi} \leftarrow m^{\text{loc}} \odot m^{\text{ret}} \odot \nabla_{\phi}. \quad (9)$$

The retain-aware mask limits exposure of locally retain-sensitive rows. Removing it reduces log forget-PPL from 7.518 to 4.705 at nearly unchanged General Avg, showing that the mask helps sustain suppression without broadening the update.

A local adapter-drift penalty is

$$\begin{aligned} R_{\text{stab}} &= \sum_{\ell} w_{\ell} \left[0.7 \|\phi_{\ell} - \phi_{\ell, \text{ref}}\|_2^2 \right. \\ &\quad \left. + 0.3 D_{\text{stat}}(\phi_{\ell}, \phi_{\ell, \text{ref}}) \right]. \end{aligned} \quad (10)$$

Algorithm 1 Protected update routing

Require: Forget set $\mathcal{D}_f$, retain set $\mathcal{D}_r$, LoRA parameters ϕ

- 1: P0: run forget-side activation; monitor separate retain batches.
- 2: P1: compute normalized Grad $\times$ Weight, ICE, and specificity scores.
- 3: Build fixed localization mask m^{loc} .
- 4: **for** each P2 step **do**
- 5: Update retain mask m^{ret} from retain-gradient statistics.
- 6: Compute bounded signals and controller weights.
- 7: Gate LoRA gradients with $m^{\text{loc}} \odot m^{\text{ret}}$.
- 8: Update ϕ using Eq. 11; apply scheduled KL and PPL checks.
- 9: **end for**

where D_{stat} penalizes tensor mean and standard-deviation-ratio shifts. The masked parameters optimize

$$\mathcal{L} = -\alpha_t \mathcal{L}_f + \beta_t \mathcal{L}_r + \lambda_{\text{KL}} \mathcal{L}_{\text{KL}} + \gamma_t R_{\text{stab}}. \quad (11)$$

The retain CE and KL terms preserve retain-side behavior at the likelihood and logit levels, while the stability term limits adapter drift.

3.5 Bounded Dynamic Safeguards

The controller reads forget loss, retain loss, LoRA drift d , and progress:

$$\begin{aligned} q_f &= \text{clip}((\mathcal{L}_f - 2.5)/1.5, 0, 1), \\ q_r &= \text{clip}(1 - \mathcal{L}_r/5.0, 0, 1), \\ q_s &= \text{clip}(1 - d, 0, 1). \end{aligned} \quad (12)$$

A two-head MLP (hidden size 64) maps $(q_f, q_r, q_s, t/T)$ to a three-way mixture π_t and a value estimate. The mixture is affinely bounded:

$$\begin{aligned} \alpha_t &= 1.5 + 2.5\pi_{t,1}, \\ \beta_t &= 1.0 + 1.5\pi_{t,2}, \\ \gamma_t &= 0.05 + 0.5\pi_{t,3}. \end{aligned} \quad (13)$$

Every ten steps, a reward-weighted update uses normalized discounted returns (discount 0.99) from ΔJ_t , where $J_t = 0.4q_f + 0.4q_r + 0.2q_s$, a learned value baseline, Dirichlet exploration, Adam at 10^{-4} , and gradient clipping. KL anchoring is evaluated every ten steps; P2 checks retain PPL every 100 steps. The controller acts as an adaptive safeguard whose outputs remain inside fixed, interpretable bounds. Appendix O gives pseudocode, and Appendix M reports loss-scale sensitivity.

4 Experiments

4.1 Main Protocol

The main study uses Qwen3-4B with LoRA on TOFU-forget10 (Maini et al., 2024) and

WMDP Bio, Chem, and Cyber (Li et al., 2024). TOFU evaluates removal of fictitious biographical facts; WMDP evaluates hazardous multiple-choice knowledge. WikiText-2 supplies retain text (Merity et al., 2017). General Avg is the mean of MMLU, ARC-Challenge, and BoolQ (Hendrycks et al., 2021; Clark et al., 2018, 2019). The main NPO, RMU, and protected-routing results are replicated with seeds 11, 22, and 33. Original denotes the fixed pre-unlearning model. GA and GA-retain provide aggressive-update references and are reported in the canonical-seed tables.

All shared-framework methods use the same base checkpoint, tokenizer, LoRA rank and target modules, optimizer family, sequence length, training budget, retain data, and evaluator. GA maximizes forget loss; GA-retain adds retain cross-entropy; NPO uses cached reference log-probabilities (Zhang et al., 2024); RMU applies representation misdirection with retain activation matching (Li et al., 2024). Method-specific controls and suppression-strength sweeps are listed in Appendix N and Appendix P.

4.2 Metrics

TOFU reports $\log_{10}$ forget perplexity ratio, ROUGE-L generation overlap (R-L) (Lin, 2004), retain perplexity ratio (R-PPL), and General Avg. Higher forget-PPL and lower R-L indicate stronger factual suppression. WMDP reports unlearning effect (UE=1−accuracy), R-PPL, and General Avg. R-PPL is interpreted by proximity to one and complements the downstream-task view given by General Avg.

For direct comparison with localized unlearning, we additionally follow PALU’s LLaMA-2-7B TOFU-forget05 Forget Quality–Model Utility protocol (Zhai et al., 2026). This setting uses the same target checkpoint and evaluation definitions for all reported methods. We also evaluate MUSE-News on the official pre-memorized LLaMA-2-7B checkpoint (Shi et al., 2024), use Granite-4.1-8B as a second shared-framework model, and run a Qwen3-14B scale diagnostic.

4.3 Implementation and Reproducibility

P0 uses 15 forget-side ascent steps at $0.8\times$ the main learning rate. P1 samples 16 forget and 8 retain batches independently; Grad $\times$ Weight uses six forget batches, ICE uses four, and specificity uses six forget plus six retain batches. P2 performs masked multi-objective training with the controller

Method	$\log_{10}$ F-PPL ↑	R-L ↓	R-PPL ≈ 1	Gen. ↑
Original	0.000	0.349	1.000	0.818
NPO	4.261 ± 0.430	0.137 ± 0.010	2.294 ± 1.037	0.734 ± 0.047
RMU	-0.228 ± 0.019	0.264 ± 0.054	0.571 ± 0.020	0.737 ± 0.046
Ours	7.935 ± 1.655	0.107 ± 0.033	1.519 ± 0.085	0.617 ± 0.020

Table 2: Main-setting TOFU-forget10 results. NPO, RMU, and Ours are mean $\pm$ standard deviation over seeds 11, 22, and 33; Original is the fixed pre-unlearning reference. Protected update routing achieves the strongest target suppression among the replicated methods. NPO/RMU retain higher General Avg; R-PPL is interpreted by proximity to one.

and safety checks in Algorithm 1. Full hyperparameters, baseline parity, dynamic-weighting pseudocode, evaluation scripts, camera-ready configurations, and table-generation scripts are included in the supplementary material.

5 Results and Analysis

5.1 Three-Seed Main Results

TOFU-forget10. Table 2 shows that protected update routing consistently moves the model toward stronger factual suppression. Its mean $\log_{10}$ forget-PPL ratio is 7.935, compared with 4.261 for NPO and -0.228 for RMU, while R-L falls to 0.107. General Avg remains 0.617, substantially above the aggressive GA reference and below the more conservative NPO/RMU points. The result demonstrates the effect of explicit route selection: the update reaches a stronger suppression region without reproducing the global collapse of unrestricted ascent.

WMDP. WMDP presents a complementary regime. Table 3 shows that NPO and RMU attain higher UE, while protected routing preserves the highest General Avg among the replicated unlearning methods: 0.459 versus 0.378 for RMU and 0.352 for NPO. The method therefore shifts the hazardous-knowledge trade-off toward retained downstream capability. The canonical-seed subset breakdown in Table 4 confirms the same pattern on Bio, Chem, and Cyber separately, where protected routing records the highest General Avg among unlearning methods in every subset.

The downstream advantage is distributed across tasks rather than driven by a single benchmark component. In the canonical WMDP runs, protected routing improves averaged MMLU and ARC-Challenge relative to NPO/RMU while retaining competitive BoolQ, yielding General Avg

Method	UE Avg ↑	R-PPL Avg ≈ 1	Gen. Avg ↑
Original	0.443	1.000	0.818
NPO	0.766 ± 0.007	0.446 ± 0.004	0.352 ± 0.023
RMU	0.760 ± 0.035	0.600 ± 0.004	0.378 ± 0.068
Ours	0.570 ± 0.038	1.705 ± 0.167	0.459 ± 0.014

Table 3: Main-setting WMDP results averaged over Bio, Chem, and Cyber. NPO, RMU, and Ours are mean $\pm$ standard deviation over seeds 11, 22, and 33; Original is the fixed reference. Protected update routing achieves the highest General Avg among the replicated unlearning methods, while NPO/RMU attain higher UE.

0.517 (Table 5). This decomposition supports the interpretation that localized update flow protects broad capability, not merely the aggregate score.

5.2 Comparison with Direct Localized Unlearning

Table 6 evaluates protected routing in PALU’s protocol-matched LLaMA-2-7B TOFU-forget05 setting. PALU reaches the highest Forget Quality at 0.7126, while our method reaches 0.6947 and records the highest Model Utility among the evaluated unlearning methods at 0.6311. Relative to NPO, protected routing improves Forget Quality by 0.0663 and Model Utility by 0.0391. It also improves both metrics over TPO and substantially improves Forget Quality over SimNPO. The comparison shows that route-based localization is competitive with prefix-aware localization while exposing a complementary set of inspectable objects: row scores, localization masks, retain masks, and adapter-drift trajectories.

Method	Forget Quality ↑	Model Utility ↑
Original	5.87×10^{-14}	0.6276
Retain reference	1.0000	0.6266
GA	5.95×10^{-11}	0.5580
GD	0.0396	0.3577
DPO	0.5453	0.5503
NPO	0.6284	0.5920
SimNPO	0.4663	0.5921
PDU	0.0021	0.5111
TPO	0.6284	0.5862
Ours	0.6947	0.6311
PALU	0.7126	0.6238

Table 6: Protocol-matched LLaMA-2-7B TOFU-forget05 comparison under the Forget Quality–Model Utility evaluation used by PALU. Among unlearning methods, PALU attains the highest Forget Quality, while protected update routing attains the highest Model Utility and improves both metrics over NPO/TPO.

Subset	Method	Forget		Retain	General
		WMDP Acc. ↓	UE ↑	R-PPL ≈ 1	Gen. ↑
Bio	Original	0.670	0.330	1.000	0.818
Bio	GA	0.200	0.800	1.60×10^7	0.374
Bio	GA-retain	0.320	0.680	0.667	0.293
Bio	NPO	0.320	0.680	0.448	0.333
Bio	RMU	0.280	0.720	0.600	0.455
Bio	Ours	0.425	0.575	1.895	0.554
Chem	Original	0.570	0.430	1.000	0.818
Chem	GA	0.240	0.760	1.60×10^7	0.293
Chem	GA-retain	0.240	0.760	0.670	0.293
Chem	NPO	0.240	0.760	0.448	0.354
Chem	RMU	0.250	0.750	0.607	0.455
Chem	Ours	0.375	0.625	1.564	0.487
Cyber	Original	0.430	0.570	1.000	0.818
Cyber	GA	0.240	0.760	1.60×10^7	0.374
Cyber	GA-retain	0.310	0.690	0.620	0.293
Cyber	NPO	0.310	0.690	0.445	0.364
Cyber	RMU	0.260	0.740	0.599	0.434
Cyber	Ours	0.338	0.663	2.006	0.508

Table 4: WMDP per-subset results for the canonical main operating point. Protected update routing records the highest General Avg among unlearning methods on Bio, Chem, and Cyber, while GA/RMU provide the largest UE. Table 3 reports the three-seed aggregate.

Method	MMLU	ARC-C	BoolQ	Gen. Avg
NPO	0.121	0.242	0.687	0.350
RMU	0.232	0.333	0.778	0.448
Ours	0.396	0.492	0.663	0.517

Table 5: Canonical-seed WMDP utility components averaged over Bio, Chem, and Cyber. The utility gain of protected routing is distributed across MMLU and ARC-Challenge rather than arising from a single aggregate score.

5.3 Stage Contributions and Routing Granularity

P0 is necessary because LoRA-B initialization makes the zero-intervention score identically zero before activation. Table 1 shows the same transition in three recorded settings: all probes are inactive before P0 and active afterward. On Qwen3-4B, every post-P0 ICE value exceeds 2×10^{-6} . This observation aligns with the stage-level ablation in Table 7, where removing P0 reduces $\log_{10}$ forget-PPL from 27.107 to 0.032.

The stage-level and component-level ablations answer complementary questions. Table 7 removes major stages under a fixed-step setting, while Table 8 isolates the retain mask, KL anchor, stability term, controller, localization mask, and P0 under the main operating point. Removing localization produces very strong suppression but lowers General Avg to 0.380, demonstrating why route breadth must be controlled. Removing P0 nearly eliminates forgetting. The retain-aware mask increases suppression from 4.705 to 7.518 at similar General

Variant	$\log_{10}$ ↑	F-PPL ↓	R-L ↓	R-PPL ≈ 1	Gen. ↑
Full	27.107	0.000	1.145	0.437	
w/o P0 activation	0.032	0.239	1.024	0.582	
Localization only	1.766	0.254	3.074	0.524	
w/o localization	24.578	0.027	1.333	0.238	
w/o dynamic weighting	23.552	0.000	1.126	0.287	

Table 7: System-level TOFU-forget10 ablation under the stronger fixed-step protocol. P0 activates measurable routes, localization controls update breadth, and dynamic weighting shifts the suppression–utility operating point.

Avg, showing how retain-side evidence can support a stronger but still localized update.

Table 9 compares module, row, and element routing at nominal $k = 0.30$ over three seeds. All three granularities achieve substantial suppression. Row routing attains the highest General Avg, 0.628, and the R-PPL closest to one, 1.365, while maintaining strong forget-PPL growth. Module routing lacks within-tensor selectivity; element routing changes the trade-off at a finer scale. The General-Avg ordering row > element > module holds for all three seeds at this routing density, supporting rows as a practical structured unit for the proposed interface.

5.4 Cross-Benchmark and Cross-Scale Evidence

MUSE-News evaluates a memorized target checkpoint from a second model family. Its initial forget loss is approximately 2.3, so the bounded forget-

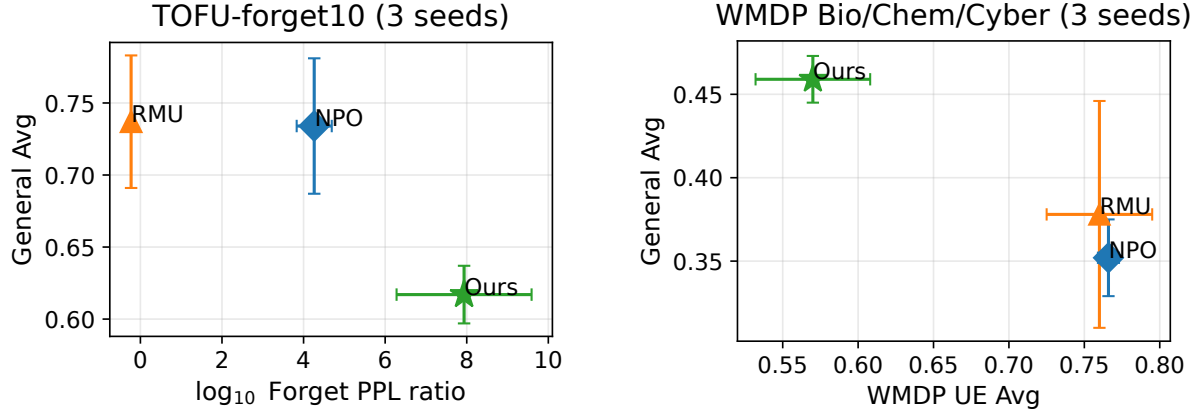


Figure 3: Three-seed forgetting-utility operating points. On TOFU, protected routing moves toward stronger target suppression. On WMDP, it moves toward higher downstream utility. Error bars show standard deviation over seeds 11, 22, and 33.

Variant	$\log_{10}$ F-PPL ↑	R-L ↓	R-PPL ≈ 1	Gen. ↑
Full main	7.518	0.181	1.443	0.512
w/o retain-aware mask	4.705	0.214	1.331	0.520
w/o KL anchoring	4.570	0.197	1.440	0.515
w/o stability regularizer	4.626	0.211	1.415	0.519
w/o dynamic weighting + safety	4.952	0.207	1.361	0.535
w/o localization mask	17.843	0.036	1.176	0.380
w/o P0 activation	0.001	0.247	1.003	0.585

Table 8: Main-setting component ablation on TOFU-forget10. P0 creates the measurable adapter state, localization controls update breadth, and the protection/stability components shape suppression strength at comparable downstream utility.

Granularity	$\log_{10}$ F-PPL ↑	R-L ↓	R-PPL ≈ 1	Gen. ↑
Module	21.87 ± 1.99	0.012 ± 0.011	1.443 ± 0.047	0.486 ± 0.050
Row	18.75 ± 1.63	0.066 ± 0.060	1.365 ± 0.056	0.628 ± 0.028
Element	19.43 ± 0.76	0.050 ± 0.050	1.500 ± 0.119	0.574 ± 0.021

Table 9: Granularity diagnostic on TOFU-forget10 at nominal $k = 0.30$ (three seeds). All granularities suppress substantially. Row-level routing trades some suppression strength for the highest General Avg and lowest retain drift in this setting; and offers the strongest utility-oriented operating point among the tested granularities.

control map is calibrated once from that initial scale, changing the center/scale from (2.5, 1.5) to (0.5, 2.0) before evaluation. Table 10 shows the resulting operating points. GA produces the strongest suppression with extreme retain drift; NPO preserves retain behavior but changes forget perplexity only slightly; protected routing reaches moderate suppression with R-PPL 1.336 and General Avg 0.383. VerbMem also decreases from 0.455 to 0.420 for our method.

The Qwen3-14B scale analysis measures P0/ICE behavior at $3.5\times$ the main model size. Table 11 records a 0/280 to 280/280 activation transition,

Method	$\log_{10}$ F-PPL ↑	R-PPL ≈ 1	Gen. ↑
Original	0.000	1.000	0.434
GA	8.114	1.1×10^8	0.333
NPO	0.071	1.126	0.414
RMU	0.538	3.238	0.323
Ours	0.211	1.336	0.383

Table 10: MUSE-News evaluation on the official pre-memorized LLaMA-2-7B checkpoint (seed 11). Controller calibration uses the initial forget-loss scale before evaluation. The result extends protected routing to a memorized LLaMA-2-7B target.

$\log_{10}$ forget-PPL 21.96, R-PPL 1.452, and General Avg 0.704 after unlearning. Granite-4.1-8B provides an additional shared-framework check: on its WMDP average, protected routing reaches UE 0.716 and General Avg 0.374, compared with 0.703/0.320 for NPO and 0.703/0.347 for RMU. Together, these results show that the routing mechanism remains effective beyond the main Qwen3-4B setting.

Metric	Before	After
Active ICE modules	0 / 280	280 / 280
ROUGE-L	0.225	0.143
General Avg	0.800	0.704
$\log_{10}$ F-PPL after unlearning		21.96
R-PPL after unlearning		1.452

Table 11: Qwen3-14B TOFU scale analysis (seed 11). The larger model reproduces the P0/ICE activation pattern, achieves strong target suppression, and retains a General Avg of 0.704.

5.5 Sensitivity and Cost

Across score-weight variants, General Avg stays within 0.583–0.617 under strong suppression; 12 safeguard perturbations retain $\log_{10}$ forget-PPL 24.26–27.96 and General Avg 0.554–0.633. P0 takes 5.05 s, localization 96.42 s once per activated state, and P2 1.61 s/step; full timings appear in Appendix F.

5.6 What Protected Routing Adds

The results expose a consistent functional decomposition. P0 converts zero-initialized adapter paths into measurable routes; P1 maps attribution, intervention effect, and forget–retain specificity to update eligibility; P2 regulates the pressure applied to those routes. Removing P0 nearly eliminates suppression, while removing localization admits a broad, high-damage update. Retain-aware masking strengthens suppression at similar downstream utility, and bounded weights maintain a stable operating range. Protected routing therefore adds an explicit update-control surface to the unlearning objective.

This control surface transfers across settings. Row-routed updates strengthen TOFU suppression, preserve higher WMDP downstream utility, and attain the highest Model Utility under PALU’s protocol while remaining close in Forget Quality. Granularity experiments show that the route unit itself changes the attainable balance. Explicit scores, masks, controller signals, and drift trajectories make each operating point directly inspectable.

6 Conclusion

Protected update routing turns LoRA unlearning into explicit update-flow control, combining strong target suppression, retained utility, and inspectable route-level evidence across multi-seed, PALU, MUSE-News, and larger-model evaluations.

Limitations

The main controlled study focuses on LoRA-based unlearning for decoder-only models, with Qwen3-4B as the principal backbone. Granite-4.1-8B, LLaMA-2-7B, and Qwen3-14B broaden the evidence, but their experiments use different checkpoints, benchmark protocols, or seed counts. A complete scaling study would repeat every main baseline and ablation across several architectures and model sizes, including full-parameter unlearning and encoder–decoder models.

Protected routing is most effective when the target can be concentrated in a selective adapter update. TOFU fits this regime better than WMDP: the method produces strong factual suppression on TOFU, whereas NPO and RMU achieve higher WMDP UE. Broader hazardous concepts may require representation-level or multi-layer changes that extend beyond a row-routed LoRA update. Row granularity also remains a design choice conditioned on the evaluated routing densities; module, row, and element masks can favor different points as density and effective parameter budget change.

ICE measures the loss change caused by zeroing a LoRA-B module after P0. It is a useful route-level effect signal, but it aggregates all rows in a module and depends on the activated adapter state and probe batches. Repeated interventions, confidence intervals for route scores, direct route-overlap measurements, and stability analysis across seeds would strengthen the localization evidence. The forget–retain specificity ratio likewise reduces shared-gradient routes without enforcing orthogonality, leaving room for residual retain-gradient mass.

The controller uses fixed bounded mappings from loss and drift signals. Sensitivity experiments show stable operation across the tested perturbations, while MUSE-News requires a one-time calibration of the forget-loss center and scale. A scale-relative or automatically normalized controller would improve transfer across checkpoints and domains. KL anchoring is evaluated periodically, so a dedicated frequency study could determine whether more frequent anchors reduce transient retain drift.

The method adds runtime through P0 activation, P1 localization, retain-mask updates, and controller evaluation. Localization can be reused for P2 sweeps sharing the same activated state, but sequential deletion requests would still require re-

freshed scores or an incremental routing mechanism. The current evaluation studies behavioral suppression and utility. It does not verify exact data removal, resistance to membership inference or re-learning attacks, or compliance with legal erasure requirements. Such settings require complementary privacy, recovery, and provenance tests beyond the behavioral metrics used here.

Acknowledgments

This work was supported in part by the collaborative research project “Research and Development of Key Technologies for Industrial Electricity Consumption Analysis and Smart Marketing Applications” between Shenzhen Power Supply Co., Ltd. and Xidian University, and by the Seed Funding for Collaborative Research Grants of Hong Kong Baptist University (Grant No. RC-SFCRG/23-24/R2/SCI/06). Generative AI tools, including ChatGPT and Codex, were used to assist with code generation and debugging. All outputs were reviewed and verified by the authors, who take full responsibility for the work.

References

- Lucas Bourtole, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. 2021. [Machine unlearning](#). In *2021 IEEE Symposium on Security and Privacy*, pages 141–159.
- Yinzhi Cao and Junfeng Yang. 2015. [Towards making systems forget with machine unlearning](#). In *2015 IEEE Symposium on Security and Privacy*, pages 463–480.
- Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. [BoolQ: Exploring the surprising difficulty of natural yes/no questions](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2924–2936. Association for Computational Linguistics.
- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.
- Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zachary C. Lipton, J. Zico Kolter, and Pratyush Maini. 2025. [OpenUnlearning: Accelerating LLM unlearning via unified benchmarking of methods and metrics](#). *Preprint*, arXiv:2506.12618.
- Ronen Eldan and Mark Russinovich. 2023. [Who’s harry potter? approximate unlearning in LLMs](#). *Preprint*, arXiv:2310.02238.
- Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. 2024. [Simplicity prevails: Rethinking negative preference optimization for LLM unlearning](#). *Preprint*, arXiv:2410.07163.
- Jiahui Geng, Qing Li, Herbert Woitschlaeger, Zongxiong Chen, Fengyu Cai, Yuxia Wang, Preslav Nakov, Hans-Arno Jacobsen, and Fakhri Karray. 2025. [A comprehensive survey of machine unlearning techniques for large language models](#). *arXiv preprint arXiv:2503.01854*.
- Phillip Huang Guo, Aaquib Syed, Abhay Sheshadri, Aidan Ewart, and Gintare Karolina Dziugaite. 2025. Mechanistic unlearning: Robust knowledge unlearning and editing via mechanistic localization. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 20964–20992.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. [Measuring massive multitask language understanding](#). In *International Conference on Learning Representations*.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. [LoRA: Low-rank adaptation of large language models](#). In *International Conference on Learning Representations*.
- Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2023. [Knowledge unlearning for mitigating privacy risks in language models](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14389–14408, Toronto, Canada. Association for Computational Linguistics.
- Hwiyeong Lee, Uiji Hwang, Hyelim Lim, and Taeuk Kim. 2025. [Does localization inform unlearning? a rigorous examination of local parameter attribution for knowledge unlearning in language models](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 21857–21869, Suzhou, China. Association for Computational Linguistics.
- Nathaniel Li, Alexander Pan, Anjali Gopal, Sumner Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew B. Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, and 38 others. 2024. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*.

Chin-Yew Lin. 2004. [ROUGE: A package for automatic evaluation of summaries](#). In *Text Summarization Branches Out*, pages 74–81. Association for Computational Linguistics.

Yujian Liu, Yang Zhang, Tommi Jaakkola, and Shiyu Chang. 2024a. [Revisiting who’s harry potter: Towards targeted unlearning from a causal intervention perspective](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8708–8731, Miami, Florida, USA. Association for Computational Linguistics.

Zheyuan Liu, Guangyao Dou, Zhaoxuan Tan, Yijun Tian, and Meng Jiang. 2024b. [Machine unlearning in generative AI: A survey](#). *arXiv preprint arXiv:2407.20516*.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. 2024. [TOFU: A task of fictitious unlearning for LLMs](#). *Preprint*, arXiv:2401.06121.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. [Pointer sentinel mixture models](#). In *International Conference on Learning Representations*.

Vaidehi Patil, Peter Hase, and Mohit Bansal. 2023. [Can sensitive information be deleted from LLMs? objectives for defending against extraction attacks](#). *arXiv preprint arXiv:2309.17410*.

Shaofei Shen, Chenhao Zhang, Alina Bialkowski, Weitong Chen, and Miao Xu. 2024. [CaMU: Disentangling causal effects in deep model unlearning](#). In *Proceedings of the 2024 SIAM International Conference on Data Mining*, pages 779–787.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. 2024. [MUSE: Machine unlearning six-way evaluation for language models](#). *Preprint*, arXiv:2407.06460.

Qizhou Wang, Bo Han, Puning Yang, Jianing Zhu, Tongliang Liu, and Masashi Sugiyama. 2024. [Unlearning with control: Assessing real-world utility for large language model unlearning](#). *arXiv preprint arXiv:2406.09179*.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. 2024. [Large language model unlearning](#). In *Advances in Neural Information Processing Systems*.

Sangyeon Yoon, Wonje Jeung, and Albert No. 2025. [R-TOFU: Unlearning in large reasoning models](#). *Preprint*, arXiv:2505.15214.

Naixin Zhai, Pengyang Shao, Binbin Zheng, Fei Shen, Long Bai, and Xun Yang. 2026. [Maximizing local entropy where it matters: Prefix-aware localized LLM unlearning](#). *Preprint*, arXiv:2601.03190.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. 2024. [Negative preference optimization: From catastrophic collapse to effective unlearning](#). *Preprint*, arXiv:2404.05868.

A Canonical-Seed Aggressive-Update References

Table 12 retains the canonical-seed GA and GA-retain results that illustrate the over-suppression regime alongside the main baselines.

Method	$\log_{10}$ F-PPL ↑	R-L ↓	R-PPL ≈ 1	Gen. ↑
Original	0.000	0.349	1.000	0.818
GA	7.372	0.000	2.72×10^7	0.303
GA-retain	7.372	0.082	202.383	0.515
NPO	4.124	0.246	2.577	0.788
RMU	-0.214	0.321	0.608	0.737
Ours	4.871	0.205	1.496	0.633

Table 12: Canonical-seed TOFU-forget10 shared-framework comparison, illustrating GA-style over-suppression alongside the main baselines.

B Seed Stability

Table 13 reports a five-seed TOFU-forget10 sweep that confirms the same strong-suppression operating region across a wider set of random seeds.

Method	$\log_{10}$ F-PPL ↑	R-L ↓	R-PPL ≈ 1	Gen. ↑
Ours	25.650 ± 1.821	0.019 ± 0.020	1.315 ± 0.158	0.518 ± 0.089
NPO	4.792 ± 0.740	0.128 ± 0.014	2.557 ± 0.746	0.749 ± 0.038
RMU	-0.224 ± 0.013	0.252 ± 0.071	0.580 ± 0.018	0.760 ± 0.046

Table 13: Five-seed TOFU-forget10 stability analysis. Values are mean $\pm$ standard deviation over seeds 11, 22, 33, 44, and 55. Protected routing maintains substantially stronger target suppression than NPO/RMU across the wider seed set.

C Granularity Construction Details

The granularity study holds the training configuration and nominal top- k fixed. Module scores are per-tensor means of each normalized channel. Element scores use per-parameter $\text{Grad} \times \text{Weight}$ and specificity, with module ICE broadcast to elements. Effective update-eligible fractions are 0.25–0.30 at $k = 0.30$ and 0.78–0.81 at $k = 0.80$.

D MUSE-News Protocol Details

The MUSE-News run uses the official pre-memorized LLaMA-2-7B checkpoint, common splits, LoRA setup, learning rate, and training budget. The protected-routing controller is calibrated

once before evaluation from the initial forget loss: the map changes from $\text{clip}((\mathcal{L}_f - 2.5)/1.5)$ to $\text{clip}((\mathcal{L}_f - 0.5)/2.0)$. The learning rate is 10^{-4} . The main paper reports the full comparison in Table 10; VerbMem changes from 0.455 to 0.420 for protected routing.

E Top-k Routing Sensitivity

Table 14 and Figure 4 show how routing density controls the forgetting–utility point. At top-k=1.0 the localization gate disappears and General Avg falls under strong forgetting, highlighting the value of selective route admission.

k fraction	Selected rows	Eff. grad	$\log_{10}$ F-PPL ↑	R-L ↓	R-PPL ≈ 1	Gen. ↑
0.30	0.300	–	21.329	0.020	1.204	0.330
0.50	0.500	–	26.362	0.063	1.257	0.264
0.70	0.700	–	25.918	0.010	1.116	0.337
0.80	0.800	–	27.107	0.000	1.145	0.337
1.00	1.000	–	28.881	0.020	1.240	0.233

Table 14: Top-k routing sensitivity. Smaller top-k values impose stricter routing, while top-k=1.0 removes the localization gate. The experiment visualizes how localization density changes the forgetting–utility operating point.

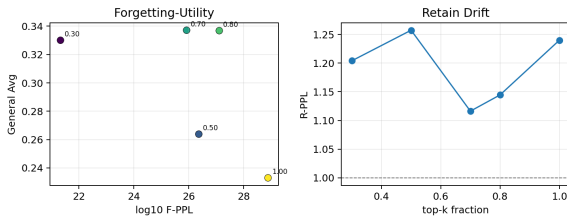


Figure 4: Top-k routing sensitivity on TOFU-forget10. Smaller fractions impose stricter route selection; top-k=1.0 admits every row.

F Runtime Overhead

Table 15 decomposes activation, localization, and recurring P2 cost under a fixed-step TOFU-forget10 profile.

Method	Total s	Activation s	Loc. s	Train s	s/ step
GA	109.19	0.00	0.00	12.44	0.25
NPO	140.36	0.00	0.00	24.39	0.49
RMU	126.11	0.00	0.00	28.46	0.57
Ours	570.23	5.05	96.42	80.75	1.61

Table 15: Runtime profiling on TOFU-forget10. The same model and sequence length are used for all methods. Activation and localization are one-time costs; the per-step column isolates the recurring routing overhead.

G General Utility Breakdown

Table 16 decomposes General Avg into MMLU, ARC-Challenge, and BoolQ, showing how protected routing distributes its utility advantage across downstream tasks.

Subset	Method	MMLU	ARC-C	BoolQ	General Avg
<i>TOFU-forget10</i>					
TOFU	Original	0.697	0.939	0.818	0.818
TOFU	GA	0.121	0.273	0.515	0.303
TOFU	GA-retain	0.333	0.485	0.727	0.515
TOFU	NPO	0.636	0.909	0.818	0.788
TOFU	RMU	0.576	0.848	0.788	0.737
TOFU	Ours	0.525	0.725	0.650	0.633
<i>WMDP</i>					
Bio	Original	0.697	0.939	0.818	0.818
Bio	GA	0.303	0.303	0.515	0.374
Bio	GA-retain	0.121	0.242	0.515	0.293
Bio	NPO	0.121	0.242	0.636	0.333
Bio	RMU	0.242	0.333	0.788	0.455
Bio	Ours	0.438	0.550	0.675	0.554
Chem	Original	0.697	0.939	0.818	0.818
Chem	GA	0.121	0.242	0.515	0.293
Chem	GA-retain	0.121	0.242	0.515	0.293
Chem	NPO	0.121	0.242	0.697	0.354
Chem	RMU	0.242	0.333	0.788	0.455
Chem	Ours	0.388	0.475	0.600	0.487
Cyber	Original	0.697	0.939	0.818	0.818
Cyber	GA	0.303	0.303	0.515	0.374
Cyber	GA-retain	0.121	0.242	0.515	0.293
Cyber	NPO	0.121	0.242	0.727	0.364
Cyber	RMU	0.212	0.333	0.758	0.434
Cyber	Ours	0.362	0.450	0.713	0.508

Table 16: General utility breakdown. General Avg is the mean of MMLU, ARC-Challenge, and BoolQ; the table reports the components to show which downstream abilities change under each unlearning method.

H Preference-Sensitivity Visualization

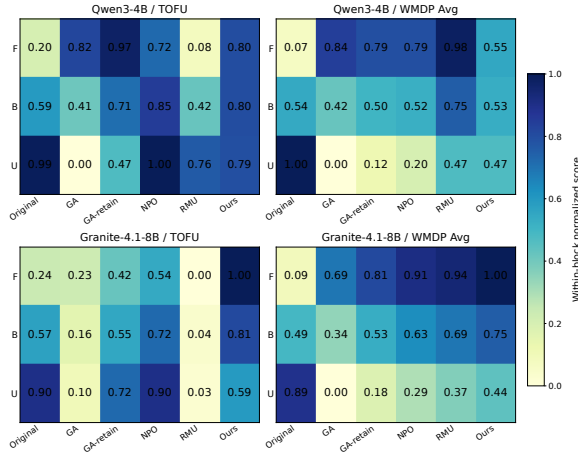


Figure 5: Preference-sensitivity visualization generated from raw metrics. F, B, and U denote forgetting-, balanced-, and utility-oriented preferences within each model–benchmark block.

I Compact Granite Summary

Task	Method	Target $\uparrow$	R-PPL ≈ 1	Gen. $\uparrow$
TOFU	NPO	7.311	1.547	0.646
	RMU	0.463	2.680	0.303
	Ours	19.434	1.341	0.417
WMDP	NPO	0.703	0.842	0.320
	RMU	0.703	1.042	0.347
	Ours	0.716	1.199	0.374

Table 17: Granite-4.1-8B single-seed diagnostic summary. Target is $\log_{10}$ F-PPL for TOFU and UE Avg for WMDP.

J Granite Diagnostic Raw Metrics

Tables 18 and 19 report the full Granite-4.1-8B shared-framework metrics. Protected routing reaches strong TOFU suppression and the highest WMDP-average General Avg among the compared methods; R-PPL and General Avg provide complementary retain and downstream views.

Method	$\log_{10}$ F-PPL $\uparrow$	R-L $\downarrow$	R-PPL ≈ 1	Gen. $\uparrow$
Original	0.000	0.224	1.000	0.667
GA	7.311	0.000	2.563×10^7	0.374
GA-retain	7.311	0.000	66.276	0.626
NPO	7.311	0.000	1.547	0.646
RMU	0.463	0.120	2.680	0.303
Ours	19.434	0.069	1.341	0.417

Table 18: Granite-4.1-8B TOFU-forget10 shared-framework metrics (seed 11). F-PPL denotes forget perplexity ratio, R-L denotes target-answer overlap, and R-PPL denotes retain perplexity ratio.

Dataset	Method	Acc. $\downarrow$	UE $\uparrow$	R-PPL ≈ 1	Gen. $\uparrow$
Bio	Original	0.730	0.270	1.000	0.667
Bio	GA	0.320	0.680	3.192×10^7	0.293
Bio	GA-retain	0.320	0.680	1.633	0.293
Bio	NPO	0.320	0.680	0.751	0.283
Bio	RMU	0.320	0.680	1.010	0.333
Bio	Ours	0.175	0.825	1.170	0.350
Chem	Original	0.480	0.520	1.000	0.667
Chem	GA	0.260	0.740	3.192×10^7	0.384
Chem	GA-retain	0.310	0.690	1.213	0.283
Chem	NPO	0.260	0.740	0.748	0.364
Chem	RMU	0.250	0.750	1.072	0.323
Chem	Ours	0.338	0.662	1.251	0.321
Cyber	Original	0.410	0.590	1.000	0.667
Cyber	GA	0.310	0.690	3.192×10^7	0.293
Cyber	GA-retain	0.310	0.690	1.102	0.293
Cyber	NPO	0.310	0.690	1.028	0.313
Cyber	RMU	0.320	0.680	1.045	0.384
Cyber	Ours	0.338	0.662	1.175	0.450

Table 19: Granite-4.1-8B WMDP Bio/Chem/Cyber shared-framework metrics (seed 11). Acc. denotes WMDP accuracy and UE denotes unlearning effect.

K Localizer Diagnostic

Table 20 varies the row-selection score while holding P2 fixed. The fused score gives the highest General Avg among the tested selectors while retaining R-L 0.000 and strong forget-PPL growth, showing the benefit of combining attribution, intervention effect, and specificity.

Variant	$\log_{10}$ F-PPL $\uparrow$	R-L $\downarrow$	R-PPL ≈ 1	Gen. $\uparrow$
full_score	25.792	0.000	1.144	0.474
random_rows	27.033	0.000	1.123	0.371
grad_only	27.483	0.000	1.160	0.408
specificity_only	23.712	0.018	1.238	0.338
ice_only	20.156	0.076	1.143	0.300
magnitude_only	22.468	0.047	1.120	0.258

Table 20: Localizer analysis on TOFU-forget10. All variants share the objective, LoRA budget, top- k fraction, and evaluation pipeline while changing the row-selection score. The combined score attains the highest General Avg, drives R-L to 0.000, and keeps R-PPL close to one; log forget-PPL separates configurations that share zero target overlap.

L Fusion-Weight Sensitivity

(w_g, w_{ice}, w_{spec})	$\log_{10}$ F-PPL $\uparrow$	R-PPL ≈ 1	Gen. $\uparrow$
0.30/0.50/0.20 (default)	25.07	1.397	0.600
0.60/0.30/0.10 (grad-heavy)	25.81	1.379	0.583
0.10/0.70/0.20 (ICE-heavy)	25.50	1.450	0.604
0.20/0.30/0.50 (spec-heavy)	23.27	1.330	0.604
0.34/0.33/0.33 (uniform)	25.29	1.395	0.617

Table 21: Fusion-weight sensitivity on TOFU-forget10 (seed 11). R-L is 0.000 for all rows, and the tested emphasis shifts provide a stable tuning range for the operating point.

M Safeguard-Constant Sensitivity

Variation	$\log_{10}$ F-PPL $\uparrow$	R-PPL ≈ 1	Gen. $\uparrow$
Default	25.07	1.397	0.600
q_f center 1.5	25.77	1.429	0.588
q_f center 3.5	26.34	1.334	0.633
q_f scale 1.0	25.77	1.429	0.588
q_f scale 2.0	25.05	1.323	0.625
Retain denominator 3.0	25.71	1.427	0.592
Retain denominator 8.0	24.99	1.395	0.588

Table 22: Selected one-at-a-time safeguard perturbations on TOFU-forget10 (seed 11 diagnostic). Across all 12 tested perturbations, including six additional interval-span variants, $\log_{10}$ F-PPL ranges 24.26–27.96, R-PPL 1.214–1.491, and General Avg 0.554–0.633. The constants materially shift the operating point and are loss-scale dependent.

The MUSE-News calibration motivates a scale-relative controller map for future cross-domain deployment.

N Implementation Details

The supplementary material contains training, evaluation, localizer, and diagnostic scripts; the archival repository records the exact configurations and release tag for every reported table.

ICE computation. The intervention-inspired effect (ICE) score is computed before the main unlearning loop, after the initial activation stage and before localized multi-objective training. P1 independently samples 16 forget and 8 retain batches (batch size 1, maximum length 128). Grad $\times$ Weight uses the first six forget batches, ICE the first four forget batches, and specificity the first six forget plus six retain batches. For each LoRA-B tensor, the probe zeros the full tensor rather than individual rows, measures the forget-loss shift relative to the unperturbed baseline, and broadcasts the resulting scalar effect to the row dimension associated with that tensor. Before combination, every score channel is min–max normalized; channels with mismatched row length are linearly interpolated to the destination row dimension. The implementation uses the resulting ICE values as operational route-effect signals; repeated-probe variance estimation is a natural extension.

Baseline parity. All comparisons use the same shared-framework training scaffold. Table 24 records each baseline objective or steering mechanism and the matched training controls.

Field	Value
Base model / adapter	Qwen3-4B base checkpoint; LoRA rank 16, alpha 32, dropout 0.05
LoRA target modules	q/k/v/o projections; gate/up/down projections
Optimizer / main LR	AdamW, betas=(0.9, 0.95), weight decay 0.01; 1×10^{-4}
Batch / sequence length	Batch size 1; maximum sequence length 128
P0 activation	15 forget-side ascent steps at $0.8 \times$ main LR; LoRA-only updates
P0 retain check	Separate retain batches every 5 steps; stop above $1.5 \times$ pre-P0 retain PPL
P1 probe batches	16 forget and 8 retain, independently sampled; Grad $\times$ Weight uses 6 forget, ICE 4 forget, specificity 6 forget + 6 retain
P2 steps / routing density	300 steps; top- $k = 0.80$ by default; diagnostics include 0.30, 0.50, 0.70, and 1.00
Fusion weights	$(w_g, w_{ice}, w_{spec}) = (0.30, 0.50, 0.20)$ by default
Retain mask	EMA coefficient $\eta = 0.9$; threshold $\tau_r = 0.99$
KL anchor	Weight 0.15; evaluated every 10 steps on cached retain batches
Dynamic weight ranges	$\alpha_t \in [1.5, 4.0]$, $\beta_t \in [1.0, 2.5]$, $\gamma_t \in [0.05, 0.55]$
Controller	Hidden size 64; mixture/value heads; discount 0.99; Adam 10^{-4} ; update every 10 steps; Dirichlet exploration and gradient clipping
P2 retain check	Retain PPL estimated every 100 steps against the configured stop threshold
Main replication seeds	11, 22, and 33; secondary diagnostics state their seed separately
Evaluation sample size	100 where available
MUSE-News adaptation	Forget-control center/scale set to 0.5/2.0 using the initial loss scale

Table 23: Shared-framework hyperparameters and probe construction for the camera-ready revision. The MUSE-News adaptation is reported separately because it changes the loss-scale mapping used by the dynamic safeguard.

Method	LoRA r/α	LR / steps	Reference or matching detail	Direct run?	release-code
GA	16 / 32	1×10^{-4} / 300	Shared forget-only ascent scaffold	No; shared-framework implementation	
GA-retain	16 / 32	1×10^{-4} / 300	Forget ascent plus retain CE	No; shared-framework implementation	
NPO	16 / 32	1×10^{-4} / 300	Cached initial-model forget log-probabilities as the reference policy	No; shared-framework implementation	
RMU	16 / 32	1×10^{-4} / 300	Token-level representation misdirection with frozen retain activation matching; layer 7; steering coefficient 6.5; 32 retain reference batches	No; shared-framework implementation	
Ours	16 / 32	1×10^{-4} / 300	Localization-score routing, retain-aware masking, retain-logit KL cache, and dynamic weighting + safety	Project method, not an external recipe	

Table 24: Baseline parity summary for the shared-framework experiments, recording matched settings and method-specific control details.

O Dynamic Weighting Pseudocode

P Baseline Trade-off Sweep

Algorithm 2 Reward-weighted bounded safeguard

- 1: Initialize a hidden-64 MLP with mixture and value heads, Adam (10^{-4}), and a trade-off buffer.
- 2: **for** each P2 step t **do**
- 3: Compute bounded signals $(q_f, q_r, q_s, t/T)$ from forget loss, retain loss, and LoRA drift.
- 4: Produce a softmax mixture π_t and map it to bounded $(\alpha_t, \beta_t, \gamma_t)$.
- 5: Apply the routed multi-objective update; evaluate cached-retain KL when scheduled.
- 6: Store ΔJ_t , where $J_t = 0.4q_f + 0.4q_r + 0.2q_s$.
- 7: **if** t is a controller-update step **then**
- 8: Form normalized discounted returns (0.99), subtract the learned value baseline, and update with Dirichlet exploration and gradient clipping.
- 9: **end if**
- 10: Every 100 steps, estimate retain PPL and stop if the configured threshold is exceeded.
- 11: **end for**

We sweep the main suppression-strength parameter for NPO and RMU under the same shared framework: `np0_beta` for NPO and `rmu_steering_coeff` for RMU. The tables and plots map the reachable forgetting–utility regions, with R-PPL reported alongside the two plotted axes.

Method	Param	$\log_{10}$ F-PPL $\uparrow$	R-L $\downarrow$	R-PPL ≈ 1	Gen. $\uparrow$
Original	main	0.000	0.349	1.000	0.818
GA	main	7.372	0.000	2.72×10^7	0.303
GA-retain	main	7.372	0.082	202.383	0.515
NPO	$\beta=0.025$	4.784	0.280	16.999	0.788
NPO	$\beta=0.05$	4.559	0.220	7.935	0.788
NPO	$\beta=0.1$	4.124	0.246	2.577	0.788
NPO	$\beta=0.2$	3.286	0.169	1.036	0.768
NPO	$\beta=0.4$	2.456	0.165	0.607	0.778
RMU	$c=2.0$	-0.213	0.312	0.608	0.727
RMU	$c=4.0$	-0.213	0.312	0.608	0.727
RMU	$c=6.5$	-0.214	0.321	0.608	0.737
RMU	$c=8.0$	-0.213	0.321	0.609	0.717
RMU	$c=10.0$	-0.214	0.321	0.609	0.717
RMU	$c=13.0$	-0.214	0.321	0.610	0.717
Ours	main	4.871	0.205	1.496	0.633

Table 25: TOFU-forget10 baseline trade-off sweep. NPO/RMU vary one suppression-strength parameter; Ours is fixed at its main operating point. The table should be read with Figure 6.

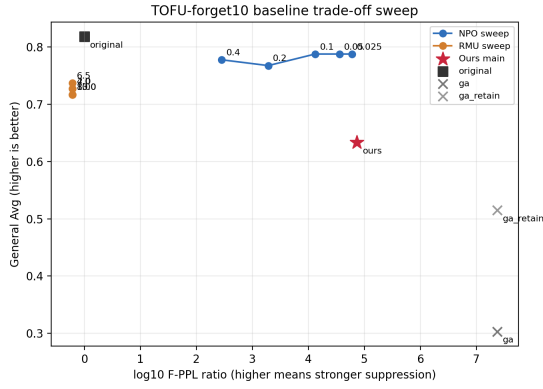


Figure 6: TOFU-forget10 baseline suppression-strength sweep. NPO and RMU vary one suppression-strength parameter; protected routing is shown at its canonical operating point.

Method	Param	UE Avg. $\uparrow$	R-PPL Avg. ≈ 1	Gen. Avg. $\uparrow$
Original	main	0.443	1.000	0.818
GA	main	0.773	1.60×10^7	0.347
GA-retain	main	0.710	0.652	0.293
NPO	$\beta=0.025$	0.703	0.503	0.320
NPO	$\beta=0.05$	0.710	0.456	0.310
NPO	$\beta=0.1$	0.710	0.447	0.350
NPO	$\beta=0.2$	0.707	0.444	0.337
NPO	$\beta=0.4$	0.703	0.442	0.354
RMU	$c=2.0$	0.743	0.601	0.448
RMU	$c=4.0$	0.743	0.601	0.455
RMU	$c=6.5$	0.737	0.602	0.448
RMU	$c=8.0$	0.743	0.601	0.448
RMU	$c=10.0$	0.727	0.603	0.444
RMU	$c=13.0$	0.730	0.602	0.444
Ours	main	0.621	1.822	0.517

Table 26: WMDP baseline trade-off sweep. NPO/RMU vary one suppression-strength parameter and are averaged over Bio, Chem, and Cyber; Ours is fixed at its main operating point. The table should be read with Figure 7.

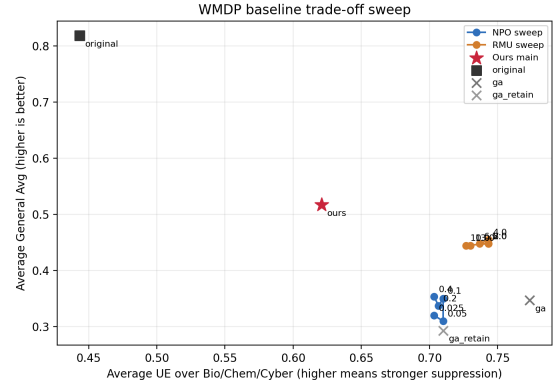


Figure 7: WMDP baseline suppression-strength sweep averaged over Bio, Chem, and Cyber. The accompanying table reports R-PPL for every point.