REXU: RECOVERY-EXPERT UNLEARNING FOR MIXTURE-OF-EXPERTS

Anonymous authors
Paper under double-blind review

ABSTRACT

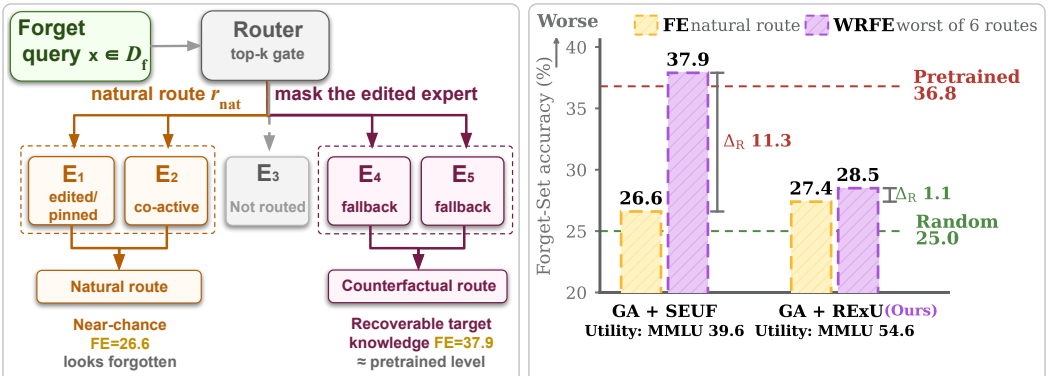
Machine unlearning aims to remove target knowledge while preserving other capabilities. Existing evaluations measure *forget efficacy* (FE) under normal inference, treating low FE as evidence of successful forgetting. This assumption can fail for Mixture-of-Experts (MoE) models, where each input activates only a sparse subset of experts: an unlearning method may suppress target knowledge along the natural route - the model's default expert selection without inference-time intervention - while leaving the same knowledge recoverable through alternative experts. We expose this *off-route recovery* through counterfactual routing and introduce **Worst-Route Forget Efficacy (WRFE)**, which measures the maximum forget-set recovery over an audited route family including the natural route. Statistically above-chance recovery under such rerouting provides direct evidence that supposedly forgotten knowledge remains accessible in the model. Empirically, we find substantial off-route recovery despite successful natural-route forgetting. On WMDP-Cyber, for example, GA+SEUF lowers FE by 10.22 percentage points, from 36.79% to 26.57%, under natural routing, yet bypassing the edited expert raises FE by 11.33 percentage points to 37.90%. Thus, natural-route FE alone can substantially underestimate residual recoverability in MoE models. To mitigate this failure, we propose **Recovery-Expert Unlearning (RExU)**, which iteratively identifies recovery experts contributing most to worst-route recovery and selectively unlearns them. Across three MoE models, four base unlearners, and two benchmarks, RExU substantially reduces WRFE while maintaining competitive natural-route FE and retain utility. Our results motivate jointly evaluating MoE unlearning through natural-route FE, WRFE, and retain utility, and establish RExU as a practical method for improving this trade-off.

1 INTRODUCTION

Machine unlearning aims to remove target knowledge such as hazardous capabilities, private or copyrighted content, from a trained model while preserving its other capabilities, at far lower cost than retraining from scratch (Cao & Yang, 2015; Li et al., 2024). Meanwhile, Mixture-of-Experts (MoE) language models have become an increasingly important architecture for scaling model capacity efficiently, activating only a small subset of experts for each token (Shazeer et al., 2017). An MoE prediction depends on the subset of router-selected experts, so different routing paths can expose different model behavior. However, standard machine-unlearning evaluation measures *forget efficacy* (FE) only under the "natural" routing and treats low FE as evidence of successful forgetting.

Research Gap. This interpretation can fail for MoE models: low FE under the natural route does not guarantee that target knowledge is no longer recoverable through alternative routes. This is possible because natural inference activates only a subset of experts, leaving knowledge stored in unselected experts untested. Existing unlearning methods interact with MoE routing in different ways. First, Full-model methods such as gradient ascent (GA; Yao & Xu 2024), gradient difference (GDIFF; Maini et al. 2024), NPO (Zhang et al., 2024), and RMU (Li et al., 2024) update all or most parameters without explicitly modeling routing. Second, localized methods such as SEUF (Zhuang et al., 2025) concentrate unlearning on high-affinity experts and use a routing anchor to stabilize their selection. Despite this targeted design, forgetting can remain *route-dependent*: masking the edited experts at inference time may redirect computation through unedited experts and recover target knowledge that appears forgotten under natural inference. Such rerouting is a practical white-

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.



(a) **Natural-route unlearning can hide recoverable target knowledge**

(b) **RExU achieves the best balance (Δ_R), substantially reducing both FE and WRFE while maintaining competitive utility.**

Figure 1: **Natural-route evaluation can hide recoverable target knowledge in MoE unlearning.** **(a)** GA+SEUF appears to forget under natural routing (FE 26.6), but an alternative route recovers near-pretrained target knowledge (WRFE 37.9). **(b)** RExU reduces the natural-to-worst-route recovery gap from 11.3 to 1.1 while improving retain utility.

box intervention for open-weight MoEs, rather than an exotic attack. More importantly, because it changes neither the weights nor the input, statistically above-chance recovery indicates that the target computation remains available in the fixed model. **Natural-route forgetting can therefore reflect path suppression rather than knowledge removal.**

In this paper, we systematically study *off-route recovery*, where target knowledge is recovered through alternative routing paths, using *counterfactual routing*. We intervene on the routing path and test whether target knowledge re-emerges through alternative experts. We formalize this behavior with *worst-route forget efficacy* (WRFE), defined as the maximum forget-set recovery over a family of route interventions. Our analysis shows that off-route recovery is largely driven by fallback experts that affinity-based selection can leave unedited. For example, on DeepSeek-V2-Lite with WMDP-Cyber, SEUF reduces natural-route FE to 26.57, yet masking the edited expert raises it to 37.90, close to the pretrained model under the same intervention (36.89). Motivated by this finding, we propose *Recovery-Expert Unlearning* (RExU), which identifies the route with the highest recovery within the audited route family, attributes that recovery to the responsible experts, and selectively unlearns them. This allows RExU to maintain competitive natural-route forgetting and retain utility while substantially reducing off-route recovery. Figure 1 illustrates both the recovery hidden by natural-route evaluation and how RExU narrows the natural-to-worst-route recovery gap while improving retain utility relative to SEUF. Our contributions are:

- **Off-route recovery and counterfactual routing audit.** We identify off-route recovery in MoE unlearning: low natural-route FE can hide target knowledge recoverable through alternative routes. We introduce a counterfactual routing audit and formalize worst-route forget efficacy (WRFE), the maximum forget-set recovery over an audited route family. See Section 3.
- **Failure modes and structural limits of MoE unlearning.** We show that natural-route FE can conflate route-fragile forgetting, utility collapse, and non-forgetting, and that affinity-based expert selection can miss recovery paths. Theoretically, we show that redundant experts can create functionally substitutable recovery routes, yielding route-cover conditions and optimization-independent lower bounds on the edit budget required for expert-local unlearning. See Section 4 and Appendix C.
- **Recovery-Expert Unlearning (RExU).** We propose RExU, which iteratively identifies the worst audited route, attributes its recovery to contributing experts, selectively unlearns those experts with the router frozen, and re-audits the route family. Across our main model-benchmark settings, RExU reduces SEUF's route gaps from 4.76-18.17 to 0.76- 2.71 points, while remaining within 0.08-2.30 utility points of the pretrained models. Results across three MoE models, four base unlearners, and two benchmarks consistently support this trend. See Section 5 and Section 6.

2 Related Work

Unlearning in MoE models. LLM unlearning uses post-hoc updates such as gradient ascent (GA; Yao & Xu 2024), gradient difference (GDIFF; Maini et al. 2024), NPO (Zhang et al., 2024), or RMU (Li et al., 2024); standard forget-set evaluation can overstate robustness, as knowledge reappears under data transformations, probing, relearning, or adversarial evaluation (Yuan et al., 2025; Joshi et al., 2024; Deeb & Roger, 2024). Prior interpretability work moreover shows that factual knowledge is distributed across many components and, in MoE models, across shared and routed experts rather than residing in a single expert (Dai et al., 2022; Li et al., 2025) - motivating our question of whether unlearning one expert removes the knowledge or merely one access path to it. Closest to us, SEUF (Zhuang et al., 2025) localizes unlearning to high-affinity experts and anchors the router against selection shift. We instead perturb the inference-time route rather than prompts, data, or optimization, showing the same anchor that stabilises natural-route evaluation is what makes forgetting route-fragile. We discuss general LLM unlearning and benchmarks, MoE architectures, routing side channels, knowledge localization, and recent route-robustness methods in Appendix A.

Generality of worst-route auditing. The underlying concern - that unlearning targets only the training-time computation path - applies beyond MoE to any input-conditioned architecture (Mixture-of-Depths, early-exit, conditional adapters); we focus on MoE since it is the predominant sparse architecture at scale, SEUF gives the cleanest localize-and-pin baseline, and router logits are directly accessible for white-box intervention.

3 Counterfactual Routing Audit

MoE. Let f_θ be a MoE language model, where θ denotes the full set of model parameters, including the router and expert weights. Let $\mathcal{L}$ denote the set of MoE layers. Each layer $l \in \mathcal{L}$ has E_l routed experts, and (l, e) denotes expert e in layer l. Given the representation of token x at layer l, the router produces logits $z_l(x) \in \mathbb{R}^{E_l}$ and the pre-top-k routing distribution

$$g_l(x) = \mathrm{softmax}\big(z_l(x)\big) \in \Delta^{E_l}. \tag{1}$$

It selects the top-k experts $\mathcal{T}_l(x) = \text{Top-}k\big(z_l(x)\big)$ and combines their outputs using the model's routing weights. Thus, the forward computation depends on which routed experts are selected.

Machine Unlearning. Given a forget set D_f and retain set D_r, post-hoc unlearning maps pretrained parameters θ_0 to θ_u with the goal of removing target knowledge in D_f while preserving non-target behavior. Conventional evaluation therefore considers retain utility and forget efficacy: retain utility $\mathrm{UT}(\theta)$ measures zero-shot MMLU accuracy under the natural route (*higher* is better), whereas forget efficacy measures the model's remaining performance on D_f (*lower* is better).

3.1 Counterfactual Routing Audit

Why does MoE unlearning need a Counterfactual Routing Audit? Natural routing exposes only one computation path for a given input. To test whether low natural-route forget efficacy reflects knowledge removal or merely suppression along the natural path, we intervene on routing decisions at inference time while holding the model parameters and input fixed. If a pre-specified alternative route produces statistically above-chance performance on the forget set, it provides a one-sided witness that target knowledge remains recoverable from the model.

Scope. The interventions below are *white-box* audits requiring access to model weights and router logits, and are therefore scoped to open-weight releases and provider-side auditing. Failure to recover supports robustness only with respect to the finite audited family $\mathcal{R}$ and does not rule out recovery under a larger family of routing interventions $\mathcal{R}' \supset \mathcal{R}$ (Lemma 1(c), Appendix C).

Definition 1 (Route intervention). *Let $T_l : \mathbb{R}^{E_l} \to \mathbb{R}^{E_l}$ be a function that, for a token x, acts on the logit vector $z_l(x)$. A* route intervention *r is a family of logit transforms $\{T_l^r\}_{l \in \mathcal{L}}$ applied before expert selection:*

$$z_l(x) \longmapsto T_l^r\big(z_l(x)\big). \tag{2}$$

Top-k expert selection and combination are then performed as usual using the transformed logits, and we write $f_\theta(x; r)$ for the resulting output. In particular, the natural route r_{nat} is the identity transform.

Remark 1. *The intervention leaves the model parameters θ and input x unchanged and does not use forget-set labels.*

These restrictions rule out distinct confounds: permitting parameter updates would measure *relearnability*, permitting input modifications would measure *prompt robustness*, and permitting label-dependent interventions could leak the answer being evaluated. Therefore, any above-chance recovery must arise from computation already available in the fixed model. Where the interventions use natural-route selection frequencies, these frequencies are computed without labels. An architecture-agnostic formulation is provided in Definition 4 of Appendix C.

We use three label-free intervention classes. They are not interchangeable: each targeting a distinct way that target knowledge can survive unlearning. Exact maps, the checkpoint-specific reference set S, and the numerical settings (σ, β, γ) are in Appendix D.

- **A1 (expert masking)** sets the router logits of a reference set S to $-\infty$, so top-k cannot select those experts. When S is the method's edited set, above-chance recovery under A1 is a certificate that the target is accessible *without the edited experts* (Lemma 1(b)). This is the probe that matters for localize-and-pin methods: they edit a small S and approximately freeze the router, so the natural route can look forgotten while an unedited fallback still implements the target.
- **A2 (gate perturbation)** adds Gaussian noise to the logits or partially re-uniformizes them. It does not remove an expert class; it asks whether a small change in gate mass is enough to restore the target. Full-model updates already move the router, so A1 is often uninformative for them; A2 is the probe that exposes their residual route sensitivity.
- **A3 (rare-expert forcing)** boosts experts that are rarely selected on the forget set under the natural route. Affinity-based selection concentrates edits on frequently chosen experts and can miss this long tail. A3 tests whether that tail is a fallback recovery path.

A *route family* $\mathcal{R}$ is a finite set containing r_{nat}. All reported worst-route scores use a six-route family

$$\mathcal{R} = \left\{ r_{\text{nat}},\ A1_S,\ A2_{\sigma=.05},\ A2_{\sigma=.10},\ A2_{\beta=.25},\ A3_{\gamma=1.0} \right\}. \tag{3}$$

Here, σ controls the Gaussian noise added to router logits, β the degree of re-uniformization, and γ the inverse-frequency logit boost applied to rare experts. For each checkpoint, $A1_S$ uses the reference set S specified in Appendix D. RExU selects experts against a four-route subset $\mathcal{R}_{\text{train}} \subset \mathcal{R}$ (Equation 8); $A2_{\sigma=.10}$ and $A3_{\gamma=1.0}$ are held out of selection and used only at reporting time. Appendix H compares these interventions with gradient-based and random route search on a representative GA+SEUF checkpoint.

3.2 Worst-Route Forget Efficacy

For multiple-choice benchmarks, FE is post-unlearning accuracy on the forget set (lower is better; Li et al. 2024; Maini et al. 2024). We make FE route-dependent: $\mathrm{FE}(\theta; r)$ reduces to the conventional metric when $r = r_{\text{nat}}$ and measures off-route recovery when $r \neq r_{\text{nat}}$. For a multiple-choice question q with answer options $\mathcal{A}$, where $|\mathcal{A}| = K$ and correct answer y, the *correct-answer margin* is

$$m_\theta(q; r) = \log p_\theta(y \mid q, \mathcal{A}; r) - \max_{\substack{a \in \mathcal{A} \\ a \neq y}} \log p_\theta(a \mid q, \mathcal{A}; r). \tag{4}$$

Under argmax decoding with ties broken against y, the model answers correctly iff $m_\theta(q; r) > 0$, so for a deterministic route

$$\mathrm{FE}(\theta; r) = \mathbb{E}_{q,y \sim D_f}\, \mathbb{I}\{m_\theta(q; r) > 0\}, \tag{5}$$

with chance level $1/K$. For stochastic routes (A2 noise), we audit N_s=3 fixed seeds and score the route by the sample maximum over seeds (Equation 88, Appendix D). Accordingly, the WRFE defined below is a maximum over the finite audited set $\mathcal{R} \times [N_s]$ and is defined relative to that set. Throughout the multiple-choice analysis, the margin serves as the attribution signal, whereas the indicator-based accuracy defines FE. The corresponding generation-based definitions for the real-world knowledge unlearning benchmark RWKU (Cao et al., 2024) are in Appendix D. We now aggregate route-specific FE into **WRFE**.

Definition 2 (Worst-route FE and route gap)**.** *For a route family $\mathcal{R}$,*

$$\mathrm{WRFE}_{\mathcal{R}}(\theta) = \max_{r \in \mathcal{R}} \mathrm{FE}(\theta; r),$$
$$\Delta_{\mathcal{R}}(\theta) = \mathrm{WRFE}_{\mathcal{R}}(\theta) - \mathrm{FE}(\theta; r_{\mathrm{nat}}). \tag{6}$$

Definition 3 (Route-robust unlearning)**.** *θ is $(\mathcal{R}, \tau)$-*route-robustly unlearned *if $\mathrm{WRFE}_{\mathcal{R}}(\theta) \leq p_{\mathrm{chance}} + \tau$ while $\mathrm{UT}(\theta) \geq u_0$, for tolerance τ and utility floor u_0, where $p_{\mathrm{chance}} = 1/K$ for multiple choice (Appendix D for generation).*

Three basic facts, formalized and proved as Lemma 1 (Appendix C), follow from properties of the maximum. *(a) Natural-route evaluation is optimistic:* since $r_{\mathrm{nat}} \in \mathcal{R}$, $\Delta_{\mathcal{R}}(\theta) \geq 0$, with equality iff the natural route attains the maximum. Thus, natural-route FE lower-bounds WRFE. *(b) Masking is a one-sided certificate:* for any feasible mask S that leaves at least k unmasked experts in each MoE layer, if $\mathrm{FE}(\theta; A1_S) > 1/K$, target knowledge is recoverable above chance through a route that activates no expert in S. When S is a localize-and-pin method's edited set, this recovery does not rely on any expert edited by the method. This certificate witnesses presence but never establishes absence. *(c) Audited WRFE is a lower bound:* $\mathcal{R} \subseteq \mathcal{R}'$ implies $\mathrm{WRFE}_{\mathcal{R}}(\theta) \leq \mathrm{WRFE}_{\mathcal{R}'}(\theta)$, so passing the audit on $\mathcal{R}$ does not certify robustness on a larger family.

4 Failure Modes of Existing MoE Unlearning

Under the same route family $\mathcal{R}$, we audit full-mode unlearning (GA, NPO, RMU) and localize-and-pin (SEUF and its multi-expert top-M extension SEUF-M) on DeepSeek-V2-Lite/WMDP-Cyber.

Table 1: **Natural-route FE can hide off-route recovery.** Results on DeepSeek-V2-Lite / WMDP-Cyber. FE measures forget-set accuracy (↓; chance: 25%). WRFE is the maximum FE over six audited routes, and $\Delta_{\mathcal{R}} = \mathrm{WRFE} - \mathrm{FE}_{\mathrm{nat}}$. MMLU measures natural-route retain utility (↑). $\mathrm{A2}_{.05}$/$\mathrm{A2}_{.10}$ denote gate-noise levels; $\mathrm{A2}_{\mathrm{re}}$ denotes re-uniformization.

Method	Nat. FE↓	A1 Mask	$\mathrm{A2}_{\sigma=0.05}$	$\mathrm{A2}_{\sigma=0.10}$	$\mathrm{A2}_{\mathrm{re}}$	A3 Rare	Worst↓	$\Delta_{\mathcal{R}}$ ↓	MMLU↑
Pretrained	36.79	36.89	36.99	36.99	36.49	36.94	36.99	0.20	55.51
Full-model unlearning									
GA	24.31	24.86	24.62	25.62	24.96	21.36	25.62	1.31	24.65
NPO	33.82	32.60	34.41	34.61	37.02	35.41	37.02	3.20	47.30
RMU	26.36	26.56	27.57	27.36	26.36	26.76	27.57	1.21	46.33
Localize-and-pin									
GA+SEUF (top-1)	26.57	37.90	26.62	26.62	26.57	26.57	37.90	11.33	39.57
GA+SEUF-M (top-4)	28.77	37.00	29.38	27.77	27.77	28.97	37.00	8.23	32.30
GA+SEUF-M (top-8)	28.17	36.80	28.77	29.18	27.77	27.57	36.80	8.63	29.45

Finding 1: Natural-route FE hides off-route recovery after localize-and-pin unlearning. GA+SEUF (top-1) reaches near-chance FE of 26.57, yet masking its edited expert raises FE to 37.90, a route gap of 11.33 (Table 1). Conditional recovery is 82.6% for GA+SEUF; detailed counts and statistical tests are in Appendix F.2 (Table 6). Gate noise and rare-expert forcing expose little recovery in this case. A1 recovery shows that target knowledge remains recoverable without activating the edited expert.

Finding 2: FE, WRFE, and utility distinguish different failure modes. SEUF and RMU have nearly identical natural-route FE (26.57 vs. 26.36), yet differ in WRFE and retain utility. SEUF hides substantial off-route recovery; GA achieves low FE with severe utility collapse, RMU shows milder utility loss, and NPO leaves target knowledge recoverable. The worst route also varies: A1 exposes the most recovery for the SEUF family, whereas A2 is worst for the full-model methods tested. Table 1 gives the per-route results; Appendix F.2 provides further comparisons. WRFE therefore summarizes recovery within the audited family, not robustness beyond it (Definition 2).

Finding 3: Affinity selection misses recovery experts, while scaling SEUF degrades utility. Natural-route affinity misses many recovery experts, including those most responsible for off-route recovery. In a matched-budget controlled comparison without anchoring (Table 11), affinity-based

selection yields 32.81 WRFE, compared with 28.54 when experts are selected by their measured contribution to route recovery. Increasing SEUF's edited set from one to eight experts barely reduces WRFE (37.90 to 36.80) while lowering MMLU from 39.57 to 29.45 (Table 1). Scaling changes both the edited set and the anchoring constraint, so it cannot isolate their effects; the matched-budget comparison isolates selection. Anchoring may further hurt utility by restricting routing on retain inputs. Attribution definitions, expert-overlap measurements, and scaling details appear in Appendix F.2 (Table 8), with matched-budget ablations in Appendix F.3 (Table 11).

Implications for MoE unlearning evaluation. Low natural-route FE does not necessarily imply that the forget target is no longer recoverable. Substantial *off-route recovery* may remain under alternative routes. Thus, MoE unlearning should be evaluated jointly using natural-route FE, WRFE, and retain utility, with robustness claims restricted to the audited routes.

Expert-local edits cannot change computations that avoid the edited set. Let θ_S modify only experts in S, with all other parameters fixed, and let $P_{\theta_0}(q, r)$ denote the reference expert support. Under identical routing rules and randomness,

$$P_{\theta_0}(q, r) \cap S = \varnothing \implies f_{\theta_S}(q; r) = f_{\theta_0}(q; r). \tag{7}$$

This invariance yields optimization-independent WRFE lower bounds and necessary edit budgets, motivating RExU's repeated auditing and recovery-driven selection. We encourage readers to refer to Appendix C for the full theoretical analysis. It explains why unlearning in MoE models can be intrinsically difficult: over-specification allows the same functional behavior to be represented by multiple experts, softmax routing can maintain several experts specialized to the same mode, and partial expert editing may therefore leave alternative recovery routes intact. As the model becomes more over-specified, additional experts can provide more functionally substitutable routes, making localized removal increasingly difficult when such redundancy is realized. The analysis further derives route-cover and minimum-edit-budget conditions showing when expert-local unlearning is structurally insufficient. Finally, it shows when a finite routing audit can still control unseen recovery: if the audited routes cover the admissible routing space sufficiently well and recovery varies smoothly between nearby routes, then the worst audited route approximates the global worst route, and RExU reduces global worst-route recovery whenever its targeted reduction dominates both recovery shifted to other routes and the finite-audit coverage error. **MoE unlearning is challenging because redundant experts can provide functionally substitutable routes, allowing target knowledge to remain recoverable even after the experts used by the natural route are edited.**

5 Recovery-Expert Unlearning (RExU)

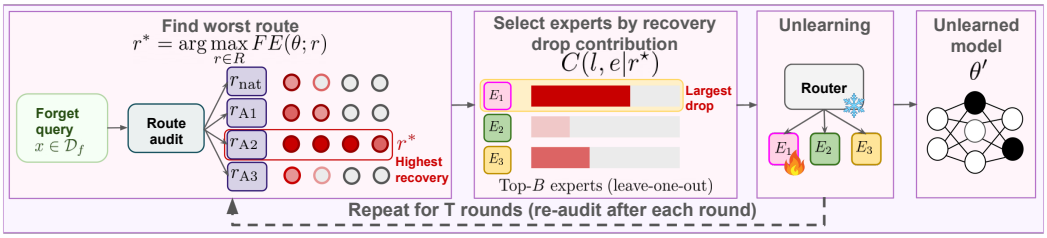


Figure 2: **Overview of RExU. Our method follows a simple principle: *if an unlearned model still recovers target knowledge under an audited route, the experts responsible for that recovery should be identified and unlearned*.** Following this principle, RExU targets the current worst route $r_t^\star$, selects experts by their recovery contribution, selectively unlearns them with the router frozen, and re-audits the route family, repeating this process for T rounds. Figure 13 in Appendix J contrasts RExU's recovery-driven, iterative selection with SEUF's natural-route affinity-based localization.

Section 4 identifies off-route recovery as a failure mode of existing MoE unlearning, particularly SEUF. We propose RExU to mitigate this failure. Figure 2 illustrates the RExU pipeline, which consists of three key components: (1) identifying the current worst audited route, (2) selecting experts by their recovery contribution and selectively unlearning them with the router frozen, and (3) re-auditing the route family to guide subsequent rounds. We ablate these components in Section 6.

5.1 WORST-ROUTE SELECTION.

At each round, RExU selects experts using a *training* route family:

$$\mathcal{R}_{\text{train}} = \{r_{\text{nat}},\ A1_{S_t},\ A2_{\sigma=.05},\ A2_{\beta}\} \tag{8}$$

a subset of the six-route audit family $\mathcal{R}$ of Section 3.1, where $A1_{S_t}$ masks the experts edited so far by the current run. The remaining two routes, $A2_{\sigma=.10}$ and rare-expert forcing $A3_{\gamma}$, are used only for final evaluation. Since $\mathcal{R}_{\text{train}} \subset \mathcal{R}$, reported WRFE is at least the training-family WRFE and includes two routes unseen during unlearning. At round t we select the highest-recovery route:

$$r_t^\star \;=\; \arg\max_{r \in \mathcal{R}_{\text{train}}} \text{FE}(\theta_t; r), \tag{9}$$

ties broken toward the route listed first in Equation 8, so that $\text{FE}(\theta_t; r_t^\star) = \text{WRFE}_{\mathcal{R}_{\text{train}}}(\theta_t)$ - the hardest constraint to satisfy in Definition 3 .

5.2 RECOVERY-EXPERT ATTRIBUTION AND UNLEARNING.

We denote an input sequence by $q = (x_1, \ldots, x_{|q|})$, where x_j is its j-th token. For multiple-choice examples, q includes the question and answer options. Forget examples are $(q, y) \sim D_f$, where y is the correct answer, while $q \sim D_r$ denotes a retain input. We make the reference answer explicit in the margin notation, writing $m_\theta(q, y; r)$.

Two-stage attribution. Computing recovery contribution requires one counterfactual forward pass per candidate. We first construct a candidate pool using *counterfactual routing affinity*, the mean gate mass on forget inputs under the worst route:

$$A_t(l, e) = \mathbb{E}_{(q,y)\sim D_f}\left[\frac{1}{|q|}\sum_{j=1}^{|q|} g_{l,e}(q, j; r_t^\star)\right], \tag{10}$$

where $g_{l,e}(q, j; r) = [\text{softmax}\, T_l^r(z_l(q, j; r))]_e$ is the pre–top-k router probability of expert (l, e) at token position j. Here, $z_l(q, j; r)$ denotes the router logits computed under the current model θ_t and route r, before applying T_l^r at layer l. These gate statistics use only the input q. We keep the N_c highest-affinity experts as the candidate pool $\mathcal{P}_t$.

High affinity, however, might not imply responsibility for recovery. Therefore, we re-rank candidates by their *recovery contribution*. Let $\theta_t^{-(l,e)}$ denote a counterfactual evaluation of θ_t in which expert (l, e)'s output is replaced by its retain-set mean activation, while all model parameters remain fixed. The distribution $p_{\theta_t^{-(l,e)}}(\cdot \mid q; r)$ and margin $m_{\theta_t^{-(l,e)}}(q, y; r)$ are computed from this ablated forward pass under the same intervention r. We define

$$C_t(l, e) = \mathbb{E}_{(q,y)\sim D_f}\left[m_{\theta_t}(q, y; r_t^\star) - m_{\theta_t^{-(l,e)}}(q, y; r_t^\star)\right].$$

Thus, $C_t(l, e) > 0$ means ablating (l, e) reduces the mean correct-answer margin under $r_t^\star$. RExU selects up to B candidates with the largest positive contribution:

$$\mathcal{E}_t = \text{Top-}B_{\,(l,e)\in\mathcal{P}_t:\, C_t(l,e)>0}\; C_t(l, e), \tag{11}$$

where B is the per-round budget, distinct from K and the router's top-k. If no candidate satisfies $C_t(l, e) > 0$, then $\mathcal{E}_t = \varnothing$ and the round makes no edit (Section 5.2).

Pluggable Expert Unlearning. Given a base unlearner $\mathcal{U}$, RExU applies its optimization steps only to the selected experts:

$$\theta_{t+1} = \mathcal{U}\big(\theta_t, \mathcal{E}_t, D_f, D_r, r_t^\star\big), \tag{12}$$

optimizing only the parameters of experts in $\mathcal{E}_t$ and freezing all others, including the router. The same intervention $r_t^\star$ is used throughout the forget update. For our margin-based instantiation, the forget objective is

$$L_{\text{forget}}(\theta) = \mathbb{E}_{(q,y)\sim D_f}\left[\text{softplus}\left(m_\theta(q, y; r_t^\star) + \delta\right)\right], \qquad \delta > 0, \tag{13}$$

which encourages the correct-answer margin below $-\delta$. By Equation 5, a negative margin flips the prediction away from y; the offset δ encourages an additional margin beyond this decision boundary.

A KL term relative to the round's starting iterate $\theta_{\text{ref}} = \theta_t$ regularizes retain behavior under natural routing:

$$L_{\text{retain}}(\theta) = \mathbb{E}_{q\sim D_r}[\text{KL}(p_{\theta_{\text{ref}}}(\cdot \mid q; r_{\text{nat}}) \,\|\, p_\theta(\cdot \mid q; r_{\text{nat}}))], \tag{14}$$

giving the per-round objective $L_{\text{total}} = L_{\text{forget}} + \lambda L_{\text{retain}}$, optimized over $\{\theta^{(l,e)} : (l,e) \in \mathcal{E}_t\}$. Here, $\mathcal{U}$ takes a prescribed number of optimization steps on L_{total}, initialized at θ_t. Other base unlearners use their objectives under the same selective-update constraint. RExU's recovery-driven route and expert selection therefore remain independent of the particular forgetting objective.

Recursive re-auditing. The detailed procedure of RExU is given in Algorithm 1 in Appendix D. RExU can be viewed as *constraint generation*. At round t, it re-audits $\mathcal{R}_{\text{train}}$, takes the most violated constraint $r_t^\star$ (Equation 9), and edits the experts most responsible for recovery there (Equation 11). Recovery is not monotone across routes: suppressing recovery on one route can raise it on another (Appendix I), so a route that was not previously worst can become the next constraint. RExU therefore repeats this audit-attribute-unlearn cycle until the stopping criterion is met or T rounds are completed. Among the iterates satisfying the utility floor, RExU returns the one with the lowest training-family WRFE. If the stopping criterion is met, measured FE satisfies the prescribed forgetting threshold on every route in $\mathcal{R}_{\text{train}}$, and retain utility meets the required floor. To assess robustness beyond the routes used for selection, we report WRFE on the larger family $\mathcal{R} \supset \mathcal{R}_{\text{train}}$, which includes two interventions never used for route selection (Section 6).

6 Experiments

6.1 Setup

Models. Following (Zhuang et al., 2025), our main experiment use Qwen (Qwen1.5-MoE-A2.7B-Chat) and DeepSeek (DeepSeek-V2-Lite) for our main results. To test whether off-route recovery and the effect of RExU to stronger, more recent MoE model, we additionally evaluate Qwen3-Next-80B-A3B-Instruct in Appendix F.2 (Table 9). Details and hyperparameters are in Appendix D.

Benchmarks. We evaluate on WMDP-Cyber and RWKU. For WMDP-Cyber, it is a four-choice benchmark, FE is post-unlearning accuracy (FE↓; chance level 25%). For RWKU, FE is measured by ROUGE-L recall (FE↓). We measure retain utility using zero-shot MMLU accuracy (UT↑). Appendix E provides example questions, reference answers, and evaluation formats.

Unlearning Baseline. Similar to SEUF, RExU is a *plug-and-play* MoE unlearning method that can be combined with different base unlearner. We consider GA, GDIFF, NPO, and RMU as base unlearners. For each base unlearner, we compare the base unlearner alone, base+SEUF, and base+RExU in the main paper. Additional results and discussion for parameter-efficient baselines (GA+LoRA (Hu et al., 2022) and GA+ESFT (Wang et al., 2024)), and two recent route-robustness methods (GRIP (Zhu et al., 2026) and TRACE (Xie et al., 2026)) are in Appendix F.2 (Table 10).

Metrics. We report standard natural-route FE, retain utility (UT), and WRFE. WRFE is always computed as the maximum over the full audit family $\mathcal{R}$.

6.2 Results

RExU improves the worst-route robustness-utility frontier across base unlearners. Across all model-benchmark settings, Figure 3 shows this robustness-utility explicitly with base+RExU points generally towards the lower-right corner. This indicates that base+RExU achieves a lower WRFE than base+SEUF while maintaining significantly higher utility. Base unlearners sometimes obtain lower WRFE, but this comes at the cost of a large utility drop. This suggests that base unlearners could achieve low FE/WRFE when collapsing retain utility. Detailed numerical results can be found in Table 5. Additional analysis is in Appendix F.2. Qualitative results are in Table 14, Appendix G.

Recursive re-auditing progressively reduces worst-route recovery. Increasing the number of RExU rounds lowers WRFE while largely preserving utility (Table 2). A single round reduces recovery, but additional rounds are needed because suppressing one recovery route can expose another. This supports the recursive audit-unlearn-re-audit design of RExU.

Targeting the worst route is more effective than fixed routing. With unlearning and expert selection fixed, targeting the current worst route $r^\star$ yields lower WRFE than targeting the natural route

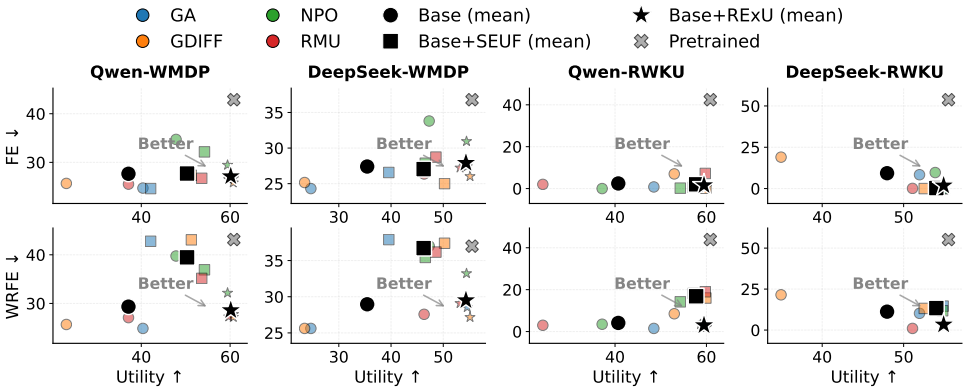


Figure 3: **SEUF's low natural-route FE can conceal substantial off-route recovery, whereas RExU achieves lower WRFE and higher utility than its corresponding SEUF variants across all four settings.** Each column shows one model-benchmark setting. The top and bottom rows plot natural-route FE and WRFE, respectively, against retain utility (zero-shot MMLU accuracy); lower-right is better. Colors identify the base unlearning objective: GA, GDIFF, NPO, and RMU. Circles (●), squares (■), and stars (★) denote Base, Base+SEUF, and Base+RExU, respectively. **Black** markers show each method family's mean across the four objectives; gray crosses (✖) mark pretrained models.

Table 2: **Ablation of RExU components on DeepSeek-V2-Lite/WMDP-Cyber.** Recovery-contribution selection yields lower WRFE than random or affinity selection at the same expert budget; additional rounds and worst-route selection also lower WRFE, but edit more distinct experts.

Variant	Nat. FE↓	WRFE↓	UT↑	Experts
(1) Number of rounds				
$T = 1$	27.88	31.10	54.70	8
$T = 5$ (default)	27.43	28.54	54.59	40
(2) Route selection ($T = 5$, $B = 8$)				
Natural route r_{nat}	26.62	36.49	54.80	8
Worst route $r^\star$ (default)	27.43	28.54	54.59	40
(3) Expert selection under $r^\star$ ($T = 5$, $B = 8$)				
Random experts	29.49	34.22	54.30	40
Affinity under $r^\star$	28.18	30.70	53.90	40
Leave-one-out contribution (default)	27.43	28.54	54.59	40

(Table 2). This shows that explicitly targeting the currently strongest recovery path is important for reducing off-route recovery.

Recovery contribution identifies more effective experts than affinity. Under the same route and edit budget, selecting experts by recovery contribution achieves lower WRFE than affinity-based or random selection (Table 2). For example, on DeepSeek/WMDP, replacing recovery contribution with affinity under $r^\star$ increases WRFE from 28.54 to 30.70, while natural-route affinity reaches 32.81 under the matched-budget comparison (Table 11). This indicates that expert activation alone is not sufficient for identifying the experts most relevant to recovery.

7 Conclusion

MoE unlearning is structurally difficult: redundant experts can preserve alternative routes to the same target knowledge, so low natural-route FE may indicate path suppression rather than true forgetting. We expose this failure with counterfactual routing audits and WRFE, and follow a simple principle with RExU: identify and unlearn the experts responsible for any remaining recovery.

Limitation. WRFE only measures recovery within a finite audited route family, so low WRFE does not prove total knowledge removal. RExU reduces but does not eliminate recoverability. See Appendix K for more.

Ethics Statement

This work studies how supposedly unlearned knowledge can be recovered in MoE models, including on hazardous-knowledge benchmarks. This creates a dual-use risk: the same audit that helps defenders diagnose incomplete unlearning could also guide misuse in open-weight settings. We frame counterfactual routing as a defender-side robustness audit, report aggregate benchmark results rather than operational harmful content, and encourage future work on MoE unlearning to include responsible disclosure practices and to report worst-route recovery as part of safety evaluation before model release.

Reproducibility Statement

Code, training scripts, hyperparameter configurations, and precomputed evaluation logs are available at the anonymous repository listed in Appendix D, together with software versions, hardware, and GPU-hour accounting. Algorithm 1 specifies the full RExU loop; the six-route audit family, seeds, and all intervention parameters are given in Section 3.1 and Appendix D; complete proofs of every formal claim appear in Appendix C.

AI Use Statement

This research utilized AI assistance for code debugging and grammatical refinement. All experimental designs, analyses, results, and conclusions were developed independently by the authors without generative AI input. We employed AI tools solely for technical implementation support and language polishing to ensure clear communication of our findings.

References

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

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

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE symposium on security and privacy*, pp. 463–480. IEEE, 2015.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8493–8502, 2022.

Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Yu Wu, et al. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1280–1297, 2024.

Aghyad Deeb and Fabien Roger. Do unlearning methods remove information from language model weights? *arXiv preprint arXiv:2410.08827*, 2024.

Ruyi Ding, Tianhong Xu, Xinyi Shen, Aidong Adam Ding, and Yunsi Fei. Moecho: Exploiting side-channel attacks to compromise user privacy in mixture-of-experts llms. In *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2159–2173, 2025.

Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zico Kolter, Zachary Lipton, and Pratyush Maini. Openunlearning: Accelerating llm unlearning via unified benchmarking of methods and metrics. *Advances in Neural Information Processing Systems*, 38, 2026.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 12 2023. URL https://zenodo.org/records/10256836.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens Van Der Maaten. Certified data removal from machine learning models. *arXiv preprint arXiv:1911.03030*, 2019.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

Shengyuan Hu, Neil Kale, Pratiksha Thaker, Yiwei Fu, Steven Wu, and Virginia Smith. Blur: A benchmark for llm unlearning robust to forget-retain overlap. *arXiv preprint arXiv:2506.15699*, 2025.

Wentao Hu, Yanbo Zhai, Xiaohui Hu, Mingkuan Zhao, Xue Liu, Kaidong Yu, Shuangyong Song, Xuelong Li, et al. Awakening dormant experts: Counterfactual routing to mitigate moe hallucinations. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 47257–47271, 2026.

Jiabao Ji, Yujian Liu, Yang Zhang, Gaowen Liu, Ramana R Kompella, Sijia Liu, and Shiyu Chang. Reversing the forget-retain objectives: An efficient llm unlearning framework from logit difference. *Advances in Neural Information Processing Systems*, 37:12581–12611, 2024.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts. *CoRR*, abs/2401.04088, 2024. doi: 10.48550/ARXIV.2401.04088. URL https://doi.org/10.48550/arXiv.2401.04088.

Abhinav Joshi, Shaswati Saha, Divyaksh Shukla, Sriram Vema, Harsh Jhamtani, Manas Gaur, and Ashutosh Modi. Towards robust evaluation of unlearning in llms via data transformations. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 12100–12119, 2024.

Jaehan Kim, Minkyoo Song, Seungwon Shin, and Sooel Son. Safemoe: Safe fine-tuning for moe llms by aligning harmful input routing. In *The Fourteenth International Conference on Learning Representations*, 2026.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*, 2020.

Junzhuo Li, Bo Wang, Xiuze Zhou, Peijie Jiang, Jia Liu, and Xuming Hu. Decoding knowledge attribution in mixture-of-experts: A framework of basic-refinement collaboration and efficiency analysis. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 22431–22446, 2025.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, et al. The wmdp benchmark: Measuring and reducing malicious use with unlearning. *arXiv preprint arXiv:2403.03218*, 2024.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013/.

Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Yuguang Yao, Chris Yuhao Liu, Xiaojun Xu, Hang Li, et al. Rethinking machine unlearning for large language models. *Nature Machine Intelligence*, 7(2):181–194, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

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

Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Evan Pete Walsh, Oyvind Tafjord, Nathan Lambert, Yuling Gu, Shane Arora, Akshita Bhagia, Dustin Schwenk, David Wadden, Alexander Wettig, Binyuan Hui, Tim Dettmers, Douwe Kiela, Ali Farhadi, and et al. Olmoe: Open mixture-of-experts language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=xXTkbTBmqq.

Thanh Nguyen-Tang and Raman Arora. Exact unlearning in reinforcement learning. *arXiv preprint arXiv:2606.04182*, 2026.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 8024–8035, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

Anil Ramakrishna, Yixin Wan, Xiaomeng Jin, Kai-Wei Chang, Zhiqi Bu, Bhanukiran Vinzamuri, Volkan Cevher, Mingyi Hong, and Rahul Gupta. Lume: Llm unlearning with multitask evaluations. *arXiv preprint arXiv:2502.15097*, pp. 10, 2025.

Raj Sanjay Shah, Jing Huang, Keerthiram Murugesan, Nathalie Baracaldo, and Diyi Yang. The unlearning mirage: A dynamic framework for evaluating llm unlearning. *arXiv preprint arXiv:2603.11266*, 2026.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

Zihan Wang, Deli Chen, Damai Dai, Runxin Xu, Zhuoshu Li, and Yu Wu. Let the expert stick to his last: Expert-specialized fine-tuning for sparse architectural large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 784–801, 2024.

Jingyi Xie, Yijun Lin, Yinjiang Xiong, Zhikun Zhang, and Sai Li. Routing-aware expert calibration for machine unlearning in mixture-of-experts language models. *arXiv preprint arXiv:2606.10338*, 2026.

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

Itay Yona, Ilia Shumailov, Jamie Hayes, and Nicholas Carlini. Stealing user prompts from mixture of experts. *arXiv preprint arXiv:2410.22884*, 2024.

Hongbang Yuan, Zhuoran Jin, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. Towards robust knowledge unlearning: An adversarial framework for assessing and improving unlearning robustness in large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 25769–25777, 2025.

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

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M Dai, Quoc V Le, James Laudon, et al. Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems*, 35:7103–7114, 2022.

Andy Zhu, Rongzhe Wei, Yupu Gu, and Pan Li. Grip: algorithm-agnostic machine unlearning for mixture-of-experts via geometric router constraints. *arXiv preprint arXiv:2601.16905*, 2026.

Haomin Zhuang, Yihua Zhang, Kehan Guo, Jinghan Jia, Gaowen Liu, Sijia Liu, and Xiangliang Zhang. Seuf: Is unlearning one expert enough for mixture-of-experts llms? In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8664–8678, 2025.

APPENDIX CONTENTS

This supplementary material provides additional details on counterfactual routing audits and Recovery-Expert Unlearning (RExU) for Mixture-of-Experts language models. It includes extended related work, notation, theoretical analysis, experimental details and dataset examples, detailed baseline comparisons, additional quantitative and qualitative results, ablation studies, intervention-sufficiency experiments, an analysis of recovery dynamics across unlearning rounds, and limitations.

The supplementary material is organized as follows:

A More Related Works . . . 15
B Notation and Abbreviations . . . 17
C Theoretical Analysis . . . 19
C.1 Margin and Forget Accuracy . . . 19
C.2 Worst-Route Forgetting . . . 19
C.3 Non-Identifiability and Over-Specification in Softmax-Gated Mixtures . . . 20
C.4 Softmax Competition and Redundant Expert Specialization . . . 22
C.5 Redundant Mode Preservation under Partial Expert Editing . . . 26
C.6 Knowledge-Route Cover and Necessary Intervention . . . 29
C.7 Minimum Edit Budget and Structural Impossibility . . . 32
D Experiment Details . . . 38
E Dataset Examples . . . 42
F Detailed Results . . . 43
F.1 Detailed Comparison Results with Other Baselines . . . 43
F.2 Additional Results . . . 44
F.3 Ablation Study . . . 52
G Qualitative Analysis of Off-Route Recovery . . . 58
H Intervention Sufficiency: Full Details . . . 60
I Balloon-Squeeze Dynamics in RExU . . . 62
J Comparison between SEUF and RExU . . . 64
K Limitations . . . 65

A More Related Works

Machine unlearning for language models. Early work introduced exact deletion and scalable approximate or certified removal for classical systems (Cao & Yang, 2015; Bourtoule et al., 2021; Guo et al., 2019; Nguyen-Tang & Arora, 2026). For LLMs, where retraining is infeasible, methods update on a forget set while regularizing retain behavior via gradient ascent, gradient difference, preference-based forgetting, or representation steering (Yao & Xu, 2024; Liu et al., 2025; Ji et al., 2024). These target dense computation graphs or are applied to MoE models without auditing routing. Our masking "certificate" (Lemma 1(b)) is unrelated to certified removal in the sense of Guo et al. (2019): it is a one-sided witness that knowledge *remains*, and low WRFE certifies nothing beyond the audited family (Lemma 1(c)).

Unlearning benchmarks. Benchmarks test forgetting without capability loss: TOFU uses fictitious authors against a retraining ideal (Maini et al., 2024); WMDP targets hazardous bio/cyber/chem knowledge (Li et al., 2024); RWKU covers real-world entities and MUSE adds a six-way protocol (Cao et al., 2024); OpenUnlearning and LUME standardize metrics and reproducibility (Dorna et al., 2026; Ramakrishna et al., 2025). A complementary line stress-tests the evaluations themselves: BLUR shows unlearning judged robust on standard splits collapses under benign forget-retain overlap and relearning (Hu et al., 2025), RWKU ships adversarial-probe and membership-inference suites (Cao et al., 2024), and OpenUnlearning meta-evaluates metric faithfulness and robustness (Dorna et al., 2026). These probes perturb the *input* or resume *training*; WRFE fixes both and perturbs the computation path, so the axes compose rather than compete.

Sparse Mixture-of-Experts language models. MoE models route each token to a few experts, improving the capacity-compute trade-off (Shazeer et al., 2017; Lepikhin et al., 2020; Fedus et al., 2022; Zhou et al., 2022). Open MoE LLMs such as Mixtral, DeepSeekMoE/DeepSeek-V2, and OLMoE show the practical value of sparse routing at scale (Jiang et al., 2024; Dai et al., 2024; Muennighoff et al., 2025). The router decides which parameters-and thus which experts-can express or suppress a target behavior, making routing a first-class object in MoE unlearning evaluation.

Routing as a surface and MoE safety. MoE routing can leak information: attacks on expert-choice routing recover prompts from token dropping in batched inference (Yona et al., 2024), and MoEcho exposes expert-execution side channels enabling prompt inference or response reconstruction (Ding et al., 2025). Related safety work, SafeMoE, regularizes harmful-input routing during fine-tuning (Kim et al., 2026). These target privacy leakage or training-time safety; we instead treat routing as a post-unlearning recovery surface, testing whether counterfactual routes recover knowledge declared forgotten. Counterfactual routing has also been used with the opposite objective: CoR steers MoE routes at inference time to *awaken* dormant experts and reduce hallucination, scoring experts by counterfactual impact (Hu et al., 2026). That routing interventions recover latent knowledge for factuality gains independently corroborates the premise of our audit - the same mechanism that awakens dormant knowledge can resurrect knowledge declared forgotten - and expert-impact scores of this kind could serve as alternative attribution signals inside RExU's loop.

Knowledge localization, editing, and expert attribution. Our expert attribution relates to locating knowledge in networks: knowledge-neuron and causal-tracing studies identify units mediating factual predictions (Dai et al., 2022), and recent MoE interpretability extends attribution to shared and routed experts (Li et al., 2025). We share the causal-intervention view but target experts whose ablation reduces forget-set recovery under counterfactual routes, distinguishing experts merely active on the natural route from those that make unlearning fail elsewhere.

Relation to SEUF-M SEUF-M and RExU both edit a fixed expert budget but differ in the selection criterion. SEUF-M picks the experts with highest *natural-route* affinity; RExU picks those whose ablation most reduces *off-route* recovery, editing (l, e) only when $C_t(l, e) > 0$ on the failing route $r^\star$ (Equation 89). Adding natural-route experts can damage high-retain-load experts that do not contribute to off-route recovery (Finding 3), lowering utility without closing the worst route, whereas RExU targets exactly the experts that cause recovery. The two are comparable only at a matched edit budget-SEUF-M's M experts versus RExU's total edited set $\bigcup_t E_t$ (at most TB, fewer when rounds re-select the same experts)-which isolates the criterion from the number of experts edited; under

it RExU attains a better robustness-utility frontier: 28.54 WRFE at 54.59 MMLU for recovery-contribution selection against 32.81 at 51.70 for natural-affinity selection at the same 40-expert budget (Table 11). When recovery on $r^\star$ is spread across a coalition rather than attributable to any single expert ($C_t(l, e) \leq 0$ for all candidates), E_t is empty and the loop cannot reduce that route; extending attribution beyond leave-one-out is left to future work.

Relation to GRIP. GRIP (Zhu et al., 2026) addresses routing shortcuts in MoE unlearning through geometric constraints on router updates. It preserves retain-side routing through training-time gradient projection or post-training analytical correction. GRIP and RExU both account for the interaction between routing and unlearning, but intervene at different stages of the optimization process. GRIP controls the geometry of router updates, whereas RExU uses counterfactual routing audits to determine which experts to edit. Specifically, RExU identifies a high-recovery route, ranks candidate experts by their ablation-based contribution to recovery, and selectively updates those experts while keeping router weights frozen. Repeated auditing then guides subsequent expert selection, making recovery under alternative routes an explicit signal for allocating unlearning updates.

Relation to TRACE. TRACE (Xie et al., 2026) addresses the mismatch between forget-side and retain-side expert activation in MoE unlearning. It identifies forget-critical experts using offline activation statistics and reweights token-level retain losses to align retain regularization with those experts' forget-side usage. RExU focuses on the contribution of individual experts to recovery under counterfactual routes. Its selection criterion measures how expert ablation changes the recovery signal on the current highest-recovery route, and the selected expert set is reconsidered after each round of unlearning. Both approaches incorporate expert-level structure into unlearning: TRACE calibrates the retain objective according to expert usage, while RExU directs selective updates through recovery attribution and iterative route auditing.

Token- and belief-level unlearning. A separate line of work targets robustness to paraphrase and surface-form variation by operating below the level of a fixed question-answer pair - e.g. suppressing target tokens directly during decoding, or updating a model's internal beliefs rather than its response to one specific prompt phrasing. This is a different axis of robustness from the one we study: our route interventions (A1-A3) fix the prompt and vary which computation answers it, while token/belief-level methods fix the computation path and vary the prompt or decoding procedure. The two are not mutually exclusive - a method robust to paraphrase could still be route-fragile, and vice versa - and combining route auditing with paraphrase/token-level robustness checks is a natural extension we did not pursue. Dynamic structured probing - multi-hop chains and entity aliasing that recover supposedly forgotten facts through indirect queries (Shah et al., 2026) - sits on the same input-side axis; whether RExU's route robustness transfers to such probes is untested.

B NOTATION AND ABBREVIATIONS

Tables 3 and 4 summarize the main notation and abbreviations used throughout this paper.

Table 3: **Summary of main notation.**

Notation	Meaning
Model and data	
$\theta_0, \theta_u, \theta_t$	Pretrained, unlearned, and round-t model parameters.
$f_\theta(q; r)$	Model output for input q under route r.
D_f, D_r	Forget and retain datasets.
$\mathcal{D}_f^{\text{cal}}, \mathcal{D}_{\text{test}}$	Forget-side calibration data and held-out test data.
$q, y, \mathcal{A}, K$	Input, correct answer, answer-option set, and its size.
$\mathcal{L}, (l, e), E_l, k$	MoE layers, layer–expert pair, experts per layer, and experts selected per token.
z_l, g_l	Router logits and pre-top-k softmax probabilities.
Routing and evaluation	
r, r_{nat}, T_l^r	Route intervention, natural route, and layer-wise logit transform.
$\mathcal{R}, \mathcal{R}_{\text{train}}$	Reporting route family and subset used during unlearning.
$A1_S, A2_\sigma, A2_\beta, A3_\gamma$	Expert masking, Gaussian gate noise, logit re-uniformization, and rare-expert forcing.
S, S_t	Masking reference set and cumulative edited set at round t.
$\hat{\pi}_{l,e}$	Natural-route selection frequency of expert (l, e).
N_s	Number of fixed seeds for stochastic route auditing.
$m_\theta(q, y; r)$	Correct-answer log-probability minus the largest incorrect-answer log-probability.
$\text{FE}(\theta; r)$	Route-specific forget accuracy (WMDP) or ROUGE-L recall (RWKU); lower is better.
$\text{WRFE}_{\mathcal{R}}(\theta)$	Maximum FE over the audited route family $\mathcal{R}$.
$\Delta_{\mathcal{R}}(\theta)$	Recovery gap: $\text{WRFE}_{\mathcal{R}}(\theta) - \text{FE}(\theta; r_{\text{nat}})$.
$\text{UT}(\theta)$	Natural-route retain utility, measured by MMLU accuracy.
$p_{\text{chance}}, \tau, u_0$	Recovery reference floor, forgetting tolerance, and minimum retain utility.
Recovery-Expert Unlearning	
$r_t^\star$	Highest-recovery route selected at round t.
$A_t(l, e), \mathcal{P}_t, N_c$	Counterfactual routing affinity, candidate expert pool, and pool size.
$\theta_t^{-(l,e)}$	Counterfactual evaluation replacing expert (l, e)'s output by its retain-set mean activation.
$C_t(l, e)$	Signed decrease in the mean recovery signal when expert (l, e) is ablated.
$\mathcal{E}_t, B, T$	Selected experts, per-round expert budget, and maximum number of rounds.
$\mathcal{U}$	Base unlearning operator applied to selected experts.
$L_{\text{forget}}, L_{\text{retain}}, \lambda$	Forget loss, retain regularization, and retain-loss weight.
$\delta, \theta_{\text{ref}}$	Margin offset and reference checkpoint for retain regularization.
θ_{best}	Best selected checkpoint satisfying the utility floor.
Structural analysis	
$P_{\theta_0}(q, r), \theta_S$	Reference expert support and checkpoint edited only within S.
G_k, U_k	Specialized expert group and its surviving unedited experts.
$\mathcal{U}_\gamma(S, r)$	Recovery witnesses whose expert supports avoid S.
$\kappa_{\gamma,\beta}$	Minimum edit budget satisfying the necessary coverage condition at recovery level γ and target WRFE β.

Table 4: **Abbreviations used throughout the paper.**

Abbreviation	Full name
Core concepts and metrics	
MoE	Mixture-of-Experts.
FE	Forget Efficacy.
WRFE (**ours**)	Worst-Route Forget Efficacy.
UT	Retain Utility.
RExU (**ours**)	Recovery-Expert Unlearning.
Unlearning methods	
GA	Gradient Ascent.
GDIFF	Gradient Difference.
NPO	Negative Preference Optimization.
RMU	Representation Misdirection for Unlearning.
SEUF	SEUF (Zhuang et al., 2025); localized MoE unlearning based on high-affinity expert selection.
SEUF-M	Multi-expert extension of SEUF using the top-M affinity-ranked experts.
Benchmarks and evaluation	
WMDP	Weapons of Mass Destruction Proxy benchmark; we use the WMDP-Cyber subset. (Li et al., 2024)
RWKU	Real-World Knowledge Unlearning benchmark. (Cao et al., 2024)
MMLU	Massive Multitask Language Understanding benchmark. (Hendrycks et al., 2020)

C THEORETICAL ANALYSIS

Definition 4 (Computation-path intervention - general)**.** *Let $f_\theta(x;\,\pi)$ denote a model whose forward pass is parameterized by a* computation path $\pi \in \Pi$*, where Π is the set of paths reachable by architecture-level interventions at inference time without modifying θ or x. A* path intervention r *is a mapping $\pi_r : \mathcal{X} \to \Pi$ that selects, for each input x, an alternative path $\pi_r(x) \in \Pi$.*

C.1 MARGIN AND FORGET ACCURACY

For a multiple-choice forget example q with options $\mathcal{A}$ and correct answer y, let $s_\theta(a; q, r) = \log p_\theta(a \mid q, \mathcal{A}; r)$ and

$$m_\theta(q; r) = s_\theta(y; q, r) - \max_{a \in \mathcal{A},\, a \neq y} s_\theta(a; q, r). \tag{15}$$

Breaking ties against y, the model is correct iff $m_\theta(q; r) > 0$, so $\mathrm{FE}(\theta; r) = \frac{1}{|D_f|} \sum_q \mathbf{1}\{m_\theta(q; r) > 0\}$.

C.2 WORST-ROUTE FORGETTING

For finite $\mathcal{R} \ni r_{\mathrm{nat}}$, let $\mathrm{WRFE}_{\mathcal{R}}(\theta) = \max_{r \in \mathcal{R}} \mathrm{FE}(\theta; r)$, and let $\Delta_{\mathcal{R}}(\theta) = \mathrm{WRFE}_{\mathcal{R}}(\theta) - \mathrm{FE}(\theta; r_{\mathrm{nat}})$ denote the worst-route gap.

Lemma 1 (Basic properties of worst-route forgetting)**.** *For any θ and any finite route family $\mathcal{R} \ni r_{\mathrm{nat}}$:*

- *(a)* ***Natural-route evaluation is optimistic.*** *$\Delta_{\mathcal{R}}(\theta) \geq 0$, with equality iff r_{nat} attains the maximum defining $\mathrm{WRFE}_{\mathcal{R}}(\theta)$.*
- *(b)* ***Masking certifies off-set recovery.*** *For any reference expert set S with at most $E_l - k$ members in every layer l (so that top-k selection remains feasible after masking) and its masking intervention $A1_S$: if $\mathrm{FE}(\theta; A1_S) > p_{\mathrm{chance}}$, the model recovers above chance under a route that activates no expert in S.*
- *(c)* ***Monotonicity in the route family.*** *$\mathcal{R} \subseteq \mathcal{R}' \Rightarrow \mathrm{WRFE}_{\mathcal{R}}(\theta) \leq \mathrm{WRFE}_{\mathcal{R}'}(\theta)$.*

Remark 2 (Finite-sample certification)**.** *Part (b) of Lemma 1 is a population-level statement. In practice $\mathrm{FE}(\theta; A1_S)$ is estimated from n held-out forget-set items, $\widehat{\mathrm{FE}} = \hat{k}/n$; at our scale ($n = 1{,}987$) sampling noise alone gives a 95% confidence half-width of roughly 2 accuracy points (Wilson score interval), so $\widehat{\mathrm{FE}} > 1/K$ is not by itself sufficient evidence. We certify a masking result only when the one-sided 95% Wilson lower bound on $\widehat{\mathrm{FE}}(\theta; A1_S)$ exceeds $1/K$; every masking certificate reported in this paper (Table 1, Finding 1) meets this bound. Two further points. The certificate concerns a single, pre-specified route ($A1_S$), so no multiplicity correction is needed; a statement that $\mathrm{WRFE} > 1/K$, being a maximum over $|\mathcal{R}|$ routes, would require one, and we make no such statement. And $1/K$ is the correct floor only if the model has no knowledge-free advantage from answer-format priors; the degraded full-model checkpoints GA (24.31) and GDIFF (25.16, Table 7) sit at chance, direct evidence that the format floor on WMDP-Cyber is at $1/K$.*

Remark 3 (Shared experts do not toggle with the route, but their contribution can still shift)**.** *In MoE architectures with always-on shared experts, let S^e denote the shared expert set activated for every token regardless of routing. Under any route intervention r that acts only on router logits (Definition 1), the shared experts remain in the active set identically across r_{nat} and any $r \in \mathcal{R}$: no route intervention turns a shared expert on or off, since S^e is never gated. This is weaker than saying their* contribution *to* FE *is unchanged: a shared expert at layer l receives the hidden state produced by layer $l-1$, which depends on which routed experts were active there, so its output can still shift across routes even though it is always included in the combination. What route intervention alone cannot do is add or remove a shared expert from the computation; any change in $\Delta_{\mathcal{R}}(\theta)$ therefore requires* some *routed expert's activation to differ across routes - shared experts cannot be the sole explanation for $\Delta_{\mathcal{R}} > 0$ acting alone, but they are not guaranteed to contribute identically either, since they inherit whatever the routed computation upstream changed. A direct probe - masking or ablating the shared experts on a route-robust checkpoint and re-measuring FE - would test whether residual worst-route recovery after RExU (Table 5, 27-33 vs. chance 25) sits partly in shared-expert*

computation that RExU never edits; we have not run this probe and flag it as a natural next step rather than a settled question.

C.3 Non-Identifiability and Over-Specification in Softmax-Gated Mixtures

Before studying how experts specialize during training, we first ask a more fundamental question: does the function implemented by a softmax-gated mixture uniquely determine its expert-level representation? In general, the answer is no. This ambiguity is stronger than the familiar fact that expert labels can be permuted. An over-specified softmax mixture can represent the same input–output function while distributing the contribution of one functional component across several experts. Consequently, a behavior that appears as one functional mode need not correspond to a unique expert, even when the overall true function is known exactly. We make this statement precise below.

Softmax-gated mixture. Consider a mixture with M experts. Expert j implements a function $g_j : \mathcal{X} \to \mathbb{R}^d$, while its routing score on input $x \in \mathcal{X}$ is denoted by $s_j(x) \in \mathbb{R}$. The corresponding softmax routing weight is

$$\pi_j(x) := \frac{\exp(s_j(x))}{\sum_{k=1}^{M} \exp(s_k(x))}. \tag{16}$$

The output of the mixture is therefore

$$F(x) := \sum_{j=1}^{M} \pi_j(x) g_j(x). \tag{17}$$

At this stage, F should be understood as the function implemented by one softmax-gated mixture block. If two parameterizations produce the same mixture output for every input, then replacing one by the other leaves all downstream deterministic computation unchanged as well. Thus the same argument applies locally inside a larger MoE network.

Permutation symmetry is only the first ambiguity. The most immediate source of non-identifiability is expert permutation. For any permutation σ of $\{1, \dots, M\}$, define

$$g'_j := g_{\sigma(j)} \qquad \text{and} \qquad s'_j := s_{\sigma(j)}.$$

Then the corresponding routing weights satisfy $\pi'_j(x) = \pi_{\sigma(j)}(x)$, and hence

$$\begin{aligned} F'(x) &= \sum_{j=1}^{M} \pi'_j(x) g'_j(x) \\ &= \sum_{j=1}^{M} \pi_{\sigma(j)}(x) g_{\sigma(j)}(x) \\ &= \sum_{k=1}^{M} \pi_k(x) g_k(x) \\ &= F(x). \end{aligned} \tag{18}$$

Thus expert identity is already identifiable, at best, only up to permutation. For unlearning, however, permutation symmetry is not the important ambiguity. Relabeling experts does not change how many distinct expert blocks support a behavior. A more consequential ambiguity appears when the mixture contains more experts than are functionally necessary.

Over-specification. We call a representation of F *over-specified* if the same function can also be represented using strictly fewer expert components. Equivalently, if $M_\star$ denotes the smallest number of experts required to represent F within the mixture function class, then a representation using $M > M_\star$ experts is over-specified. Over-specification raises a different question from permutation symmetry. Suppose one expert contributes a particular function to the mixture. Can that contribution be split across several experts without changing F at all?

Theorem 1 (Exact Expert-Splitting Non-Identifiability)**.** *Consider the softmax-gated mixture in Equation 17, and fix an expert $j^\star \in \{1, \dots, M\}$. Pick an integer $K \geq 2$ and positive coefficients $\lambda_1, \dots, \lambda_K$ satisfying $\sum_{a=1}^{K} \lambda_a = 1$. Replace expert $j^\star$ by K experts where,*

$$g_{j^\star,a}(x) := g_{j^\star}(x), \qquad a = 1, \dots, K, \tag{19}$$

and assign them routing scores $s_{j^\star,a}(x) := s_{j^\star}(x) + \log \lambda_a$. All remaining experts and routing scores are left unchanged. Then the resulting $(M + K - 1)$-expert softmax mixture implements exactly the same function as the original mixture $F_{\text{split}}(x) = F(x)$ for every input $x \in \mathcal{X}$. Moreover, the original routing mass of expert $j^\star$ is divided among the K copies according to

$$\pi^{\text{split}}_{j^\star,a}(x) = \lambda_a \pi_{j^\star}(x). \tag{20}$$

Proof. Fix an arbitrary input $x \in \mathcal{X}$. Denote the original softmax normalization by $Z(x) := \sum_{j=1}^{M} \exp(s_j(x))$. After splitting expert $j^\star$, the new normalization becomes

$$\begin{aligned} Z_{\text{split}}(x) &= \sum_{j \neq j^\star} \exp(s_j(x)) + \sum_{a=1}^{K} \exp(s_{j^\star}(x) + \log \lambda_a) \\ &= \sum_{j \neq j^\star} \exp(s_j(x)) + \sum_{a=1}^{K} \lambda_a \exp(s_{j^\star}(x)) \\ &= \sum_{j \neq j^\star} \exp(s_j(x)) + \exp(s_{j^\star}(x)) \sum_{a=1}^{K} \lambda_a \\ &= \sum_{j \neq j^\star} \exp(s_j(x)) + \exp(s_{j^\star}(x)) \\ &= Z(x), \end{aligned} \tag{21}$$

where the fourth equality uses $\sum_{a=1}^{K} \lambda_a = 1$. Because the normalization is unchanged, every unsplit expert retains exactly the same routing weight:

$$\pi^{\text{split}}_{j}(x) = \frac{\exp(s_j(x))}{Z_{\text{split}}(x)} = \frac{\exp(s_j(x))}{Z(x)} = \pi_j(x), \qquad j \neq j^\star. \tag{22}$$

For the a-th copy of expert $j^\star$, we instead obtain

$$\begin{aligned} \pi^{\text{split}}_{j^\star,a}(x) &= \frac{\exp(s_{j^\star}(x) + \log \lambda_a)}{Z_{\text{split}}(x)} \\ &= \frac{\lambda_a \exp(s_{j^\star}(x))}{Z(x)} \\ &= \lambda_a \pi_{j^\star}(x). \end{aligned} \tag{23}$$

This proves Equation 20. It remains to verify that the mixture output is unchanged. Using Equation 19, Equation 22, and Equation 23,

$$\begin{aligned} F_{\text{split}}(x) &= \sum_{j \neq j^\star} \pi^{\text{split}}_{j}(x) g_j(x) + \sum_{a=1}^{K} \pi^{\text{split}}_{j^\star,a}(x) g_{j^\star,a}(x) \\ &= \sum_{j \neq j^\star} \pi_j(x) g_j(x) + \sum_{a=1}^{K} \lambda_a \pi_{j^\star}(x) g_{j^\star}(x) \\ &= \sum_{j \neq j^\star} \pi_j(x) g_j(x) + \pi_{j^\star}(x) g_{j^\star}(x) \sum_{a=1}^{K} \lambda_a \\ &= \sum_{j=1}^{M} \pi_j(x) g_j(x) \\ &= F(x). \end{aligned} \tag{24}$$

Since x was arbitrary, the equality holds for every $x \in \mathcal{X}$. □

What the splitting theorem implies. Theorem 1 gives a stronger form of non-identifiability than label permutation. The functional contribution $\pi_{j^\star}(x)g_{j^\star}(x)$ can be represented equivalently as $\sum_{a=1}^{K} \lambda_a \pi_{j^\star}(x)g_{j^\star}(x)$ for any point $(\lambda_1, \ldots, \lambda_K)$ in the interior of the $(K-1)$-dimensional probability simplex. Hence the ambiguity is not a finite collection of label permutations. There is a continuum of expert-level decompositions that implement exactly the same input–output function. This distinction is central to the later unlearning analysis. If $g_{j^\star}$ represents a functional behavior associated with some underlying data mode, the model function alone does not determine whether that behavior is represented by one expert carrying routing mass $\pi_{j^\star}$ or by several experts whose routing masses sum to the same quantity. Functional equivalence therefore does not imply a unique expert-level location of the behavior.

Why this matters for expert specialization. Theorem 1 establishes that redundant expert representations are possible without changing the implemented mixture function. It does not, however, imply that training will produce several experts specialized to the same mode. Non-identifiability characterizes the space of functionally equivalent representations, whereas specialization depends on how routing preferences evolve during optimization. The next subsection therefore studies the dynamics of the softmax gate. Because routing probabilities depend on relative scores, persistent differences in score updates accumulate as changes in routing odds. We use this mechanism to characterize when a group of experts can specialize toward the same underlying data mode without collapsing to a single dominant expert.

C.4 Softmax Competition and Redundant Expert Specialization

The previous subsection showed that an over-specified softmax mixture can represent the same functional component using several experts. This establishes that redundant expert representations are possible, but it does not explain why training would associate different experts with different regions of the data. We now study the routing mechanism that creates this specialization. The key property of softmax gating is competition: experts are compared through relative, rather than independent, routing scores. A persistent advantage for one expert or one expert group therefore accumulates as a growing routing preference. We first derive this mechanism exactly and then show how it can produce specialization without necessarily eliminating redundancy within the winning group.

Task-induced routing advantage. Consider the softmax-gated mixture $F(x) = \sum_{j=1}^{M} \pi_j(x)g_j(x)$ with $\pi_j(x) = \exp(s_j(x))/\sum_{k=1}^{M}\exp(s_k(x))$. For a training pair (x, y), let $\ell(F(x), y)$ be a differentiable task loss and write $v(x, y) := \nabla_F \ell(F(x), y)$.

Differentiating the softmax mixture with respect to the routing score $s_j(x)$ gives

$$\begin{aligned}\frac{\partial F(x)}{\partial s_j(x)} &= \sum_{i=1}^{M} \frac{\partial \pi_i(x)}{\partial s_j(x)} g_i(x) \\ &= \sum_{i=1}^{M} \pi_i(x)\big(\mathbf{1}\{i=j\} - \pi_j(x)\big)g_i(x) \\ &= \pi_j(x)\big(g_j(x) - F(x)\big).\end{aligned} \tag{25}$$

Applying the chain rule,

$$\frac{\partial \ell(F(x), y)}{\partial s_j(x)} = \pi_j(x)\,\langle v(x, y), g_j(x) - F(x)\rangle\,. \tag{26}$$

It is therefore natural to define the routing advantage of expert j as $A_j(x, y) := -\langle v(x, y), g_j(x) - F(x)\rangle$. A positive A_j means that increasing the routing score of expert j locally moves the mixture output in a direction that decreases the task loss.

The exact parameter update of a practical router depends on its parameterization and on gradients from other training examples. For this reason, rather than assuming that the routing scores

themselves are free optimization variables, we write $\delta_j^{(t)}(x) := s_j^{(t+1)}(x) - s_j^{(t)}(x)$ for the actual score increment induced by the training update. Under direct gradient descent in score space, $\delta_j^{(t)}(x) = \eta_t \pi_j^{(t)}(x) A_j^{(t)}(x, y)$, but the results below only require conditions on the induced relative score increments.

Lemma 2 (Softmax Log-Odds Accumulation). *For any two experts i and j and any iteration $T \geq 1$,*

$$\log \frac{\pi_i^{(T)}(x)}{\pi_j^{(T)}(x)} = \log \frac{\pi_i^{(0)}(x)}{\pi_j^{(0)}(x)} + \sum_{t=0}^{T-1} \left(\delta_i^{(t)}(x) - \delta_j^{(t)}(x) \right). \tag{27}$$

Proof. Softmax normalization cancels in the ratio, so at every iteration $\pi_i^{(t)}(x)/\pi_j^{(t)}(x) = \exp(s_i^{(t)}(x) - s_j^{(t)}(x))$. Taking logarithms at iterations t and $t+1$ gives

$$\begin{aligned} \log \frac{\pi_i^{(t+1)}(x)}{\pi_j^{(t+1)}(x)} - \log \frac{\pi_i^{(t)}(x)}{\pi_j^{(t)}(x)} &= \left(s_i^{(t+1)} - s_i^{(t)} \right) - \left(s_j^{(t+1)} - s_j^{(t)} \right) \\ &= \delta_i^{(t)}(x) - \delta_j^{(t)}(x). \end{aligned} \tag{28}$$

Summing over $t = 0, \ldots, T-1$ yields Equation 27. □

Lemma 2 isolates the competitive mechanism. Softmax does not amplify the absolute score of an expert; it amplifies accumulated *differences* in routing scores. A persistent relative advantage therefore translates directly into increasing routing odds.

Competition between an expert group and the remaining mixture. Consider a latent data mode k, and let $\mathcal{X}_k$ denote the inputs belonging to that mode. We allow several experts to represent the same mode and denote their index set by $G_k \subseteq \{1, \ldots, M\}$. For an input $x \in \mathcal{X}_k$, the total routing mass assigned to this group is $\Pi_{G_k}^{(T)}(x) := \sum_{i \in G_k} \pi_i^{(T)}(x)$. The next lemma shows that a persistent score advantage of the group over all outside experts forces the outside routing mass to disappear.

Lemma 3 (Competitive Amplification of a Mode-Specialized Group). *Fix $x \in \mathcal{X}_k$. Suppose there exists $\Delta_k > 0$ and nonnegative step weights η_t such that, for every t, every $i \in G_k$, and every $j \notin G_k$,*

$$\delta_i^{(t)}(x) - \delta_j^{(t)}(x) \geq \eta_t \Delta_k.$$

Define $H_T := \sum_{t=0}^{T-1} \eta_t$. Then, for every $i \in G_k$ and $j \notin G_k$,

$$\frac{\pi_j^{(T)}(x)}{\pi_i^{(T)}(x)} \leq \frac{\pi_j^{(0)}(x)}{\pi_i^{(0)}(x)} e^{-\Delta_k H_T}.$$

Consequently, for any fixed $i \in G_k$,

$$\Pi_{G_k}^{(T)}(x) \geq \frac{1}{1 + C_{i,k}(x) e^{-\Delta_k H_T}}, \qquad C_{i,k}(x) := \sum_{j \notin G_k} \frac{\pi_j^{(0)}(x)}{\pi_i^{(0)}(x)}. \tag{29}$$

Proof. By Lemma 2,

$$\begin{aligned} \log \frac{\pi_i^{(T)}(x)}{\pi_j^{(T)}(x)} &= \log \frac{\pi_i^{(0)}(x)}{\pi_j^{(0)}(x)} + \sum_{t=0}^{T-1} \left(\delta_i^{(t)}(x) - \delta_j^{(t)}(x) \right) \\ &\geq \log \frac{\pi_i^{(0)}(x)}{\pi_j^{(0)}(x)} + \Delta_k H_T. \end{aligned} \tag{30}$$

Exponentiating and taking reciprocals gives the pairwise ratio bound. To control the total routing mass outside G_k, we write

$$\begin{aligned} 1 - \Pi_{G_k}^{(T)}(x) &= \sum_{j \notin G_k} \pi_j^{(T)}(x) \\ &= \pi_i^{(T)}(x) \sum_{j \notin G_k} \frac{\pi_j^{(T)}(x)}{\pi_i^{(T)}(x)} \\ &\le \pi_i^{(T)}(x) C_{i,k}(x) e^{-\Delta_k H_T} \\ &\le \Pi_{G_k}^{(T)}(x) C_{i,k}(x) e^{-\Delta_k H_T}. \end{aligned} \tag{31}$$

Rearranging yields Equation 29. □

If $H_T \to \infty$, the exponential term in Equation 29 vanishes, and therefore $\Pi_{G_k}^{(T)}(x) \to 1$. Thus a persistent relative score advantage is sufficient for the expert group G_k to capture asymptotically all routing mass on mode-k inputs. This result explains specialization across modes, but it does not yet determine what happens *inside* the specialized group. A second condition is needed to distinguish redundant specialization from winner-take-all collapse.

Competition within the specialized group. Suppose that the accumulated relative score drift between experts in G_k remains bounded. Specifically, assume that there exists $B_k < \infty$ such that

$$\left| \sum_{t=0}^{T-1} \left(\delta_i^{(t)}(x) - \delta_\ell^{(t)}(x) \right) \right| \le B_k$$

for every T and every $i, \ell \in G_k$. Lemma 2 then gives

$$e^{-B_k} \frac{\pi_\ell^{(0)}(x)}{\pi_i^{(0)}(x)} \le \frac{\pi_\ell^{(T)}(x)}{\pi_i^{(T)}(x)} \le e^{B_k} \frac{\pi_\ell^{(0)}(x)}{\pi_i^{(0)}(x)}.$$

Hence no expert inside G_k can asymptotically dominate another by an unbounded routing ratio. We can now state the main specialization result compactly.

Theorem 2 (Redundant Expert Specialization)**.** *Fix a mode k and an expert group G_k with $|G_k| \ge 2$. Suppose that*

1. *G_k has a persistent positive score-increment gap over every expert outside the group, as in Lemma 3; and*
2. *the accumulated score-increment difference between any two experts inside G_k is uniformly bounded by some $B_k < \infty$.*

If $H_T \to \infty$, then the group specializes to mode k in the sense that $\Pi_{G_k}^{(T)}(x) \to 1$ for every input for which the two conditions hold. Moreover, every expert $i \in G_k$ retains a nonvanishing fraction of the group routing mass. In particular,

$$\liminf_{T \to \infty} \pi_i^{(T)}(x) \ge \frac{1}{1 + e^{B_k} \displaystyle\sum_{\ell \in G_k \setminus \{i\}} \frac{\pi_\ell^{(0)}(x)}{\pi_i^{(0)}(x)}} > 0. \tag{32}$$

Proof. Fix an input $x \in \mathcal{X}_k$ and an expert $i \in G_k$. By the first assumption of the theorem, the experts in G_k maintain a persistent positive score-increment advantage over every expert outside the group. Hence Lemma 3 gives

$$\Pi_{G_k}^{(T)}(x) \ge \frac{1}{1 + C_{i,k}(x) e^{-\Delta_k H_T}}, \tag{33}$$

where $C_{i,k}(x) = \sum_{j \notin G_k} \pi_j^{(0)}(x) / \pi_i^{(0)}(x)$. Because all softmax probabilities are strictly positive, $C_{i,k}(x)$ is finite. Since $\Delta_k > 0$ and $H_T \to \infty$, the exponential term satisfies $e^{-\Delta_k H_T} \to 0$.

Therefore, $\liminf_{T\to\infty} \Pi_{G_k}^{(T)}(x) \geq 1$. On the other hand, $\Pi_{G_k}^{(T)}(x)$ is a sum of routing probabilities and thus cannot exceed one. Hence $\limsup_{T\to\infty} \Pi_{G_k}^{(T)}(x) \leq 1$, which together with the previous inequality yields $\Pi_{G_k}^{(T)}(x) \to 1$.

It remains to determine whether this concentration of routing mass onto G_k necessarily collapses to a single expert. Consider any $\ell \in G_k \setminus \{i\}$. By Lemma 2, their routing ratio satisfies

$$\log \frac{\pi_\ell^{(T)}(x)}{\pi_i^{(T)}(x)} = \log \frac{\pi_\ell^{(0)}(x)}{\pi_i^{(0)}(x)} + \sum_{t=0}^{T-1} \left(\delta_\ell^{(t)}(x) - \delta_i^{(t)}(x)\right). \tag{34}$$

The bounded internal-drift assumption gives

$$\left|\sum_{t=0}^{T-1} \left(\delta_\ell^{(t)}(x) - \delta_i^{(t)}(x)\right)\right| \leq B_k,$$

so in particular

$$\sum_{t=0}^{T-1} \left(\delta_\ell^{(t)}(x) - \delta_i^{(t)}(x)\right) \leq B_k.$$

Substituting this inequality into Equation 34 gives

$$\log \frac{\pi_\ell^{(T)}(x)}{\pi_i^{(T)}(x)} \leq \log \frac{\pi_\ell^{(0)}(x)}{\pi_i^{(0)}(x)} + B_k. \tag{35}$$

Exponentiating both sides,

$$\frac{\pi_\ell^{(T)}(x)}{\pi_i^{(T)}(x)} \leq e^{B_k} \frac{\pi_\ell^{(0)}(x)}{\pi_i^{(0)}(x)}. \tag{36}$$

Thus the routing mass of another expert in G_k can exceed that of i only by a multiplicative factor that is independent of T. In particular, specialization of the group does not by itself permit an unbounded separation among experts inside the group. We now express the total routing mass of G_k relative to expert i:

$$\begin{aligned} \Pi_{G_k}^{(T)}(x) &= \sum_{\ell \in G_k} \pi_\ell^{(T)}(x) \\ &= \pi_i^{(T)}(x) + \sum_{\ell \in G_k \setminus \{i\}} \pi_\ell^{(T)}(x) \\ &= \pi_i^{(T)}(x) \left[1 + \sum_{\ell \in G_k \setminus \{i\}} \frac{\pi_\ell^{(T)}(x)}{\pi_i^{(T)}(x)}\right]. \end{aligned} \tag{37}$$

Applying Equation 36 to each ratio in the sum yields

$$\Pi_{G_k}^{(T)}(x) \leq \pi_i^{(T)}(x) \left[1 + e^{B_k} \sum_{\ell \in G_k \setminus \{i\}} \frac{\pi_\ell^{(0)}(x)}{\pi_i^{(0)}(x)}\right]. \tag{38}$$

Rearranging,

$$\pi_i^{(T)}(x) \geq \frac{\Pi_{G_k}^{(T)}(x)}{1 + e^{B_k} \sum_{\ell \in G_k \setminus \{i\}} \frac{\pi_\ell^{(0)}(x)}{\pi_i^{(0)}(x)}}. \tag{39}$$

The denominator in Equation 39 is finite and independent of T. Since $\Pi_{G_k}^{(T)}(x) \to 1$, taking the lower limit gives

$$\liminf_{T\to\infty} \pi_i^{(T)}(x) \geq \frac{1}{1 + e^{B_k} \sum_{\ell \in G_k \setminus \{i\}} \frac{\pi_\ell^{(0)}(x)}{\pi_i^{(0)}(x)}} > 0. \tag{40}$$

Because $i \in G_k$ was arbitrary, every expert in the group retains nonvanishing routing mass even though the total mass of the group converges to one. Hence the group specializes to mode k without collapsing to a single expert, which proves the claimed redundant-specialization behavior. □

Interpretation. The theorem separates two effects that should not be conflated. The *between-group* score gap determines whether a set of experts specializes toward a mode. Because softmax routing depends exponentially on score differences, a persistent linear growth in relative score produces exponential suppression of competing experts. The *within-group* score differences answer a different question: whether specialization collapses onto a single expert or remains redundant. If these differences accumulate without bound, one expert may eventually dominate. If they remain bounded, several experts retain nonvanishing routing mass even after the group as a whole has specialized.

This distinction connects directly to the non-identifiability result of the previous subsection. Expert splitting shows that one functional mode may admit several equivalent expert representations. The present theorem shows that softmax competition can separate such a group from experts associated with other modes without necessarily resolving the redundancy inside the group. Thus, specialization of a functional mode does not imply identification of a unique expert.

C.5 Redundant Mode Preservation under Partial Expert Editing

The previous subsection showed that several experts can jointly specialize toward the same data mode while retaining nonvanishing routing mass. Redundant specialization, however, becomes relevant to unlearning only if the remaining experts can still reproduce the specialized computation after some members of the group have been modified. We now show that this is precisely the case when the specialized experts approximate the same underlying mode. The argument is simple but consequential: a convex combination of several accurate approximations to the same function remains an accurate approximation even after an arbitrary subset of those experts is removed and the remaining routing weights are renormalized.

Redundant approximation of a common mode. Consider a mode k and the corresponding specialized expert group G_k introduced in the previous subsection. We write $g_k^\star(x)$ for the underlying mode-specific representation on an input $x \in \mathcal{X}_k$. The function $g_k^\star$ should be interpreted as the common functional behavior that the experts in G_k have learned to approximate; it need not correspond to the parameters of any particular fitted expert. Suppose that the experts in G_k approximate this mode uniformly to accuracy $\varepsilon_k \geq 0$, so that

$$\|g_i(x) - g_k^\star(x)\| \leq \varepsilon_k, \qquad i \in G_k,$$

where $\|\cdot\|$ denotes a fixed norm on the expert output space. Now consider an unlearning intervention that modifies a subset $S \subsetneq G_k$. The experts that remain unmodified are $U_k := G_k \setminus S$, which is nonempty because the edit is only partial.

A counterfactual route may redistribute routing mass over these surviving experts. Denote the resulting normalized routing weights by $\alpha_i(x; r) \geq 0$ for $i \in U_k$, where $\sum_{i \in U_k} \alpha_i(x; r) = 1$. The representation produced by the surviving experts under route r is therefore

$$g_{U_k}(x; r) := \sum_{i \in U_k} \alpha_i(x; r) g_i(x).$$

The following lemma shows that removing experts from a redundantly specialized group does not increase the approximation error as long as the route remains a convex combination of the surviving specialists.

Lemma 4 (Convex Stability of Redundant Specialists)**.** *Suppose $\|g_i(x) - g_k^\star(x)\| \leq \varepsilon_k$ for every $i \in U_k$. Then, for any nonnegative routing weights $\{\alpha_i(x; r)\}_{i \in U_k}$ satisfying $\sum_{i \in U_k} \alpha_i(x; r) = 1$,*

$$\|g_{U_k}(x; r) - g_k^\star(x)\| \leq \varepsilon_k.$$

Proof. Using the definition of $g_{U_k}(x; r)$ and the fact that the routing weights sum to one, we can write

$$\begin{aligned} g_{U_k}(x; r) - g_k^\star(x) &= \sum_{i \in U_k} \alpha_i(x; r) g_i(x) - g_k^\star(x) \sum_{i \in U_k} \alpha_i(x; r) \\ &= \sum_{i \in U_k} \alpha_i(x; r) \big(g_i(x) - g_k^\star(x)\big). \end{aligned} \tag{41}$$

Applying the triangle inequality and using $\alpha_i(x;r) \geq 0$ gives

$$\begin{aligned} \|g_{U_k}(x;r) - g_k^\star(x)\| &\leq \sum_{i \in U_k} \alpha_i(x;r) \|g_i(x) - g_k^\star(x)\| \\ &\leq \varepsilon_k \sum_{i \in U_k} \alpha_i(x;r) \\ &= \varepsilon_k. \end{aligned} \tag{42}$$

Thus every convex combination of the surviving specialists remains within the same approximation radius ε_k of the underlying mode. □

Lemma 4 already reveals an important asymmetry between ordinary parameter editing and functional removal. An edit may substantially alter some members of G_k, yet the unedited members remain inside the same functional neighborhood of $g_k^\star$. Renormalizing their routing weights therefore reconstructs the mode without requiring the edited experts at all. To determine whether this functional approximation is sufficient to recover the target behavior, we next propagate the representation error through the downstream computation.

Recovery margin after rerouting. For a fixed input x, let $R_x(h) \in \mathbb{R}^{|\mathcal{Y}|}$ denote the vector of answer scores produced by the remaining network when the output of the considered MoE block is set to h. All downstream parameters are held fixed in this definition. If $y^\star$ is the target answer associated with x, define its downstream margin as

$$m_x(h) := R_{x,y^\star}(h) - \max_{y \neq y^\star} R_{x,y}(h).$$

A positive value of $m_x(h)$ therefore means that the target answer is strictly preferred to every competing answer. We assume that the margin is locally Lipschitz around the mode representation: there exists $L_k \geq 0$ such that

$$|m_x(h) - m_x(h')| \leq L_k \|h - h'\|$$

for the representations considered below. This condition controls how much a small change in the specialized representation can alter the final target margin. For example, if every downstream answer score is L-Lipschitz in h, then the margin is $2L$-Lipschitz. We can now state the main result.

Theorem 3 (Redundant Mode Preservation under Partial Expert Editing)**.** *Fix a mode-k input $x \in \mathcal{X}_k$ whose underlying mode representation has positive target margin $m_x(g_k^\star(x)) \geq \gamma_k$ for some $\gamma_k > 0$. Suppose the specialized experts satisfy $\|g_i(x) - g_k^\star(x)\| \leq \varepsilon_k$ for every $i \in G_k$. Consider any partial edit $S \subsetneq G_k$ that leaves the surviving set $U_k = G_k \setminus S$ nonempty, and suppose an admissible counterfactual route r forms a normalized convex combination of these surviving experts. Then the target margin after rerouting satisfies*

$$m_x(g_{U_k}(x;r)) \geq \gamma_k - L_k \varepsilon_k. \tag{43}$$

Consequently, if $\gamma_k > L_k \varepsilon_k$, the target remains recoverable under the counterfactual route despite the partial expert edit.

Proof. Because the edit is partial, the surviving set $U_k = G_k \setminus S$ is nonempty. By assumption, every surviving expert remains within ε_k of the common mode representation:

$$\|g_i(x) - g_k^\star(x)\| \leq \varepsilon_k, \qquad \forall i \in U_k.$$

The counterfactual route r assigns nonnegative normalized weights $\alpha_i(x;r)$ to the surviving experts, with $\sum_{i \in U_k} \alpha_i(x;r) = 1$. Therefore,

$$\begin{aligned} g_{U_k}(x;r) - g_k^\star(x) &= \sum_{i \in U_k} \alpha_i(x;r) g_i(x) - g_k^\star(x) \\ &= \sum_{i \in U_k} \alpha_i(x;r) g_i(x) - \left(\sum_{i \in U_k} \alpha_i(x;r) \right) g_k^\star(x) \\ &= \sum_{i \in U_k} \alpha_i(x;r) \left(g_i(x) - g_k^\star(x) \right). \end{aligned} \tag{44}$$

Taking norms on both sides and applying the triangle inequality gives

$$\begin{aligned}\|g_{U_k}(x;r) - g_k^\star(x)\| &= \left\| \sum_{i\in U_k} \alpha_i(x;r)\,(g_i(x) - g_k^\star(x)) \right\| \\ &\le \sum_{i\in U_k} \|\alpha_i(x;r)\,(g_i(x) - g_k^\star(x))\| \\ &= \sum_{i\in U_k} \alpha_i(x;r)\,\|g_i(x) - g_k^\star(x)\| \\ &\le \sum_{i\in U_k} \alpha_i(x;r)\varepsilon_k \\ &= \varepsilon_k \sum_{i\in U_k} \alpha_i(x;r) \\ &= \varepsilon_k.\end{aligned} \tag{45}$$

The third line uses $\alpha_i(x;r) \ge 0$, and the final line uses the normalization of the routing weights. Hence the representation reconstructed from the surviving experts remains within the same approximation radius ε_k of the underlying mode. We now propagate this representation error through the downstream computation. By the assumed L_k-Lipschitz continuity of the target margin,

$$\begin{aligned}|m_x(g_{U_k}(x;r)) - m_x(g_k^\star(x))| &\le L_k\,\|g_{U_k}(x;r) - g_k^\star(x)\| \\ &\le L_k\varepsilon_k,\end{aligned} \tag{46}$$

where the second inequality follows from Equation 45. To obtain a lower bound on the recovered margin, recall that for any real numbers a and b and any $c \ge 0$, the inequality $|a-b| \le c$ implies

$$-c \le a - b \le c.$$

Applying this relation with $a = m_x(g_{U_k}(x;r))$, $b = m_x(g_k^\star(x))$, and $c = L_k\varepsilon_k$ yields

$$\begin{aligned}-L_k\varepsilon_k &\le m_x(g_{U_k}(x;r)) - m_x(g_k^\star(x)) \\ &\le L_k\varepsilon_k.\end{aligned} \tag{47}$$

The left inequality in Equation 47 gives

$$\begin{aligned}-L_k\varepsilon_k &\le m_x(g_{U_k}(x;r)) - m_x(g_k^\star(x)) \\ m_x(g_k^\star(x)) - L_k\varepsilon_k &\le m_x(g_{U_k}(x;r))\,,\end{aligned} \tag{48}$$

where the second line is obtained by adding $m_x(g_k^\star(x))$ to both sides. Reversing the two sides for readability,

$$m_x(g_{U_k}(x;r)) \ge m_x(g_k^\star(x)) - L_k\varepsilon_k. \tag{49}$$

The underlying mode representation is assumed to recover the target with margin at least γ_k, i.e., $m_x(g_k^\star(x)) \ge \gamma_k$. Substituting this lower bound into Equation 49 gives

$$\begin{aligned}m_x(g_{U_k}(x;r)) &\ge m_x(g_k^\star(x)) - L_k\varepsilon_k \\ &\ge \gamma_k - L_k\varepsilon_k.\end{aligned} \tag{50}$$

This proves Equation 43. Finally, suppose $\gamma_k > L_k\varepsilon_k$. Then $\gamma_k - L_k\varepsilon_k > 0$, and the bound above implies $m_x(g_{U_k}(x;r)) > 0$. By definition of the target margin,

$$\begin{aligned}m_x(g_{U_k}(x;r)) > 0 \quad &\iff \quad R_{x,y^\star}(g_{U_k}(x;r)) > \max_{y\ne y^\star} R_{x,y}(g_{U_k}(x;r)) \\ &\implies \widehat{y}(x;r) = y^\star.\end{aligned} \tag{51}$$

Hence the target remains recoverable under the counterfactual route even after the partial expert edit. □

Exact expert splitting as a special case. The connection to the non-identifiability result of the first subsection becomes particularly transparent when the experts are exact split copies. Suppose $g_i(x) = g_k^\star(x)$ for every $i \in G_k$. Then $\varepsilon_k = 0$, and for any nonempty surviving subset U_k,

$$\begin{aligned} g_{U_k}(x;r) &= \sum_{i\in U_k} \alpha_i(x;r) g_i(x) \\ &= g_k^\star(x) \sum_{i\in U_k} \alpha_i(x;r) \\ &= g_k^\star(x). \end{aligned} \tag{52}$$

The target margin is therefore preserved exactly: $m_x(g_{U_k}(x;r)) = m_x(g_k^\star(x))$. Thus, in the exact splitting regime, editing all but one redundant copy still does not remove the functional mode. Any surviving copy is sufficient to reconstruct it under an appropriate route.

Interpretation. Theorem 3 provides the missing connection between redundant specialization and unlearning. The previous two subsections established that one functional component can admit several expert representations and that several such experts can remain jointly specialized to the same mode. The present result shows why this redundancy matters after an unlearning intervention. The critical quantity is not the number of experts that were edited, but whether the intervention removes all sufficiently accurate functional substitutes. If even one nonempty collection of specialists remains and counterfactual routing can place its mass on those survivors, their convex combination stays within ε_k of the original mode. The downstream target margin can therefore decrease by at most $L_k\varepsilon_k$. This yields a simple recovery criterion. If $\gamma_k > L_k\varepsilon_k$ then the mode remains recoverable after any partial edit that leaves a surviving specialist route. The result also clarifies why expert-level localization is insufficient as an unlearning criterion. Identifying and modifying a highly associated expert may remove one representation of the mode while leaving another functionally equivalent or sufficiently accurate representation untouched.

C.6 Knowledge-Route Cover and Necessary Intervention

The previous subsection considered one redundantly specialized expert group and showed that a target mode can remain recoverable whenever a sufficiently accurate specialist survives the edit. In a multilayer MoE, however, recovery need not be confined to one expert group. The same target may be supported by different collections of experts across layers, and different routing interventions may expose different collections. This suggests that the relevant object for robust unlearning is not an individual expert, but the complete routed computation through which the target remains recoverable. We now formalize this view and derive a necessary coverage condition for route-robust forgetting.

Routed expert support. Denote by $\mathcal{E}$ the set of routed expert blocks in the model. An element $e \in \mathcal{E}$ identifies both the MoE layer and the expert within that layer. For a forget example q and an admissible route r, we write $P_\theta(q,r) \subseteq \mathcal{E}$ for the set of expert blocks that receive positive post-routing weight when checkpoint θ processes q under r. Thus, $P_\theta(q,r)$ records the routed experts that actually participate in the computation. We call it a *route support* rather than a path because a top-k MoE may activate several experts simultaneously within the same layer. Suppose an unlearning intervention modifies only a set $S \subseteq \mathcal{E}$ of expert blocks. We denote the resulting checkpoint by θ_S. The intervention is *expert-local* when all expert parameters outside S, all shared model parameters, and the router parameters remain equal to those of the reference checkpoint θ_0. Formally,

$$(\theta_S)_e = (\theta_0)_e \quad \text{for every } e \notin S,$$

while $(\theta_S)_{\text{shared}} = (\theta_0)_{\text{shared}}$ and $(\theta_S)_{\text{rout}} = (\theta_0)_{\text{rout}}$. For stochastic routing interventions, such as additive gate noise, we compare the two checkpoints under the same realization of the intervention randomness. This ensures that whenever the router receives the same hidden representation, the same counterfactual routing transformation is applied. The following lemma gives the structural property that drives the route-cover result.

Lemma 5 (Preservation of an Untouched Recovery Computation)**.** *Fix a forget example q and a route r. Let θ_S be an expert-local intervention. If $P_{\theta_0}(q,r) \cap S = \varnothing$, then the complete model output under route r is unchanged: $f_{\theta_S}(q;r) = f_{\theta_0}(q;r)$.*

Proof. We compare the forward computations of θ_0 and θ_S on the same input q under the same routing rule r. Let $h_\ell^{(0)}$ and $h_\ell^{(S)}$ denote the hidden representations entering the ℓ-th MoE layer under θ_0 and θ_S, respectively. Before the first layer in which an editable expert can be used, the two models have identical parameters. Therefore their hidden representations are identical, so initially $h_1^{(S)} = h_1^{(0)}$. Assume that the two representations entering some MoE layer ℓ are equal, so that $h_\ell^{(S)} = h_\ell^{(0)}$. Because the router parameters are unchanged, equal router inputs produce the same router scores. Applying the same routing intervention r to these equal scores therefore gives the same active experts and the same post-routing weights under the two checkpoints. Write $w_{\ell,t,e}^{(0)}$ and $w_{\ell,t,e}^{(S)}$ for the post-routing weight assigned to expert e for token t. The preceding argument gives

$$w_{\ell,t,e}^{(S)} = w_{\ell,t,e}^{(0)} \qquad \text{for every } t \text{ and } e.$$

Whenever $w_{\ell,t,e}^{(0)} > 0$, the expert e belongs to $P_{\theta_0}(q, r)$. The assumption $P_{\theta_0}(q, r) \cap S = \varnothing$ therefore implies that every expert used by the routed computation lies outside the edit set. Hence its parameters are unchanged:

$$(\theta_S)_e = (\theta_0)_e \qquad \text{whenever } w_{\ell,t,e}^{(0)} > 0.$$

For token t, denote the corresponding MoE output under the two checkpoints by $u_{\ell,t}^{(0)}$ and $u_{\ell,t}^{(S)}$. Because both the active experts and their routing weights are identical, we obtain

$$\begin{aligned} u_{\ell,t}^{(S)} &= \sum_{e \in \mathcal{E}_\ell} w_{\ell,t,e}^{(S)} g_e^{(S)}\left(h_{\ell,t}^{(S)}\right) \\ &= \sum_{e \in \mathcal{E}_\ell} w_{\ell,t,e}^{(0)} g_e^{(0)}\left(h_{\ell,t}^{(0)}\right) \\ &= u_{\ell,t}^{(0)}. \end{aligned} \tag{53}$$

The second equality uses three facts simultaneously: $h_\ell^{(S)} = h_\ell^{(0)}$, the routing weights are equal, and every expert with nonzero routing weight is unedited. All shared transformations following this MoE computation are also unchanged. Applying the same residual, normalization, attention, and other shared mappings to identical inputs therefore produces identical representations at the next layer. Hence $h_{\ell+1}^{(S)} = h_{\ell+1}^{(0)}$. The argument can now be repeated through the remaining layers. By induction, the two checkpoints produce identical hidden representations throughout the complete forward computation, and therefore identical final outputs:

$$f_{\theta_S}(q; r) = f_{\theta_0}(q; r).$$

This completes the proof of Lemma 5 . □

Lemma 5 identifies the key asymmetry of expert-local unlearning. An intervention can only alter a routed computation once that computation encounters an edited expert. If its entire expert support avoids S, the computation is functionally invisible to the intervention.

Recovery witnesses and uncovered supports. We now translate this structural preservation result into a bound on forget efficacy. Let the forget set contain n_f examples $q_1, \ldots, q_{n_f}$. We write the forget efficacy under route r as

$$\mathrm{FE}(\theta; r) = \frac{1}{n_f} \sum_{i=1}^{n_f} a_\theta(q_i; r),$$

where $a_\theta(q_i; r) \in [0, 1]$ is the per-example recovery score. For WMDP, $a_\theta(q_i; r)$ is the indicator that the model selects the correct answer. The argument below only requires the score to be a deterministic function of the model output and to lie in $[0, 1]$. Fix a recovery level $\gamma \in (0, 1]$. We call (q_i, r) a *γ-recovery witness* of the reference checkpoint if $a_{\theta_0}(q_i; r) \geq \gamma$. For a given edit set S, define the uncovered witnesses under route r by

$$\mathcal{U}_\gamma(S, r) := \{i : a_{\theta_0}(q_i; r) \geq \gamma \text{ and } P_{\theta_0}(q_i, r) \cap S = \varnothing\}. \tag{54}$$

An index i belongs to $\mathcal{U}_\gamma(S, r)$ precisely when the reference model recovers the target strongly enough under route r, but the entire expert support responsible for that recovery avoids the intervention. We can now quantify the unavoidable residual recovery contributed by these untouched computations.

Theorem 4 (Knowledge-Route Cover)**.** *Let θ_S be any expert-local intervention on $S \subseteq \mathcal{E}$. Then, for every audited route r,*

$$\mathrm{FE}(\theta_S; r) \geq \frac{\gamma}{n_f} \left| \mathcal{U}_\gamma(S, r) \right|. \tag{55}$$

Consequently,

$$\mathrm{WRFE}_{\mathcal{R}}(\theta_S) \geq \frac{\gamma}{n_f} \max_{r \in \mathcal{R}} \left| \mathcal{U}_\gamma(S, r) \right|. \tag{56}$$

Therefore, if route-robust forgetting requires $\mathrm{WRFE}_{\mathcal{R}}(\theta_S) \leq \beta$ for some $\beta \in [0, 1]$, every feasible edit set must satisfy

$$\left| \mathcal{U}_\gamma(S, r) \right| \leq \left\lfloor \frac{\beta n_f}{\gamma} \right\rfloor \qquad \text{for every } r \in \mathcal{R}. \tag{57}$$

Proof. Fix an arbitrary route $r \in \mathcal{R}$. By definition,

$$\mathrm{FE}(\theta_S; r) = \frac{1}{n_f} \sum_{i=1}^{n_f} a_{\theta_S}(q_i; r).$$

Every recovery score is nonnegative. We may therefore discard all terms whose indices do not belong to $\mathcal{U}_\gamma(S, r)$ and obtain

$$\begin{aligned} \mathrm{FE}(\theta_S; r) &= \frac{1}{n_f} \sum_{i=1}^{n_f} a_{\theta_S}(q_i; r) \\ &= \frac{1}{n_f} \left[\sum_{i \in \mathcal{U}_\gamma(S,r)} a_{\theta_S}(q_i; r) + \sum_{i \notin \mathcal{U}_\gamma(S,r)} a_{\theta_S}(q_i; r) \right] \\ &\geq \frac{1}{n_f} \sum_{i \in \mathcal{U}_\gamma(S,r)} a_{\theta_S}(q_i; r). \end{aligned} \tag{58}$$

Consider now one index $i \in \mathcal{U}_\gamma(S, r)$. From the definition of the uncovered set, $P_{\theta_0}(q_i, r) \cap S = \varnothing$. Applying Lemma 5 gives us $f_{\theta_S}(q_i; r) = f_{\theta_0}(q_i; r)$. Because the per-example recovery score is a deterministic function of the final model output, identical outputs imply identical recovery scores: $a_{\theta_S}(q_i; r) = a_{\theta_0}(q_i; r)$. The same definition of $\mathcal{U}_\gamma(S, r)$ also gives $a_{\theta_0}(q_i; r) \geq \gamma$. Hence every uncovered witness satisfies $a_{\theta_S}(q_i; r) = a_{\theta_0}(q_i; r) \geq \gamma$. Substituting this pointwise lower bound into Equation 58 gives

$$\begin{aligned} \mathrm{FE}(\theta_S; r) &\geq \frac{1}{n_f} \sum_{i \in \mathcal{U}_\gamma(S,r)} a_{\theta_S}(q_i; r) \\ &= \frac{1}{n_f} \sum_{i \in \mathcal{U}_\gamma(S,r)} a_{\theta_0}(q_i; r) \\ &\geq \frac{1}{n_f} \sum_{i \in \mathcal{U}_\gamma(S,r)} \gamma \\ &= \frac{\gamma}{n_f} \left| \mathcal{U}_\gamma(S, r) \right|. \end{aligned} \tag{59}$$

This proves Equation 55. The worst-route forget efficacy is $\mathrm{WRFE}_{\mathcal{R}}(\theta_S) = \max_{r \in \mathcal{R}} \mathrm{FE}(\theta_S; r)$. Applying the route-wise lower bound to every $r \in \mathcal{R}$ therefore yields

$$\begin{aligned} \mathrm{WRFE}_{\mathcal{R}}(\theta_S) &= \max_{r \in \mathcal{R}} \mathrm{FE}(\theta_S; r) \\ &\geq \max_{r \in \mathcal{R}} \frac{\gamma}{n_f} \left| \mathcal{U}_\gamma(S, r) \right| \\ &= \frac{\gamma}{n_f} \max_{r \in \mathcal{R}} \left| \mathcal{U}_\gamma(S, r) \right|, \end{aligned} \tag{60}$$

which proves Equation 56. Next, suppose now that θ_S satisfies the desired route-robustness condition $\mathrm{WRFE}_{\mathcal{R}}(\theta_S) \le \beta$. For any route $r \in \mathcal{R}$, Equation 55 and the definition of WRFE imply

$$\begin{aligned} \frac{\gamma}{n_f} |\mathcal{U}_\gamma(S, r)| &\le \mathrm{FE}(\theta_S; r) \\ &\le \mathrm{WRFE}_{\mathcal{R}}(\theta_S) \\ &\le \beta. \end{aligned} \tag{61}$$

Multiplying by the positive quantity n_f/γ gives

$$|\mathcal{U}_\gamma(S, r)| \le \frac{\beta n_f}{\gamma}.$$

Since the cardinality $|\mathcal{U}_\gamma(S, r)|$ is an integer, it follows that

$$|\mathcal{U}_\gamma(S, r)| \le \left\lfloor \frac{\beta n_f}{\gamma} \right\rfloor.$$

Because the choice of r was arbitrary, this condition must hold for every route in $\mathcal{R}$, proving Equation 57. □

Interpretation. Theorem 4 turns residual recovery into a structural coverage requirement. If a reference recovery witness uses only experts outside the edit set, the intervention cannot alter that computation at all. Each such witness therefore contributes at least γ/n_f to the post-unlearning forget efficacy under the corresponding route. The resulting lower bound does not depend on how well the optimization procedure updates the experts inside S. Even an oracle optimizer cannot suppress a recovery computation whose entire support lies outside the editable set. The obstruction is therefore structural rather than optimization-induced.

The theorem also clarifies why the correct object is a *route cover*, rather than a list of individually important experts. Editing an expert with high forget affinity may destroy one recovery computation while leaving another support untouched. Robust forgetting requires the edit set to cover sufficiently many recovery witnesses under *every* admissible route. For binary recovery scores, such as WMDP accuracy, one may take $\gamma = 1$. The necessary condition then becomes

$$|\mathcal{U}_1(S, r)| \le \lfloor \beta n_f \rfloor \qquad \text{for every } r \in \mathcal{R}.$$

Thus, under each route, at most a β fraction of the originally recoverable forget examples may retain a computation whose entire expert support avoids the edit set.

Exact knowledge-route cover. A particularly transparent special case arises when the objective is to eliminate every γ-recovery witness. In that case no uncovered witness may remain, so $\mathcal{U}_\gamma(S, r) = \varnothing$ for every route r. Equivalently, every recovery witness must satisfy $P_{\theta_0}(q_i, r) \cap S \neq \varnothing$. The edit set must therefore intersect the support of every sufficiently strong recovery computation.

This exact-cover formulation will be useful when we later impose a graph structure on the recovery supports. Before doing so, however, we first ask a simpler question: given the necessary coverage condition above, how large must the edit set be? The next subsection converts the route-cover requirement into a minimum intervention budget and shows when route-robust forgetting is structurally impossible.

C.7 Minimum Edit Budget and Structural Impossibility

Theorem 4 gives a necessary coverage condition for route-robust forgetting. We now quantify the smallest expert budget for which that condition can be satisfied. For a recovery level $\gamma \in (0, 1]$ and a target threshold $\beta \in [0, 1]$, define the maximum admissible number of uncovered witnesses under any route as

$$m_{\gamma,\beta} := \left\lfloor \frac{\beta n_f}{\gamma} \right\rfloor.$$

The family of structurally admissible edit sets is

$$\mathcal{C}_{\gamma,\beta} := \{S \subseteq \mathcal{E} : |\mathcal{U}_\gamma(S, r)| \le m_{\gamma,\beta} \quad \forall r \in \mathcal{R}\}, \tag{62}$$

and the corresponding minimum edit budget is

$$\kappa_{\gamma,\beta} := \min_{S \in \mathcal{C}_{\gamma,\beta}} |S|. \tag{63}$$

If $\mathcal{C}_{\gamma,\beta} = \varnothing$, we set $\kappa_{\gamma,\beta} = +\infty$. This quantity depends only on the recovery supports, the audited route family, and the target tolerance; it is independent of how the selected experts are optimized.

Theorem 5 (Minimum Edit Budget and Structural Impossibility)**.** *Let θ_S be any expert-local intervention supported on $S \subseteq \mathcal{E}$. If $\mathrm{WRFE}_{\mathcal{R}}(\theta_S) \le \beta$, then $|S| \ge \kappa_{\gamma,\beta}$. Equivalently, every budget $B < \kappa_{\gamma,\beta}$ is structurally insufficient: any expert-local intervention satisfying $|S| \le B$ must have $\mathrm{WRFE}_{\mathcal{R}}(\theta_S) > \beta$.*

Proof. Assume that an expert-local intervention on S satisfies $\mathrm{WRFE}_{\mathcal{R}}(\theta_S) \le \beta$. For every $r \in \mathcal{R}$, Theorem 4 gives

$$\begin{aligned} \frac{\gamma}{n_f} |\mathcal{U}_\gamma(S, r)| &\le \mathrm{FE}(\theta_S; r) \\ &\le \mathrm{WRFE}_{\mathcal{R}}(\theta_S) \\ &\le \beta. \end{aligned} \tag{64}$$

Multiplying by $n_f/\gamma > 0$ yields $|\mathcal{U}_\gamma(S, r)| \le \beta n_f/\gamma$. Since the left-hand side is an integer, $|\mathcal{U}_\gamma(S, r)| \le \lfloor \beta n_f/\gamma \rfloor = m_{\gamma,\beta}$. Because this holds for every route, $S \in \mathcal{C}_{\gamma,\beta}$, and hence

$$\kappa_{\gamma,\beta} = \min_{S' \in \mathcal{C}_{\gamma,\beta}} |S'| \le |S|. \tag{65}$$

Thus every intervention achieving the desired worst-route threshold must edit at least $\kappa_{\gamma,\beta}$ experts.

Conversely, fix $B < \kappa_{\gamma,\beta}$ and any $S \subseteq \mathcal{E}$ with $|S| \le B$. Then $|S| < \kappa_{\gamma,\beta}$, so $S \notin \mathcal{C}_{\gamma,\beta}$. Therefore there exists some $r^\star \in \mathcal{R}$ such that $|\mathcal{U}_\gamma(S, r^\star)| > m_{\gamma,\beta}$. Integer-valuedness gives $|\mathcal{U}_\gamma(S, r^\star)| \ge m_{\gamma,\beta} + 1$, and Theorem 4 implies

$$\begin{aligned} \mathrm{WRFE}_{\mathcal{R}}(\theta_S) &\ge \mathrm{FE}(\theta_S; r^\star) \\ &\ge \frac{\gamma}{n_f} |\mathcal{U}_\gamma(S, r^\star)| \\ &\ge \frac{\gamma}{n_f} \left(m_{\gamma,\beta} + 1\right). \end{aligned} \tag{66}$$

Since $m_{\gamma,\beta} = \lfloor \beta n_f/\gamma \rfloor$, we have $m_{\gamma,\beta} + 1 > \beta n_f/\gamma$, and therefore $\mathrm{WRFE}_{\mathcal{R}}(\theta_S) > \beta$. □

The theorem isolates a structural obstruction rather than an optimization one. If $\mathfrak{I}(S)$ denotes all expert-local interventions supported on S, then for every $B < \kappa_{\gamma,\beta}$,

$$\inf_{\substack{S \subseteq \mathcal{E} \\ |S| \le B}} \inf_{\theta \in \mathfrak{I}(S)} \mathrm{WRFE}_{\mathcal{R}}(\theta) > \beta. \tag{67}$$

The inner infimum allows arbitrary optimization over all editable parameters, and the outer infimum allows the best support of size at most B. The lower bound therefore persists even under an oracle optimizer because the remaining recovery witnesses never encounter the editable experts.

A Simple Example. Consider three recovery supports $P_1 = \{e_1, e_2\}$, $P_2 = \{e_2, e_3\}$, and $P_3 = \{e_4, e_5\}$. No single expert intersects all three supports: editing e_2 blocks P_1 and P_2 but leaves P_3 untouched, while an expert from P_3 leaves at least one of the first two supports untouched. The set $S^\star = \{e_2, e_4\}$ intersects all three, so the exact-cover budget is $\kappa_\gamma^{\mathrm{exact}} = 2$. The example also shows why support count alone is insufficient: three recovery supports require only two edits because P_1 and P_2 share the bottleneck expert e_2.

Exact-cover formulation. When no γ-recovery witness may remain untouched, define the recovery support family

$$\mathscr{P}_\gamma := \{P_{\theta_0}(q_i, r) : a_{\theta_0}(q_i; r) \ge \gamma,\ r \in \mathcal{R}\}.$$

The minimum exact-cover budget becomes

$$\kappa_\gamma^{\mathrm{exact}} := \min_{S \subseteq \mathcal{E}} \{|S| : S \cap P \ne \varnothing \quad \forall P \in \mathscr{P}_\gamma\}. \tag{68}$$

Thus the experts form the ground set, the recovery supports form a hyperedge family, and $\kappa_\gamma^{\mathrm{exact}}$ is the minimum hitting-set size of the recovery-support hypergraph. Any set smaller than this budget leaves at least one support untouched, and Lemma 5 then guarantees that the corresponding routed computation is preserved exactly. The hitting-set formulation identifies the correct combinatorial object but does not yet explain which recovery structures produce a large minimum cover. A family containing many supports may still have a small hitting set when those supports share common experts, while disjoint supports force distinct interventions. The next subsection specializes this hypergraph view to a graph-realizable recovery structure, where the minimum exact cover admits a cut–flow characterization.

C.8 From Finite Audits to Global Route Robustness

RExU identifies the highest-recovery route only within a finite audit family. This route need not be the globally worst route over all admissible routing interventions. The relevant question is therefore not whether RExU can identify the exact global maximizer, but whether progress on the finite audited family can still imply progress over a broader route space. Two effects separate these quantities. First, after RExU suppresses the currently worst audited route, recovery may move to another audited route, producing the balloon-squeeze behavior observed across rounds. Second, an unseen route outside the audit family may recover more strongly than every audited route. We show that these two losses can be separated explicitly: global worst-route recovery decreases whenever the reduction achieved on the targeted route exceeds both the cross-route offset within the audit family and the recovery that can remain hidden between audited routes.

Differentiable admissible route space. Let $\Omega \subset \mathbb{R}^p$ be a compact convex parameter space of admissible routing interventions. For $\boldsymbol{\xi} \in \Omega$, let $r_{\boldsymbol{\xi}}$ denote the corresponding route, so that $\mathcal{R}_{\mathrm{adm}} = \{r_{\boldsymbol{\xi}} : \boldsymbol{\xi} \in \Omega\}$. A finite audit family is represented by $\widehat{\Omega} = \{\widehat{\boldsymbol{\xi}}_1, \dots, \widehat{\boldsymbol{\xi}}_M\} \subset \Omega$. We assume that every admissible route lies within distance ε of an audited route,

$$\forall\, \boldsymbol{\xi} \in \Omega, \qquad \exists\, \widehat{\boldsymbol{\xi}} \in \widehat{\Omega} \quad \text{such that} \quad \|\boldsymbol{\xi} - \widehat{\boldsymbol{\xi}}\|_2 \le \varepsilon. \tag{69}$$

Thus ε measures the resolution of the finite audit: smaller ε means that no admissible route is far from the routes that are explicitly tested.

For a forget example q_i, recall the correct-answer margin $m_\theta(q_i; r)$ from Equation 15, and write $m_i^\theta(\boldsymbol{\xi}) = m_\theta(q_i; r_{\boldsymbol{\xi}})$. The routed computation is assumed to be differentiable with respect to $\boldsymbol{\xi}$ throughout Ω, with uniformly bounded route sensitivity,

$$\left\|\nabla_{\boldsymbol{\xi}} m_i^\theta(\boldsymbol{\xi})\right\|_2 \le L_{\mathrm{route}} \qquad \text{for every } i \text{ and } \boldsymbol{\xi} \in \Omega, \tag{70}$$

for the checkpoints considered below. Hence nearby routes cannot change the correct-answer margin arbitrarily strongly.

Smooth margins alone do not make FE smooth, because FE changes whenever a margin crosses zero. The only audited examples that an unseen nearby route can newly recover are therefore those already close to this decision boundary. For $\rho \ge 0$, we write

$$\Gamma_\theta(\rho) = \max_{\widehat{\boldsymbol{\xi}} \in \widehat{\Omega}} \frac{1}{n_f} \sum_{i=1}^{n_f} \mathbb{I}\left\{-\rho < m_i^\theta(\widehat{\boldsymbol{\xi}}) \le 0\right\}. \tag{71}$$

This quantity is the largest fraction of forget examples, over the audited routes, that are currently not recovered but lie within ρ of becoming recovered. In particular, $\Gamma_\theta(L_{\mathrm{route}}\varepsilon)$ measures the recovery that may remain hidden between the finite audited routes.

We use

$$W_{\mathrm{adm}}(\theta) = \sup_{\boldsymbol{\xi} \in \Omega} \mathrm{FE}(\theta; r_{\boldsymbol{\xi}}), \qquad W_{\mathrm{audit}}(\theta) = \max_{\widehat{\boldsymbol{\xi}} \in \widehat{\Omega}} \mathrm{FE}(\theta; r_{\widehat{\boldsymbol{\xi}}}) \tag{72}$$

for the worst recovery over the admissible and audited route families, respectively.

Targeted reduction and cross-route offset. Consider one RExU round and hold its finite audit family $\widehat{\Omega}$ fixed while comparing the pre- and post-update checkpoints. At checkpoint θ_t, RExU selects

$$\widehat{\boldsymbol{\xi}}_t^\star \in \arg\max_{\widehat{\boldsymbol{\xi}} \in \widehat{\Omega}} \mathrm{FE}\big(\theta_t; r_{\widehat{\boldsymbol{\xi}}}\big), \tag{73}$$

so that $\mathrm{FE}(\theta_t; r_{\widehat{\boldsymbol{\xi}}_t^\star}) = W_{\mathrm{audit}}(\theta_t)$. The recovery removed from this targeted route during the round is

$$d_t = \mathrm{FE}\big(\theta_t; r_{\widehat{\boldsymbol{\xi}}_t^\star}\big) - \mathrm{FE}\big(\theta_{t+1}; r_{\widehat{\boldsymbol{\xi}}_t^\star}\big). \tag{74}$$

After the update, another audited route may become worse than the route that was targeted. We write the resulting cross-route offset as

$$b_t = W_{\mathrm{audit}}(\theta_{t+1}) - \mathrm{FE}\big(\theta_{t+1}; r_{\widehat{\boldsymbol{\xi}}_t^\star}\big) \geq 0. \tag{75}$$

If the targeted route remains the worst audited route, then $b_t = 0$. If another route overtakes it, b_t measures exactly how much of the targeted improvement is offset at the level of the audited maximum. Balloon-squeeze behavior is one mechanism that makes this term positive.

Theorem 6 (Finite-Audit Control and Global Route-Space Descent)**.** *Under Equation 69 and Equation 70, the gap between global and audited worst-route FE satisfies*

$$0 \leq W_{\mathrm{adm}}(\theta) - W_{\mathrm{audit}}(\theta) \leq \Gamma_\theta\big(L_{\mathrm{route}}\varepsilon\big). \tag{76}$$

For the RExU round in Equations 73–75,

$$W_{\mathrm{adm}}(\theta_{t+1}) \leq W_{\mathrm{adm}}(\theta_t) - d_t + b_t + \Gamma_{\theta_{t+1}}\big(L_{\mathrm{route}}\varepsilon\big). \tag{77}$$

Consequently,

$$d_t > b_t + \Gamma_{\theta_{t+1}}\big(L_{\mathrm{route}}\varepsilon\big) \tag{78}$$

is sufficient for $W_{\mathrm{adm}}(\theta_{t+1}) < W_{\mathrm{adm}}(\theta_t)$.

Proof. We first control how much the recovery margin can change between an arbitrary admissible route and its nearest audited route. Fix $\boldsymbol{\xi} \in \Omega$ and choose $\widehat{\boldsymbol{\xi}} \in \widehat{\Omega}$ satisfying $\|\boldsymbol{\xi} - \widehat{\boldsymbol{\xi}}\|_2 \leq \varepsilon$. Because Ω is convex, the segment $\widehat{\boldsymbol{\xi}} + s(\boldsymbol{\xi} - \widehat{\boldsymbol{\xi}})$ remains in Ω for $s \in [0,1]$. The fundamental theorem of calculus gives

$$\begin{aligned} m_i^\theta(\boldsymbol{\xi}) - m_i^\theta(\widehat{\boldsymbol{\xi}}) &= \int_0^1 \frac{d}{ds} m_i^\theta\left(\widehat{\boldsymbol{\xi}} + s(\boldsymbol{\xi} - \widehat{\boldsymbol{\xi}})\right) ds \\ &= \int_0^1 \nabla_{\boldsymbol{\xi}} m_i^\theta\left(\widehat{\boldsymbol{\xi}} + s(\boldsymbol{\xi} - \widehat{\boldsymbol{\xi}})\right)^\top (\boldsymbol{\xi} - \widehat{\boldsymbol{\xi}})\, ds. \end{aligned} \tag{79}$$

Taking absolute values, applying Cauchy–Schwarz, and then using Equation 70,

$$\begin{aligned} \left|m_i^\theta(\boldsymbol{\xi}) - m_i^\theta(\widehat{\boldsymbol{\xi}})\right| &\leq \int_0^1 \left\|\nabla_{\boldsymbol{\xi}} m_i^\theta\left(\widehat{\boldsymbol{\xi}} + s(\boldsymbol{\xi} - \widehat{\boldsymbol{\xi}})\right)\right\|_2 \left\|\boldsymbol{\xi} - \widehat{\boldsymbol{\xi}}\right\|_2 ds \\ &\leq \int_0^1 L_{\mathrm{route}} \left\|\boldsymbol{\xi} - \widehat{\boldsymbol{\xi}}\right\|_2 ds \\ &\leq L_{\mathrm{route}}\varepsilon. \end{aligned} \tag{80}$$

Suppose now that example q_i is recovered under the unseen route $r_{\boldsymbol{\xi}}$, so $m_i^\theta(\boldsymbol{\xi}) > 0$. From Equation 80,

$$\begin{aligned} m_i^\theta(\widehat{\boldsymbol{\xi}}) &\geq m_i^\theta(\boldsymbol{\xi}) - \left|m_i^\theta(\boldsymbol{\xi}) - m_i^\theta(\widehat{\boldsymbol{\xi}})\right| \\ &> -L_{\mathrm{route}}\varepsilon. \end{aligned} \tag{81}$$

Hence an example recovered by the unseen route must fall into one of two cases at the nearby audited route: it is already recovered there, or its margin is negative but lies within $L_{\mathrm{route}}\varepsilon$ of zero. Pointwise,

$$\mathbb{I}\left\{m_i^\theta(\boldsymbol{\xi}) > 0\right\} \leq \mathbb{I}\left\{m_i^\theta(\widehat{\boldsymbol{\xi}}) > 0\right\} + \mathbb{I}\left\{-L_{\mathrm{route}}\varepsilon < m_i^\theta(\widehat{\boldsymbol{\xi}}) \leq 0\right\}. \tag{82}$$

Averaging over the forget set therefore yields

$$\begin{aligned} \mathrm{FE}(\theta; r_{\boldsymbol{\xi}}) &= \frac{1}{n_f}\sum_{i=1}^{n_f} \mathbb{I}\left\{m_i^\theta(\boldsymbol{\xi}) > 0\right\} \\ &\leq \frac{1}{n_f}\sum_{i=1}^{n_f} \mathbb{I}\left\{m_i^\theta(\widehat{\boldsymbol{\xi}}) > 0\right\} + \frac{1}{n_f}\sum_{i=1}^{n_f} \mathbb{I}\left\{-L_{\mathrm{route}}\varepsilon < m_i^\theta(\widehat{\boldsymbol{\xi}}) \leq 0\right\} \\ &\leq W_{\mathrm{audit}}(\theta) + \Gamma_\theta\big(L_{\mathrm{route}}\varepsilon\big). \end{aligned} \tag{83}$$

Since $\boldsymbol{\xi}$ was arbitrary, taking the supremum over Ω gives

$$W_{\mathrm{adm}}(\theta) \le W_{\mathrm{audit}}(\theta) + \Gamma_\theta\big(L_{\mathrm{route}}\varepsilon\big).$$

Conversely, $\widehat{\Omega} \subseteq \Omega$, so every audited route is also admissible and $W_{\mathrm{audit}}(\theta) \le W_{\mathrm{adm}}(\theta)$. Combining these two inequalities proves Equation 76.

We next track what one RExU update does to the audited maximum. Because $\widehat{\boldsymbol{\xi}}_t^\star$ is the worst audited route at θ_t,

$$W_{\mathrm{audit}}(\theta_t) = \mathrm{FE}\big(\theta_t; r_{\widehat{\boldsymbol{\xi}}_t^\star}\big).$$

Equation 74 gives the post-update recovery on that same route, while Equation 75 accounts for any audited route that becomes worse after the update. Therefore,

$$\begin{aligned} W_{\mathrm{audit}}(\theta_{t+1}) &= \mathrm{FE}\big(\theta_{t+1}; r_{\widehat{\boldsymbol{\xi}}_t^\star}\big) + b_t \\ &= \mathrm{FE}\big(\theta_t; r_{\widehat{\boldsymbol{\xi}}_t^\star}\big) - d_t + b_t \\ &= W_{\mathrm{audit}}(\theta_t) - d_t + b_t. \end{aligned} \tag{84}$$

Thus d_t is the recovery removed from the route RExU explicitly targets, whereas b_t is the part of that improvement lost because another audited route becomes more severe.

Finally, applying Equation 76 to the post-update checkpoint and then using Equation 84,

$$\begin{aligned} W_{\mathrm{adm}}(\theta_{t+1}) &\le W_{\mathrm{audit}}(\theta_{t+1}) + \Gamma_{\theta_{t+1}}\big(L_{\mathrm{route}}\varepsilon\big) \\ &= W_{\mathrm{audit}}(\theta_t) - d_t + b_t + \Gamma_{\theta_{t+1}}\big(L_{\mathrm{route}}\varepsilon\big) \\ &\le W_{\mathrm{adm}}(\theta_t) - d_t + b_t + \Gamma_{\theta_{t+1}}\big(L_{\mathrm{route}}\varepsilon\big), \end{aligned} \tag{85}$$

where the final inequality follows from $W_{\mathrm{audit}}(\theta_t) \le W_{\mathrm{adm}}(\theta_t)$. This proves Equation 77. If $d_t > b_t + \Gamma_{\theta_{t+1}}(L_{\mathrm{route}}\varepsilon)$, then the final two terms cannot offset the targeted reduction, and Equation 85 gives $W_{\mathrm{adm}}(\theta_{t+1}) < W_{\mathrm{adm}}(\theta_t)$. □

What the bound means. Theorem 6 separates the loss of progress into two distinct mechanisms. The term b_t is internal to the audited family: even when RExU successfully suppresses the route it targets, another audited route can become the new worst route and offset part of that reduction. The term $\Gamma_{\theta_{t+1}}(L_{\mathrm{route}}\varepsilon)$ comes from routes that were never audited. The product $L_{\mathrm{route}}\varepsilon$ determines how far an unseen route can move an example's margin relative to its nearby audited route, while $\Gamma_\theta(\cdot)$ counts how many examples are close enough to the decision boundary for such a change to alter FE. A dense audit family reduces ε, and route-stable computation reduces L_{route}; both shrink the margin band in which unseen recovery can hide.

A useful limiting case occurs when no audited forget example has margin in $(-L_{\mathrm{route}}\varepsilon, 0]$. Then $\Gamma_\theta(L_{\mathrm{route}}\varepsilon) = 0$, and Equation 76 gives $W_{\mathrm{adm}}(\theta) = W_{\mathrm{audit}}(\theta)$ even though the global worst route itself may never have been evaluated. More generally, RExU does not need to identify the exact global worst route. It is sufficient that the finite audit family represents the admissible route space well enough, the recovery margin changes smoothly between nearby routes, and the reduction on the targeted route is larger than the recovery that moves elsewhere:

$$\underbrace{d_t}_{\text{targeted recovery reduction}} > \underbrace{b_t}_{\text{cross-route offset}} + \underbrace{\Gamma_{\theta_{t+1}}\big(L_{\mathrm{route}}\varepsilon\big)}_{\text{finite-audit coverage error}}. \tag{86}$$

Under this condition, the worst recovery over the entire admissible route space must decrease, even though RExU only searches a finite subset of that space.

Scope under hard top-k routing. The preceding result applies directly when the route parameterization is differentiable throughout Ω. Hard top-k routing is only piecewise smooth because the selected expert set can change when two router scores cross. Let $s_{(k)}$ and $s_{(k+1)}$ be the k-th and $(k+1)$-st largest router scores at a routed decision, and suppose a route perturbation changes every score by at most ζ. For any currently selected expert e and unselected expert j,

$$s'_e - s'_j \ge (s_{(k)} - \zeta) - (s_{(k+1)} + \zeta) = \big(s_{(k)} - s_{(k+1)}\big) - 2\zeta. \tag{87}$$

Thus $s_{(k)} - s_{(k+1)} > 2\zeta$ preserves the active top-k support. Within such support-stable regions the routed computation may be analyzed by the differentiable argument above. Crossing a top-k boundary is not covered by this guarantee; controlling the full hard-routing space requires either separate coverage of the resulting routing regions or a smooth routing relaxation.

Implication for RExU. The main implication is that RExU does not need to identify the exact global worst route. If the audited routes cover the routing space reasonably well and recovery changes smoothly between nearby routes, then the worst audited route is already a good approximation of the global worst route. After each round, RExU reduces recovery on the current worst audited route, but part of this improvement can be lost if recovery shifts to another audited route or remains hidden in an unseen route. Therefore, global worst-route recovery decreases whenever the reduction achieved by RExU is larger than these two effects.

D EXPERIMENT DETAILS

Algorithm 1 RExU: Recovery-Expert Unlearning

Input: Initial model θ_0; forget and retain sets D_f, D_r; set of route intervention $\mathcal{R}_{\text{train}}$; unlearner $\mathcal{U}$; budget B; pool size $N_c \geq B$; rounds T; tolerances τ, u_0; chance $p_{\text{chance}} = 1/K$
Output: best feasible checkpoint θ_{best}

$\theta \leftarrow \theta_0$; $\theta_{\text{best}} \leftarrow \theta_0$;
for $t = 0, \ldots, T$ **do**
  Compute $u_t \leftarrow \text{UT}(\theta_t)$
  Compute $r_t^\star \leftarrow \arg\max_{r \in \mathcal{R}_{\text{train}}} \text{FE}(\theta_t; r)$ ▷ $FE(\theta_t, r_t^*) = \text{WRFE}_{\mathcal{R}_{\text{train}}}(\theta_t)$
  if $u_t < u_0$ **or** $\text{FE}(\theta_t, r_t^*) \leq p_{\text{chance}} + \tau$ **then**
    Set $T \leftarrow t$
    break
  end if
  Compute $A_t(l, e)$ by Equation 10
  Select $\mathcal{P}_t \leftarrow \text{Top}_{N_c}\{A_t(l, e)\}$
  Compute $C_t(l, e)$, $(l, e) \in \mathcal{P}_t$, by Equation 89
  Select $\mathcal{E}_t \leftarrow$ top-B of $\{(l, e) \in \mathcal{P}_t : C_t(l, e) > 0\}$
  if $\mathcal{E}_t = \varnothing$ **then**
    Set $T \leftarrow t$
    break
  end if
  $\theta_{t+1} \leftarrow \mathcal{U}(\theta_t; \mathcal{E}_t, D_f, D_r, r_t^\star)$ ▷ from Equation 12
end for
return $\theta_{\text{best}} \in \arg\min_{(\theta_t)_{t=0,1,\ldots,T}} \text{FE}(\theta_t, r_t^*)$

Code and reproducibility. All experiments use PyTorch (Paszke et al., 2019) (Python 3.12, CUDA 12.6, PyTorch 2.9). Code, training scripts, hyperparameter configs, and precomputed logs are at `https://anonymous.4open.science/r/rexu-82E2/`; camera-ready release will be under MIT License with a Zenodo DOI.

Tasks and datasets. We evaluate MoE unlearning on two established LLM unlearning benchmarks: WMDP (Li et al., 2024) and RWKU (Cao et al., 2024). For WMDP, we use the WMDP-Cyber subset, a four-choice multiple-choice benchmark for hazardous cybersecurity knowledge. Forget efficacy (FE) is measured as post-unlearning accuracy on held-out WMDP-Cyber questions; lower is better and chance performance is 25%. For RWKU, we follow the original benchmark protocol and evaluate forgetting of real-world entity knowledge using fill-in-the-blank and question-answer prompts. Following prior work, we select 100 target entities for unlearning and use their provided `train_original_passage` descriptions as the forget set. For RWKU, FE is measured by ROUGE-L recall (Lin, 2004) between the model output and the reference answer; lower is better, with an ideal value of 0.

RExU on generation benchmarks (RWKU). Algorithm 1 is stated for multiple-choice forget sets; on RWKU we keep the loop unchanged and replace its two multiple-choice primitives. The recovery signal is the mean per-token log-likelihood of the forget passage x under route r, $\ell_\theta(x; r) = \frac{1}{|x|} \sum_j \log p_\theta(x_j \mid x_{<j}; r)$, which plays the role of the margin m_θ: the worst route $r_t^\star$ maximizes $\mathbb{E}_{x \sim D_f}[\ell_{\theta_t}(x; r)]$ in Equation 9, and the recovery contribution $C_t(l, e)$ of Equation 89 is the drop in this quantity when (l, e) is ablated. The forget metric that enters the route-robust criterion is ROUGE-L recall on the calibration probes, whose floor is $p_{\text{chance}} = 0$ (Definition 3 with $1/K$ replaced by 0), with the same tolerance $\tau = 0.01$ (1.0 on the $\times 100$ scale of Table 5). The base unlearner $\mathcal{U}$ and the KL retain term are applied to the forget passages exactly as on WMDP. Every RWKU run exits on the round budget (final WRFE 2.1-4.1 against the threshold 1.0).

Target MoE models. We evaluate two primary open-weight MoE LLMs: Qwen1.5-MoE-A2.7B-Chat (Qwen)[1] and DeepSeek-V2-Lite (DeepSeek).[2] These models represent different MoE construction regimes: Qwen is upcycled from the dense Qwen-1.8B model, whereas DeepSeek is pretrained as an MoE from scratch. Qwen has 14.3B total parameters and 2.7B activated parameters per token, with 24 MoE layers that each select 4 of 60 routed experts. DeepSeek has approximately 16B total parameters and 2.4B activated parameters per token; its 27 decoder layers include 26 MoE layers, each selecting 6 of 64 routed experts. To examine generalization at a larger scale, we additionally evaluate Qwen3-Next-80B-A3B-Instruct,[3] which has 80B total parameters and approximately 3B activated parameters per token. It comprises 48 MoE layers in a hybrid architecture combining Gated DeltaNet and gated attention, with each MoE layer selecting 10 of 512 routed experts. This extends our evaluation to a substantially larger model and expert population while retaining a similar range of activated parameter counts per token. Unless otherwise stated, DeepSeek is the primary model for analysis, Qwen is included in the main comparisons in Section 6, and the Qwen3-Next evaluation is reported in Appendix F.2.

Evaluation metrics. We report both natural-route forgetting and worst-route recovery. Natural-route FE evaluates the unlearned model under its default router decisions. Worst-route FE, denoted WRFE, is the maximum FE over the audited route family $\mathcal{R}$; lower is better. The route gap $\Delta_{\mathcal{R}} = \mathrm{WRFE} - \mathrm{FE}(r_{\mathrm{nat}})$ measures how much natural-route evaluation underestimates residual recoverability. For utility, we report zero-shot MMLU accuracy using the LM Evaluation Harness (Gao et al., 2023). For WMDP-Cyber, we additionally report residual accuracy above chance, $\mathrm{FE} - 25\%$, and the removed fraction relative to the pretrained model. For RWKU, we use the original prompt templates: "Please complete the blank in the following question. Question:" for fill-in-the-blank tasks and "Please briefly answer the following question. Question:" for question-answer tasks.

Baselines. We compare RExU against standard LLM unlearning baselines: Gradient Ascent (GA) (Yao & Xu, 2024), Gradient Difference (GDIFF) (Maini et al., 2024), Negative Preference Optimization (NPO) (Zhang et al., 2024), and Representation Misdirection for Unlearning (RMU) (Li et al., 2024). We also compare against MoE-specific and parameter-efficient baselines: SEUF (Zhuang et al., 2025), multi-expert SEUF-M with $M \in \{1, 4, 8\}$, LoRA (Hu et al., 2022), and ESFT (Wang et al., 2024); GA+LoRA, GA+ESFT, and the concurrent route-robustness methods GRIP (Zhu et al., 2026) and TRACE (Xie et al., 2026) are evaluated under the identical protocol and reported in Table 10 (Appendix F.2). To isolate the effect of recovery-driven expert selection, we include matched-budget selection baselines: random experts, natural-route affinity experts, counterfactual-affinity experts, and the signed recovery-contribution selector used by RExU (Table 11). The *route-aware full-model* baseline of Tables 9 and 13 is Algorithm 1 with the base unlearner applied to all parameters rather than to $\mathcal{E}_t$, i.e. worst-route selection without expert localization.

Checkpoint selection. For baseline methods, checkpoints are selected using only natural-route validation metrics, following the protocol of SEUF (Zhuang et al., 2025). Counterfactual routing interventions are applied only after checkpoint selection, so the baseline audit does not leak route information into model selection. For RExU, route interventions are used during training only through the training route family $\mathcal{R}_{\mathrm{train}}$; θ_{best} is selected on the calibration set $\mathcal{D}_f^{\mathrm{cal}}$, and final worst-route metrics are the maximum over the full family $\mathcal{R} \supset \mathcal{R}_{\mathrm{train}}$, evaluated on $\mathcal{D}_{\mathrm{test}}$ (below).

[1] Official model card: https://huggingface.co/Qwen/Qwen1.5-MoE-A2.7B-Chat. Model configuration: https://huggingface.co/Qwen/Qwen1.5-MoE-A2.7B-Chat/blob/main/config.json.

[2] Official model card: https://huggingface.co/deepseek-ai/DeepSeek-V2-Lite. Model configuration: https://huggingface.co/deepseek-ai/DeepSeek-V2-Lite/blob/main/config.json.

[3] Official model card: https://huggingface.co/Qwen/Qwen3-Next-80B-A3B-Instruct. Model configuration: https://huggingface.co/Qwen/Qwen3-Next-80B-A3B-Instruct/blob/main/config.json.

Counterfactual routing intervention definitions. We consider three intervention classes targeting distinct off-route failure modes. Let $\hat{\pi}_{l,e}$ denote the selection frequency of expert (l, e) on D_f under r_{nat}.

- **A1 – expert masking** (label-free): $(T_l^r(z))_e = -\infty$ if $(l, e) \in S$, and $(T_l^r(z))_e = z_e$ otherwise, for a reference expert set S. For checkpoints with an identifiable edited set, we use $S = S(\theta)$. For SEUF-M, this is the Top- M set ranked by natural-route selection frequency $\hat{\pi}_{l,e}$ over all (l, e), matching both its selection criterion and edit budget. This set is recoverable from the checkpoint and equals the weight-difference set (Appendix H). For the pretrained model and full-model methods, which have no identifiable edited subset, we use the same singleton reference set S_0 selected by top-1 SEUF under natural routing. When S is the edited set, above-chance FE under A1 witnesses that target knowledge remains recoverable without activating any expert in S.
- **A2 – gate perturbation**: additive noise $T_l^r(z) = z + \epsilon_l$, $\epsilon_l \sim \mathcal{N}(0, \sigma^2 I_{E_l})$, with $\sigma \in \{.05, .10\}$; or re-uniformization $T_l^r(z) = (1 - \beta)z + \beta\bar{z}\mathbf{1}$, where $\bar{z} = \frac{1}{E_l}\mathbf{1}^\top z$ and $\beta = .25$ (parameter choices are discussed in Appendix D).
- **A3 – rare-expert forcing**: $(T_l^r(z))_e = z_e - \gamma \log(\hat{\pi}_{l,e} + \kappa)$, with $\gamma = 1.0$ and $\kappa = 10^{-6}$ for numerical stability (Appendix D). This intervention up-weights long-tail fallback experts that are rarely selected for forget-set examples under the natural route.

Training and reporting route families. RExU treats robust MoE unlearning as constraint generation over a finite route family. During training, the loop audits

$$\mathcal{R}_{\text{train}} = \{r_{\text{nat}}, A1_{S_t}, A2_{\sigma=.05}, A2_\beta\},$$

where r_{nat} is the natural route, $A1_{S_t}$ masks the experts edited so far by the current run, $A2_{\sigma=.05}$ adds Gaussian router-logit noise, and $A2_\beta$ re-uniformizes the gate ($\beta = .25$); ties in Equation 9 are broken toward the route listed first. All reported worst-route numbers (Tables 1, 5, 9, 10 and the ablation tables) are the maximum over the full six-route family

$$\mathcal{R}(\theta) = \mathcal{R}_{\text{train}} \cup \{A2_{\sigma=.10},\ A3_\gamma\},$$

evaluated on the final checkpoint with $A1$ instantiated on that checkpoint's own edited set $S(\theta)$ Section 3.1). $A2_{\sigma=.10}$ and $A3_\gamma$ are never used for selection, and by Lemma 1(c) the reported $\text{WRFE}_{\mathcal{R}}$ is at least the training-family value $\text{WRFE}_{\mathcal{R}_{\text{train}}}$ the loop optimized, so reporting on $\mathcal{R}$ cannot make RExU appear more robust than the audit it was trained against; it also exposes RExU to the same masking probe, on its own edited set, that Finding 1 applies to SEUF.

Stochastic routes and seeds. For the stochastic A2 noise routes we audit a fixed, finite set of seeds rather than estimating a population quantile:

$$\text{FE}(\theta; r) = \max_{s \in [N_s]} \text{acc}_s(\theta; r), \qquad N_s = 3, \tag{88}$$

the sample maximum over the N_s audited seeds, consistent with the worst-route framing of Definition 2; WRFE is thus a maximum over the finite audited set $\mathcal{R} \times [N_s]$ and is defined relative to that set.

Remark 4 (Termination in practice)**.** *Algorithm 1 can exit for three reasons: the route-robust criterion $w \le p_{\text{chance}} + \tau$ is met, utility falls below u_0, or the round budget T is exhausted. Across every configuration reported in Table 5, the loop exits on the round budget: no run drives $\text{WRFE}_{\mathcal{R}_{\text{train}}}$ down to the chance-plus-tolerance threshold (on WMDP-Cyber, final WRFE ranges 27.2-33.2 against a threshold of 26.0 at $\tau = 0.01$; on RWKU, 2.1-4.1 ROUGE-L recall against a threshold of 1.0). RExU therefore reduces worst-route recovery substantially and returns a near-zero route gap without certifying the fixed-point stopping criterion of Definition 3; we report the realized WRFE and $\Delta_{\mathcal{R}}$ directly rather than treating termination itself as evidence of robustness.*

Evaluation Protocol Interventions are used only for evaluation unless stated otherwise, and the WMDP-Cyber test set is held for final reporting: route selection, expert attribution, checkpoint selection, and recovery-driven training use only forget-corpus calibration data and D_r. Every FE and WRFE we report, including the per-round trajectory of Figure 12, is evaluated on $\mathcal{D}_{\text{test}}$; the loop's internal decisions - $r_t^\star$, C_t, and θ_{best} - are made on $\mathcal{D}_f^{\text{cal}}$ alone. For each model we report

natural-route FE, $\mathrm{WRFE}_{\mathcal{R}}$, the route gap $\Delta_{\mathcal{R}}$, and retain utility UT; the full per-route breakdown is reported for the failure-mode analysis (Table 1), and edited-expert counts appear in Table 11 with per-run details in the released configs. By Lemma 1(a), $\Delta_{\mathcal{R}} \geq 0$: natural-route FE lower-bounds worst-route recovery, so a large $\Delta_{\mathcal{R}}$ quantifies how much the natural route *underestimates* recoverable accuracy.

Recovery is not an evaluation artifact. Two controls rule out leakage as the cause. *(i) Reporting family contains the training family:* WRFE is the maximum over $\mathcal{R} \supset \mathcal{R}_{\mathrm{train}}$, so it is bounded below by the training audit (Lemma 1(c)) and includes two intervention classes ($A2_{\sigma=.10}$, $A3_{\gamma}$) never used for selection. *(ii) No information added:* A1 masking only removes the edited expert, so the recovery it exposes - a return to within a point of the pretrained value (Finding 1) - reflects knowledge in *unedited* experts by Lemma 1(b), not the metric.

Hyperparameters and compute. Unless otherwise stated, all methods use AdamW (Loshchilov & Hutter, 2019). For RExU, we use budget $B = 8$ experts per round, candidate pool size $N_c = 32$, a round budget of $T = 5$ on DeepSeek and $T = 7$ on Qwen, margin slack $\delta = 0.1$, retain weight $\lambda = 1.0$, and stopping tolerances $\tau = 0.01$ and $u_0 = 0.45$. The KL retain reference is reset to the model at the start of each round. A2 noise routes use $N_s = 3$ fixed seeds. We train with learning rate 2×10^{-4}, micro-batch size 8, gradient accumulation 16, and 80 update steps per round. Experiments on the two primary models (Qwen, DeepSeek) are run on a single NVIDIA B200 GPU with approximately 180GB memory; one RExU run on DeepSeek takes approximately 3.0 GPU-hours.

Intervention parameter choices and sensitivity. For A2, $\sigma \in \{.05, .10\}$ spans a small-perturbation regime (comparable in scale to typical router-logit magnitude) up to a level still well short of randomizing the route; $\beta = .25$ moves the gate a quarter of the way toward uniform, a moderate re-uniformization rather than a full flattening ($\beta = 1$). For A3, $\kappa = 10^{-6}$ keeps the logarithm finite when $\hat{\pi}_{l,e} = 0$ and is otherwise insensitive, since $\hat{\pi}_{l,e} \gg \kappa$ for any expert with non-negligible natural-route frequency; $\gamma = 1.0$ makes the up-weighting scale linearly with log-inverse-frequency (an expert selected $10\times$ less often gets a logit boost of $\gamma \ln 10 \approx 2.3$). All four values ($\sigma$, β, γ, κ) were chosen as representative points in each regime rather than swept on a grid; Table 13 sweeps RExU's own hyperparameters (B, N_c) but we did not run an analogous sensitivity sweep over the intervention parameters themselves, so how worst-route recovery varies continuously with σ, β, or γ - rather than at the audited points - is not characterized here. Tuned hyperparameters, selected checkpoints, edited-expert counts, trainable parameter counts, and GPU-hours for every baseline are included in the released configs and logs.

Cost accounting. Per round, Algorithm 1 costs: an audit of $\mathcal{R}_{\mathrm{train}}$ on $\mathcal{D}_f^{\mathrm{cal}}$ (three deterministic routes plus the stochastic route at N_s=3 seeds, i.e. six evaluations), one routing-statistics pass for Equation 10, N_c=32 leave-one-out ablation evaluations for C_t, one utility evaluation, and 80 update steps (micro-batch 8, accumulation 16) confined to the B=8 selected experts - approximately 40 forward evaluations of the 128-item calibration set per round plus the confined update. End to end, T=5 rounds on DeepSeek measure 3.0 GPU-hours on one B200; per-variant GPU-hours for every ablation setting are in Tables 12 and 13.

E Dataset Examples

Figure 4 illustrates the multiple-choice format of WMDP-Cyber and the fill-in-the-blank and short-answer formats of RWKU.

(a) WMDP-Cyber
Multiple-choice evaluation

Question
Which command is used to change the permissions of a file?
A. `chgrp`
B. `chown`
C. `chmod`
D. `ch`

Reference label: C

FE: Accuracy↓
Chance accuracy: 25%

(b) RWKU
Generation-based evaluation

Fill-in-the-blank query
Stephen King's debut novel, titled ___, established him in the horror genre.

Reference answer: Carrie

Short-answer query
What is the title of Stephen King's first published novel?

Reference answer: Carrie

FE: ROUGE-L recall↓

Figure 4: **Example evaluation items from WMDP-Cyber and RWKU.** WMDP-Cyber evaluates answer-choice accuracy, whereas RWKU evaluates generated answers against reference answers using ROUGE-L recall. Reference annotations are shown for scoring; instruction prefixes and model-specific chat formatting are omitted for readability.

F Detailed Results

F.1 Detailed Comparison Results with Other Baselines

Table 5 reports natural-route FE, WRFE, and retain utility for each method across both models and benchmarks.

Table 5: **RExU reduces off-route recovery while preserving retain utility.** Results on Qwen and DeepSeek across WMDP-Cyber and RWKU. FE measures natural-route forget performance; WRFE is its maximum over the six-route audit family $\mathcal{R}$ (Equation 3); lower is better for both. A1 uses the checkpoint-specific masking set defined in Section 3.1. WMDP scores are accuracy percentages (chance: 25%); RWKU scores are ROUGE-L recall multiplied by 100. UT is natural-route zero-shot MMLU accuracy (↑). Each block compares a base unlearner with its SEUF and RExU variants. Colored parenthetical values denote absolute improvements (+) or declines (−) in score points.
💡 **Takeaway:** Across all 16 matched configurations, RExU lowers WRFE relative to SEUF. For example, on Qwen/WMDP, GDIFF+RExU reduces WRFE from 43.08 to 27.18 (+15.90) and increases UT from 51.29 to 60.72 (+9.43). For GA on the same setting, the route gap (WRFE − FE) falls from 18.17 to 1.06 (+17.11). UT improves over SEUF in 14 of 16 configurations; the exceptions are NPO on DeepSeek/RWKU, from 54.79 to 54.70 (−0.09), and RMU on Qwen/RWKU, from 59.75 to 59.20 (−0.55). Relative to pretrained models, RExU's utility declines range from −0.08 to −2.30 points.

Method	Qwen (WMDP)			DeepSeek (WMDP)			Qwen (RWKU)			DeepSeek (RWKU)		
	FE↓	WRFE↓	UT↑	FE↓	WRFE↓	UT↑	FE↓	WRFE↓	UT↑	FE↓	WRFE↓	UT↑
Pretrained	42.88	43.13	60.84	36.79	36.99	55.51	42.43	43.90	60.84	53.76	55.20	55.51
GA	24.71	24.86	40.37	24.31	25.62	24.65	0.78	1.50	48.49	8.39	10.20	51.95
GA+SEUF	24.61	42.78	42.10	26.57	37.90	39.57	0.60	18.20	57.09	0.00	14.60	54.85
GA+RExU	26.72	27.78	60.76	27.43	28.54	54.59	1.20	2.40	59.95	1.50	2.80	55.05
GDIFF	25.67	25.67	23.13	25.16	25.62	23.45	7.00	8.50	52.96	19.01	21.50	34.95
GDIFF+SEUF	27.33	43.08	51.29	25.01	37.39	50.25	0.10	16.00	59.87	0.00	13.20	52.53
GDIFF+RExU	25.82	27.18	60.72	26.02	27.18	55.12	0.80	2.10	60.02	1.20	2.60	55.10
NPO	34.72	39.75	47.83	33.82	37.02	47.30	0.00	3.50	37.18	9.70	12.20	53.88
NPO+SEUF	32.16	36.92	54.23	27.87	35.43	46.56	0.20	14.20	54.28	0.00	11.80	54.79
NPO+RExU	29.45	32.16	59.43	30.95	33.22	54.45	1.80	3.40	58.70	2.50	4.10	54.70
RMU	25.52	27.08	37.12	26.36	27.57	46.33	2.00	3.00	24.20	0.10	1.00	51.09
RMU+SEUF	26.72	35.15	53.61	28.73	36.17	48.59	7.23	19.00	59.75	1.30	14.00	53.88
RMU+RExU	26.52	27.28	59.69	27.23	29.10	53.21	2.50	4.10	59.20	2.10	3.80	54.80

F.2 Additional Results

F.2.1 Item-level matched-pair breakdown

Table 6 gives the full item-level counts behind the $82.6\% \rightarrow 16.7\%$ conditional recovery figures cited in the abstract, Contributions, and Finding 1 (Section 4): on the same 1,987 WMDP-Cyber items, an item is *conditionally recoverable* if the pretrained model answers it correctly and the natural route answers it incorrectly, and the *conditional recovery rate* is the fraction of such items that A1 masking flips back to correct.

Table 6: **Expert masking reveals substantial recovery after SEUF and residual recovery after RExU.** Item-level results on DeepSeek-V2-Lite/WMDP-Cyber ($n = 1{,}987$). A1 masks each checkpoint's edited set: the selected expert for GA+SEUF and the accumulated set S_t for GA+RExU. N_{ijk} counts items with correctness indicators (i, j, k) for pretrained, natural-route, and A1 predictions, respectively (1: correct; 0: incorrect). Thus, N_{101} counts recovered pretrained-correct answers, N_{100} counts those remaining incorrect, and N_{001} counts newly correct answers that were incorrect before unlearning. Conditional recovery is $N_{101}/(N_{101} + N_{100})$; brackets give 95% Wilson intervals. b and c count natural-correct/A1-wrong and natural-wrong/A1-correct items, respectively; continuity-corrected McNemar tests compare these paired outcomes within each checkpoint.
💡 **Takeaway:** A1 recovers $190/230$ (82.6%) of the conditionally recoverable items for GA+SEUF, versus $35/210$ (16.7%) for GA+RExU.

Method	N_{101}	N_{100}	N_{001}	b	c	Cond. recovery	McNemar (p)
GA+SEUF	190	40	50	15	240	82.6% [77.2,87.0]	χ^2=196.8, $p<10^{-44}$
GA+RExU	35	175	7	20	42	16.7% [12.2,22.3]	χ^2=7.1, p=0.008

F.2.2 NATURAL-ROUTE REPRODUCTION

Table 7 reproduces the SEUF-style natural-route evaluation on DeepSeek-V2-Lite/WMDP-Cyber. Figure 5 visualizes the corresponding forgetting–utility trade-off. The raw full-model GA and GDIFF baselines reach near-chance forget accuracy, but only by sacrificing substantial retain utility. Wrapping the base unlearners with SEUF generally recovers utility: the effect is particularly pronounced for GDIFF, whose MMLU accuracy increases from 23.45 to 50.25 while its natural-route FE remains near chance (25.16 versus 25.01). However, this natural-route comparison alone does not test whether the target behavior can re-emerge under alternative routing paths.

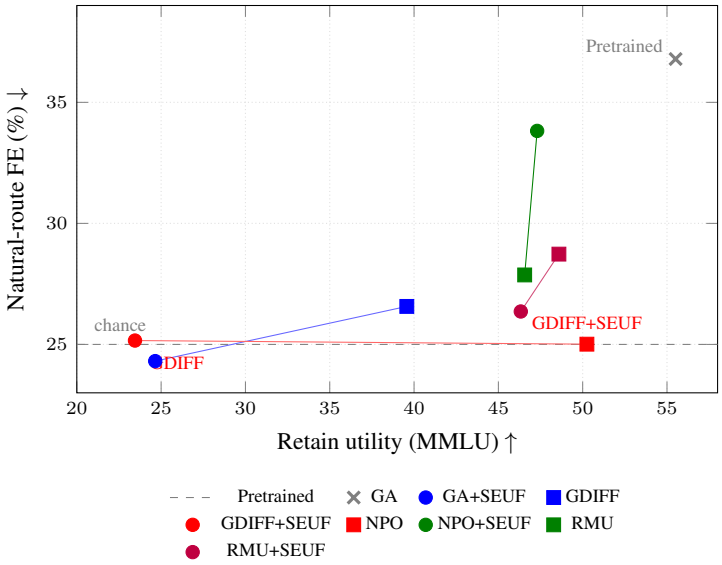


Figure 5: **Natural-route forget efficacy alone can obscure the forgetting–utility trade-off.** Results on DeepSeek-V2-Lite/WMDP-Cyber. Each point plots natural-route forget accuracy against zero-shot MMLU retain utility; lower-right is better. Circles denote raw full-model unlearners and squares denote their SEUF-wrapped counterparts; arrows connect each base method to its SEUF variant. GA and GDIFF reach near-chance FE only with severe utility loss. The largest utility recovery occurs for GDIFF+SEUF, which increases MMLU from 23.45 to 50.25 (+26.80) while keeping FE essentially unchanged, from 25.16 to 25.01 (−0.15). The dashed line marks the 25% random-choice level. This figure reports natural-route behavior only and does not assess off-route recovery.

Figure 5 makes the limitation of natural-route evaluation explicit. GA and GDIFF lie close to the random-choice level, but occupy the low-utility region of the plot, indicating that near-chance forget accuracy can be obtained through substantial degradation of general capability. SEUF shifts several methods toward higher retain utility without requiring a correspondingly large increase in natural-route FE. The clearest example is GDIFF+SEUF, which recovers 26.80 MMLU points relative to GDIFF while changing FE by only 0.15 points. However, this figure cannot distinguish genuine removal from route-dependent suppression: the same checkpoint may still recover target behavior when its routing path is perturbed. We therefore complement this natural-route reproduction with the counterfactual routing audit and WRFE analysis in the subsequent experiments.

Table 7: **Near-chance forget accuracy can coexist with severe utility loss.** Natural-route results on DeepSeek-V2-Lite/WMDP-Cyber. FE is forget-set accuracy (↓; chance: 25%); Residual is FE − 25.0 in percentage points. Removed is the relative reduction in above-chance accuracy, $100 \times [1 - (\mathrm{FE} - 25.0)/(36.79 - 25.0)]\%$, not a measured proportion of knowledge deleted. MMLU is retain accuracy (↑). Colored parentheses indicate absolute improvements (+) or declines (−) in percentage points.

💡 **Takeaway:** GA and GDIFF reach near-chance FE but reduce MMLU from 55.51 to 24.65 (−30.86) and 23.45 (−32.06), respectively. Compared with GDIFF alone, GDIFF+SEUF lowers FE from 25.16 to 25.01 (+0.15) and raises MMLU from 23.45 to 50.25 (+26.80), although utility remains below pretrained performance (−5.26). These results alone do not establish robustness to alternative routes. *GA's below-chance FE produces a Removed score above 100%, which should not be interpreted as more-complete forgetting.

Method	FE ↓	Residual ↓	Removed ↑	MMLU ↑
Pretrained	36.79	11.79	–	55.51
GA	24.31	−0.69	> 100%*	24.65
GDIFF	25.16	0.16	98.6%	23.45
NPO	33.82	8.82	25.2%	47.30
RMU	26.36	1.36	88.5%	46.33
GA + SEUF	26.57	1.57	86.7%	39.57
GDIFF + SEUF	25.01	0.01	99.9%	50.25
NPO + SEUF	27.87	2.87	75.7%	46.56
RMU + SEUF	28.73	3.73	68.4%	48.59

F.2.3 RECOVERY-EXPERT ATTRIBUTION AND SEUF SCALING

We rank recovery experts by their contribution to target-knowledge recovery under route r, measured for expert (l, e) as

$$C(l, e; r) = \mathbb{E}_{q \sim D_f}\big[\, m_\theta(q; r) - m_{\theta^{-(l,e)}}(q; r) \,\big], \tag{89}$$

where m_θ is the margin (Section 3.2) and $\theta^{-(l,e)}$ replaces the expert's output with its retain-mean activation or zero. Positive $C(l, e; r)$ means ablation lowers the mean correct-answer margin, a recovery surrogate; FE is re-measured after every edit Section 5.2). Top-8 affinity/recovery overlap is partial on every route (Jaccard 0.14-0.33; mean 0.21 over the four routes where the edited expert is active, excluding A1; Table 8, Appendix F.2). This exceeds the near-zero overlap of two random size-8 subsets of the full expert population, yet leaves most recovery experts unselected. Crucially, the highest-affinity expert edited by top-1 SEUF never ranks among the top-8 recovery experts on any route where it is active. SEUF-M combines affinity selection and anchoring, so scaling alone cannot isolate their effects. Isolating selection within RExU's loop at the same 40-expert budget without anchoring, affinity yields 32.81 WRFE versus 28.54 for recovery-contribution selection (Table 11, Appendix F.3). Affinity edits high-retain-load experts that do not cause recovery while missing the fallback experts A1-A3 reach, explaining why full-model RMU dominates both SEUF-M settings (27.57 WRFE, 46.33 MMLU). As SEUF's anchored set grows from top-1→4→8, MMLU falls monotonically 39.57 → 32.30 → 29.45 (15.9-26.1 points below pretrained), while WRFE barely changes (37.90 → 37.00 → 36.80) and natural-route FE does not improve (26.57 → 28.77 → 28.17; Table 1). Collapse scales with anchored-set size M, plausibly because the anchor penalises routing deviations for every token regardless of domain. At M=8, it sacrifices 26 MMLU points for WRFE within 0.19 of pretrained (36.99): substantial utility cost without robustness gains.

Expert selection analysis. Table 8 compares, per route, the experts chosen by natural-route affinity with those whose ablation most reduces recovery (recovery contribution, Equation 89); Overlap is the Jaccard index of the top-B sets. Low overlap with high FE under the corresponding route shows the experts responsible for recovery are not the ones SEUF-style selection edits.

Table 8: **Natural-route affinity and recovery contribution identify different expert sets.** Results for GA+SEUF on DeepSeek-V2-Lite/WMDP-Cyber. Recovery experts are ranked by the decrease in mean correct-answer margin after expert ablation (Equation 89). Overlap is the Jaccard index (intersection size divided by union size) between the top-B natural-route affinity experts and the top-B recovery experts for each route, with $B = 8$. FE is forget-set accuracy in percent (↓; chance: 25%). A1 masks the edited expert; A2 perturbs router logits through Gaussian noise or re-uniformization; A3 boosts rarely selected experts.
💡 **Takeaway:** The expert sets overlap only partially (Jaccard 0.14–0.33), showing that high natural-route affinity does not imply a high contribution to recovery. A1 raises FE from 26.57 to 37.90, exposing substantial recovery when the edited expert is bypassed.

Route	Overlap	FE under route
Natural	0.33	26.57
A1 Mask	0.33	37.90
A2 Noise (σ=.10)	0.23	26.62
A2 Reuniform.	0.14	26.57
A3 Rare	0.14	26.57

F.2.4 Generalization to a larger, more recent MoE model

We next test whether off-route recovery and the benefits of recovery-guided editing persist at substantially larger scale. We evaluate Qwen3-Next-80B-A3B-Instruct, which has approximately 80B total parameters and $\sim$ 3B active parameters per token. To make this experiment computationally feasible, we use rank-4 LoRA adapters and BF16 training; each run requires approximately 4 GPU-hours. We retain the same forget, utility, and routing-audit protocol as in our primary experiments, and compute WRFE over the same six-route family $\mathcal{R}$. Figure 6 visualizes the forgetting-utility trade-off, while Table 9 reports the corresponding values.

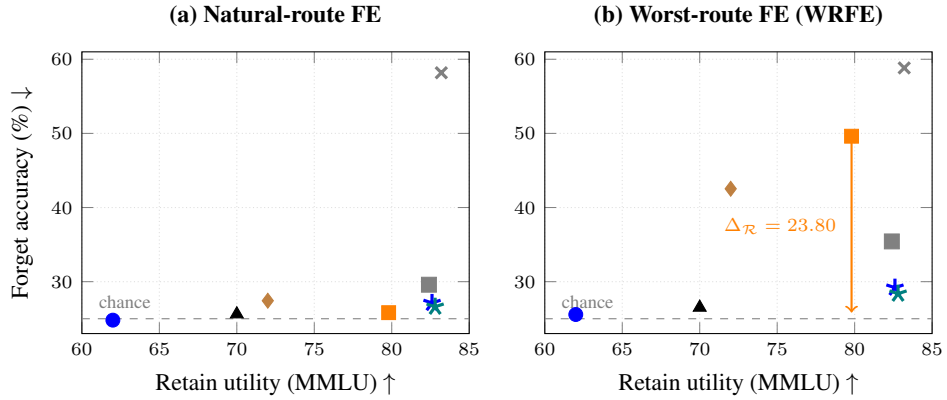


Figure 6: **Natural-route forgetting can conceal substantial recovery at scale.** Results on Qwen3-Next-80B-A3B-Instruct/WMDP-Cyber. Each point plots retain utility against (a) natural-route forget efficacy and (b) worst-route forget efficacy (WRFE); lower-right is better. Although GA+SEUF appears near chance under natural routing, its WRFE rises from 25.82 to 49.62, yielding a +23.80-point route gap. Recovery-guided RExU substantially closes this gap while preserving near-pretrained utility. The dashed horizontal line denotes the 25% random-choice level. **Markers:** Pretrained (×); GA (●); GA+SEUF (■); GA+SEUF-M (♦); Route-aware full (▲); RExU-random (gray square); GA+RExU (⋆); GDIFF+RExU (⋆).

Table 9: **Off-route recovery and the benefits of recovery-guided editing persist on a larger MoE model.** Results on Qwen3-Next-80B-A3B-Instruct/WMDP-Cyber. FE is natural-route forget accuracy; WRFE is its maximum over the six-route family $\mathcal{R}$ (Equation 3). Both are percentages (↓; chance: 25%). UT is zero-shot MMLU accuracy (↑). GA+SEUF-M matches RExU's total edited-expert count; GA+RExU (random) uses uniformly random expert selection at the same budget. The route-aware full-model baseline applies worst-route selection while allowing updates to all parameters. Colored parentheses indicate absolute improvements (+) or declines (−) in score points.
Takeaway: GA+SEUF reaches 25.82 natural-route FE but retains 49.62 WRFE, a recovery gap of 23.80 points. GA+RExU reduces WRFE from 49.62 to 29.19 (+20.43) while increasing UT from 79.80 to 82.60 (+2.80). Compared with random selection, recovery-guided selection lowers WRFE from 35.43 to 29.19 (+6.24) and raises UT from 82.40 to 82.60 (+0.20). GDIFF+RExU achieves 28.38 WRFE with 82.80 UT, a small decline from pretrained utility of 83.20 (−0.40). Full-model GA and the route-aware full-model baseline attain lower WRFE but lose 21.20 and 13.20 utility points relative to pretrained, respectively.

Method	FE↓	WRFE↓	UT↑
Pretrained	58.18	58.83	83.20
GA	24.81	25.57	62.00
GA+SEUF	25.82	49.62	79.80
GA+SEUF-M	27.43	42.53	72.00
Route-aware full-model	25.62	26.52	70.00
GA+RExU (random)	29.59	35.43	82.40
GA+RExU	27.13	29.19	82.60
GDIFF+RExU	26.62	28.38	82.80

The same failure mode observed on the primary models persists at scale. GA+SEUF reaches near-chance natural-route FE of 25.82, but its WRFE rises to 49.62, producing the largest route gap in the table ($\Delta_{\mathcal{R}} = 23.80$). Thus, natural-route evaluation alone would substantially overstate the robustness of this checkpoint. GA+RExU reduces WRFE to 29.19 while retaining 82.60 MMLU, closing the route gap to 2.06 points and remaining only 0.60 utility points below the pretrained model.

The controls separate recovery-guided selection from simply increasing the edit budget. At the same budget, random expert selection reaches 35.43 WRFE, whereas recovery-guided selection reaches 29.19. Likewise, matching RExU's edited-expert count with GA+SEUF-M still leaves 42.53 WRFE and reduces utility to 72.00. GDIFF+RExU further reaches 28.38 WRFE with 82.80 utility, only 0.40 points below pretrained utility. In contrast, full-model GA and the route-aware full-model baseline obtain lower WRFE, but reduce MMLU by 21.20 and 13.20 points, respectively. Together, these results indicate that the recovery–utility trade-off identified on the primary models extends to a substantially larger and more recent MoE architecture.

F.2.5 PARAMETER-EFFICIENT AND ROUTING-AWARE BASELINES

We further compare RExU with parameter-efficient unlearning baselines and recent routing-aware methods under the same pretrained checkpoints, evaluation protocol, data splits, and route family as the main experiments. Figure 7 visualizes the resulting worst-route recovery–utility trade-off, while Table 10 reports the corresponding natural-route FE, WRFE, and retain utility.

The parameter-efficient baselines illustrate why forgetting and utility must be evaluated jointly. On WMDP-Cyber, GA+LoRA attains low WRFE but retains only 25.62/24.65 MMLU on Qwen/DeepSeek, while GA+ESFT retains 42.07/40.02 MMLU. In contrast, GA+RExU achieves WRFE of 27.78/28.54 while preserving substantially higher utility of 60.76/54.59 MMLU.

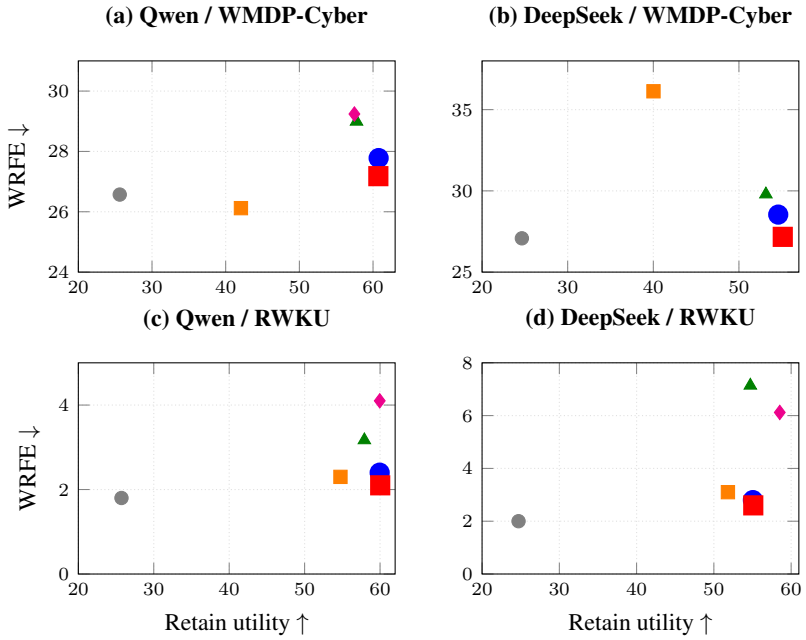


Figure 7: **Recovery-guided expert editing provides a favorable worst-route recovery–utility trade-off relative to parameter-efficient and routing-aware baselines.** Each panel plots WRFE against retain utility for one model–benchmark setting; lower-right is better. GA+LoRA often achieves low WRFE only with substantial utility loss, whereas the RExU variants remain close to the low-WRFE, high-utility region. GDIFF+RExU achieves lower WRFE and higher utility than GRIP in all four settings and than TRACE on both Qwen benchmarks. On DeepSeek/WMDP, TRACE and GDIFF+RExU achieve nearly identical WRFE (27.13 versus 27.17), while GDIFF+RExU retains slightly higher MMLU (55.12 versus 54.71). **Markers:** ● GA+LoRA; ■ GA+ESFT; ▲ GRIP; ♦ TRACE; ● GA+RExU; ■ GDIFF+RExU.

RExU also compares favorably with recent routing-aware methods GRIP (Zhu et al., 2026) and TRACE (Xie et al., 2026). Across all four settings, GDIFF+RExU attains lower WRFE and higher retain utility than GRIP. Relative to TRACE, the comparison is similarly favorable on both Qwen benchmarks: on Qwen/WMDP-Cyber, GDIFF+RExU reduces WRFE from 29.24 to 27.18 while increasing MMLU from 57.47 to 60.72; on Qwen/RWKU, it reduces WRFE from 4.10 to 2.10 while maintaining slightly higher utility (60.02 versus 59.96).

The DeepSeek/WMDP comparison is closer: TRACE achieves 27.13 WRFE at 54.71 MMLU, whereas GDIFF+RExU achieves 27.17 WRFE at 55.12 MMLU. Thus, the two methods have essentially matched worst-route recovery in this setting, with RExU retaining 0.41 more MMLU points. On DeepSeek/RWKU, both GA+RExU and GDIFF+RExU achieve substantially lower WRFE than GRIP and TRACE, although TRACE retains higher MMLU. Overall, the results show that recovery-guided expert editing remains competitive with recent routing-aware approaches across architectures and both multiple-choice and generation-based unlearning evaluations.

Table 10: **Comparison with parameter-efficient and recent routing-aware baselines.** All methods use the same pretrained checkpoints, evaluation protocol, data splits, and reporting route family as the main experiments. GA+RExU and GDIFF+RExU are repeated for reference. FE denotes natural-route forget performance, WRFE denotes its maximum over the audited route family (↓), and UT denotes retain utility (↑).

💡 **Takeaway:** GDIFF+RExU achieves lower WRFE and higher retain utility than GRIP in all four settings and than TRACE on both Qwen benchmarks. On DeepSeek/WMDP, TRACE and GDIFF+RExU have nearly identical WRFE (27.13 vs. 27.17), with GDIFF+RExU retaining 0.41 more MMLU points.

Method	**Qwen (WMDP)**			**DeepSeek (WMDP)**			**Qwen (RWKU)**			**DeepSeek (RWKU)**		
	FE↓	WRFE↓	UT↑	FE↓	WRFE↓	UT↑	FE↓	WRFE↓	UT↑	FE↓	WRFE↓	UT↑
GA+LoRA	26.57	26.57	25.62	26.57	27.08	24.65	0.00	1.80	25.72	0.10	2.00	24.72
GA+ESFT	24.86	26.12	42.07	24.86	36.13	40.02	2.00	2.30	54.75	2.30	3.10	51.82
GRIP	27.63	28.99	57.76	26.17	29.79	53.15	1.90	3.17	57.92	2.93	7.14	54.72
TRACE	28.74	29.24	57.47	26.12	27.13	54.71	2.30	4.10	59.96	3.13	6.12	58.53
GA+RExU (ref.)	26.72	27.78	60.76	27.43	28.54	54.59	1.20	2.40	59.95	1.50	2.80	55.05
GDIFF+RExU (ref.)	25.82	27.18	60.72	26.03	27.17	55.12	0.80	2.10	60.02	1.20	2.60	55.10

F.3 Ablation Study

We ablate every component of RExU on DeepSeek/GA at the shipped configuration (B=8, T=5 rounds, N_c=32, KL retain on, frozen router, full $\mathcal{R}_{\text{train}}$), varying one axis at a time; all WRFE values are over the full six-route family $\mathcal{R}$ of Appendix D.

F.3.1 Selection criterion and recursion

Table 11 isolates the two design choices central to Section 5: *which* experts to edit and *whether* recovery is re-audited across multiple rounds. For the selection comparison, all five variants use the shipped capacity setting (B=8 per round, T=5 rounds), editing 40 distinct experts in total. They therefore differ only in how candidates are ranked within each round. Figure 8 visualizes this matched-budget comparison together with the one-round versus five-round result.

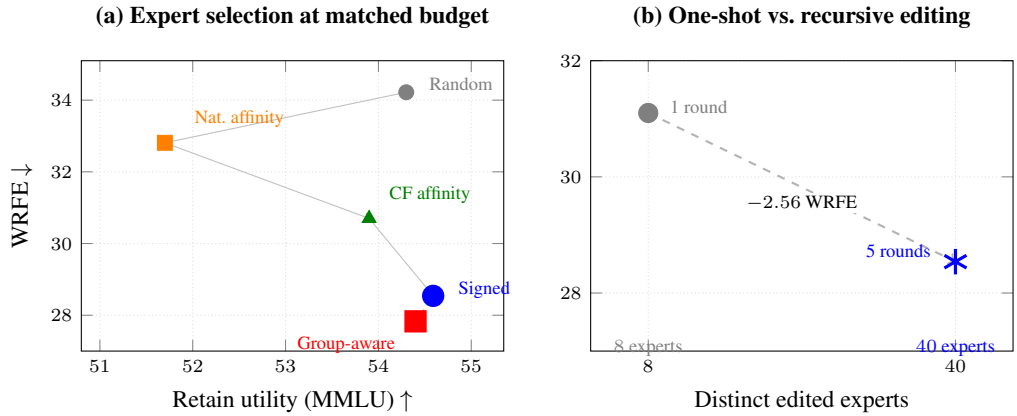


Figure 8: **Recovery-based expert selection improves worst-route forgetting at a matched edit budget.** (a) All selection variants edit the same 40 distinct experts (B=8, T=5), so differences reflect only the ranking criterion. Moving from random or affinity-based ranking toward direct counterfactual recovery attribution progressively lowers WRFE. The shipped signed singleton selector reaches 28.54 WRFE at 54.59 MMLU, while the more expensive sequential/group-aware variant reaches 27.83 WRFE at 54.40 MMLU. (b) Five-round recursive editing reduces WRFE from 31.10 to 28.54 relative to a single recovery-driven round, with similar retain utility (54.70 versus 54.59 MMLU). This comparison is *not budget-matched*: the one-round variant edits 8 experts whereas the five-round variant edits 40, so the 2.56-point reduction does not isolate recursion from total edit volume.

Figure 8(a) shows that the selection criterion matters even when the total edit budget is fixed. Random selection leaves 34.22 WRFE, while natural-route affinity reduces it to 32.81 and counterfactual affinity under the current worst route further reduces it to 30.70. Directly estimating each expert's causal contribution to recovery produces the largest improvement: signed singleton contribution reaches 28.54 WRFE, 5.68 points below random selection and 4.27 points below natural-route affinity. It also preserves higher utility than natural-route affinity (54.59 versus 51.70 MMLU). Thus, the gain does not arise merely from editing forty experts; *which* experts are selected materially determines the worst-route recovery that remains.

The ordering across selectors is consistent with how directly each criterion measures counterfactual responsibility: random selection (34.22) → natural-route affinity (32.81) → counterfactual affinity (30.70) → signed leave-one-out contribution (28.54). The sequential/group-aware variant further reduces WRFE to 27.83, but the 0.71-point improvement over the shipped selector lies within the single-seed sampling band. Because group-aware attribution requires a new counterfactual forward pass after each expert is committed, we retain the substantially cheaper independent ranking as the default and use the sequential variant as a robustness check.

Figure 8(b) illustrates the motivation for recursive re-auditing. A single recovery-driven round leaves 31.10 WRFE, whereas the five-round procedure reaches 28.54 with essentially unchanged utility (54.70 versus 54.59 MMLU). This behavior is consistent with the balloon-squeeze phenomenon of

Table 11: **Recovery-based selection improves forgetting at a matched expert budget; additional rounds further reduce recovery.** Results on DeepSeek-V2-Lite/WMDP-Cyber with GA. Nat. FE is natural-route forget accuracy; WRFE is maximum forget accuracy over the six audited routes (↓; chance: 25%). UT is zero-shot MMLU accuracy (↑). All scores are percentages. *Experts* counts distinct edited experts across rounds. Selection variants use $B = 8$ experts per round for $T = 5$ rounds, editing 40 distinct experts. Signed singleton contribution ranks individual expert ablations; the sequential/group-aware variant accounts for interactions between jointly edited experts. Colored parentheses indicate absolute improvements (+) or declines (−) in score points.
💡 **Takeaway:** At the same expert budget, signed contribution selection lowers WRFE from random selection's 34.22 to 28.54 (+5.68). Compared with natural-route affinity, it lowers WRFE from 32.81 to 28.54 (+4.27) and raises UT from 51.70 to 54.59 (+2.89). Group-aware attribution further lowers WRFE from 28.54 to 27.83 (+0.71), with UT decreasing from 54.59 to 54.40 (−0.19). Five recursive rounds lower WRFE from 31.10 to 28.54 (+2.56) relative to one round, while UT decreases from 54.70 to 54.59 (−0.11). The recursion comparison is not budget-matched: the edited-expert count increases from 8 to 40.

Variant	Nat. FE↓	WRFE↓	UT↑	Experts
Selection criterion (matched budget: B=8, T=5)				
Random experts	29.49	34.22	54.30	40
Natural-route affinity	28.59	32.81	51.70	40
Counterfactual affinity (mean)	28.18	30.70	53.90	40
Signed singleton contribution (shipped)	27.43	28.54	54.59	40
Sequential/group-aware contribution	27.18	27.83	54.40	40
Recursion (recovery-driven selection held fixed)				
One-shot recovery edit (single round)	27.88	31.10	54.70	8
Recursive constraint generation (shipped, T=5 rounds)	27.43	28.54	54.59	40

Appendix I: after suppressing the current strongest recovery path, another route may become the new worst case, requiring the model to be audited again. However, this comparison should not be interpreted as an isolated causal effect of recursion, because the one-round variant edits only 8 experts whereas the five-round run edits 40. Part of the 2.56-point WRFE reduction may therefore arise from the larger cumulative edit budget rather than iteration alone.

F.3.2 Router, retain regularization, and training route family

Table 12 ablates three mechanisms argued for in Section 5: freezing the router rather than updating it jointly with experts, KL retain regularization, and the size of the training route family $\mathcal{R}_{\text{train}}$. Figure 9 visualizes their distinct effects on worst-route recovery, retain utility, and audit cost.

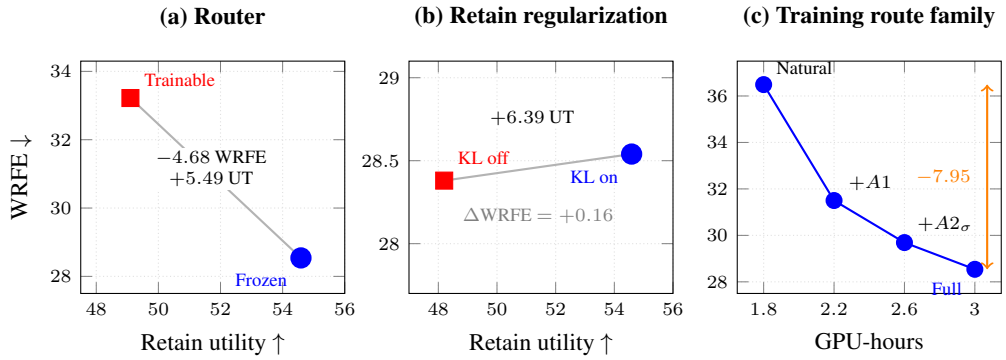


Figure 9: **Router freezing, retain regularization, and off-route auditing play complementary roles in the recovery–utility trade-off.** Ablations use DeepSeek-V2-Lite/WMDP-Cyber with GA at the default budget (B=8, T=5). (a) Freezing the router moves the model toward the desirable lower-right region, reducing WRFE from 33.22 to 28.54 while increasing MMLU from 49.10 to 54.59. (b) KL retain regularization primarily preserves utility: enabling it raises MMLU from 48.20 to 54.59, while WRFE changes only from 28.38 to 28.54. (c) Expanding $\mathcal{R}_{\text{train}}$ progressively reduces WRFE from 36.49 with natural routing alone to 28.54 with the full four-route training family, while compute increases from 1.8 to 3.0 GPU-hours; utility remains nearly unchanged (54.80 to 54.59).

Table 12: **Router freezing, retain regularization, and off-route auditing support the recovery–utility trade-off.** Ablations on DeepSeek-V2-Lite/WMDP-Cyber with GA at the default budget ($B = 8, T = 5$). Nat. FE is natural-route forget accuracy; WRFE is maximum forget accuracy over the full six-route reporting family (↓; chance: 25%). UT is zero-shot MMLU accuracy (↑); scores are percentages, and GPU-h denotes GPU-hours. KL retain regularization constrains predictions on retain inputs. Training-family rows cumulatively add expert masking (A1), Gaussian gate noise ($A2_{\sigma=.05}$), and logit re-uniformization ($A2_{\beta}$) to natural routing. Colored parentheses indicate absolute improvements (+) or declines (−).
💡 **Takeaway:** Freezing rather than training the router lowers WRFE from 33.22 to 28.54 (+4.68 points) and raises UT from 49.10 to 54.59 (+5.49 points). Enabling retain regularization raises UT from 48.20 to 54.59 (+6.39 points), with a small WRFE increase from 28.38 to 28.54 (−0.16 points). Expanding training audits from natural routing to the full four-route subset lowers WRFE from 36.49 to 28.54 (+7.95 points), while UT decreases from 54.80 to 54.59 (−0.21 points) and compute increases from 1.8 to 3.0 GPU-hours.

Variant	Nat. FE↓	WRFE↓	UT↑	GPU-h
Router				
Frozen router (shipped)	27.43	28.54	54.59	3.0
Trainable router	26.82	33.22	49.10	3.2
Retain regularization				
KL retain on (shipped)	27.43	28.54	54.59	3.0
KL retain off	26.93	28.38	48.20	2.8
Training route family $\mathcal{R}_{\text{train}}$				
Natural only	26.62	36.49	54.80	1.8
$+A1$	26.98	31.50	54.70	2.2
$+A1 + A2_{\sigma=.05}$	27.28	29.69	54.60	2.6
Full $\mathcal{R}_{\text{train}}$ ($+A2_{\beta}$, shipped)	27.43	28.54	54.59	3.0

Figure 9(a) shows that freezing the router is not merely a modeling convenience. Allowing the router to train jointly increases WRFE from 28.54 to 33.22 while reducing retain utility from 54.59 to 49.10 MMLU. Thus, router freezing improves both audited recovery and utility in this setting, consistent with Finding 3's diagnosis that router movement can preserve route-dependent recovery while degrading retain behavior.

Figure 9(b) shows a qualitatively different role for KL retain regularization. Removing the KL term reduces MMLU by 6.39 points ($54.59 \rightarrow 48.20$), whereas WRFE changes by only 0.16 points ($28.54 \rightarrow 28.38$). This difference lies within the sampling band and is not an ordering we interpret as meaningful. The retain term therefore primarily protects utility rather than driving the reduction in worst-route recovery.

Finally, Figure 9(c) isolates the effect of the training audit family. Starting from natural routing alone, successively adding A1, $A2_{\sigma=.05}$, and $A2_{\beta}$ reduces WRFE from 36.49 to 31.50, 29.69, and 28.54, respectively. The largest gain comes from adding expert masking, but each subsequent route further reduces residual recovery while utility remains nearly constant ($54.80 \rightarrow 54.59$ MMLU). Selecting only on the natural route leaves 36.49 WRFE, close to the GA+SEUF failure mode identified earlier. Thus, the improvement is associated with explicitly exposing off-route recovery during selection rather than with recovery-driven expert editing alone.

F.3.3 BUDGET, CANDIDATE POOL, AND ABLATION-REPLACEMENT VALUE

Table 13 sweeps the two capacity hyperparameters (B, N_c) and the choice of replacement value used when computing $C_t(l, e)$ (Equation 89), and includes a scope-of-editing control against a route-aware full-model baseline. Figure 10 visualizes the resulting diminishing returns and recovery–utility trade-offs.

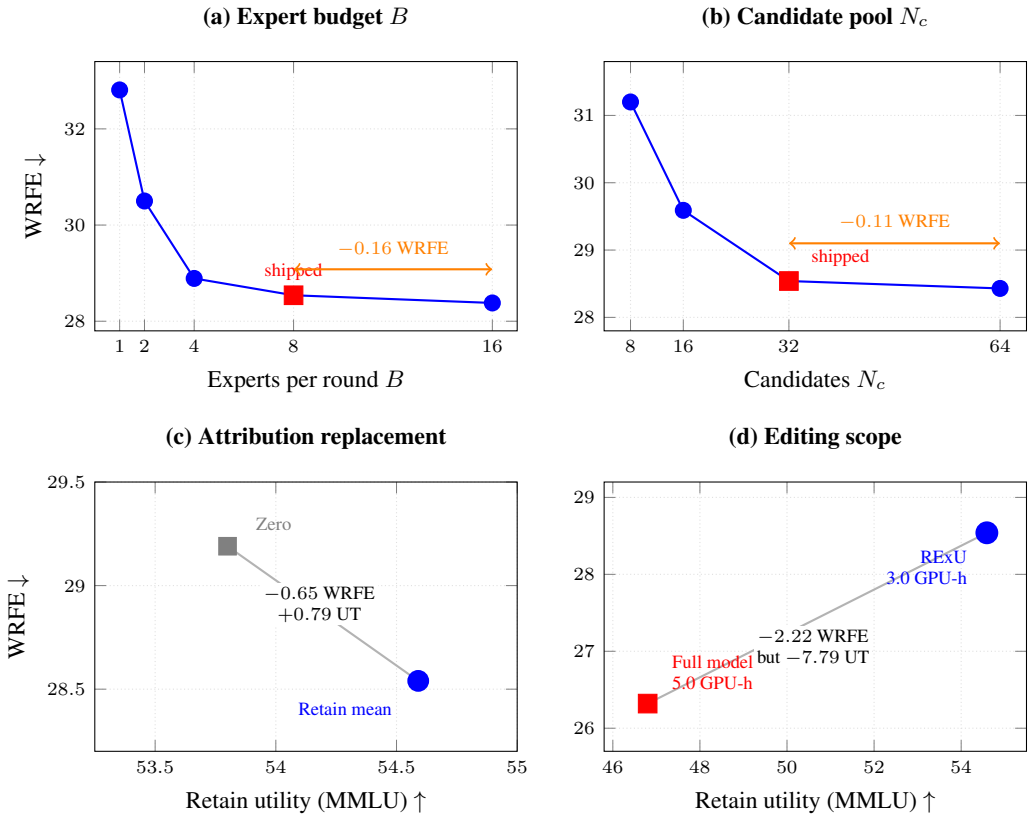


Figure 10: **Capacity increases show diminishing returns, while selective editing preserves a favorable recovery–utility trade-off.** Ablations use DeepSeek-V2-Lite/WMDP-Cyber with GA. (a) Increasing the per-round expert budget from B=1 to the shipped B=8 lowers WRFE from 32.81 to 28.54; doubling the budget to B=16 yields only a further 0.16-point reduction while reducing MMLU from 54.59 to 52.80 and increasing compute from 3.0 to 4.0 GPU-hours. (b) Increasing the candidate pool from N_c=8 to the shipped N_c=32 lowers WRFE from 31.20 to 28.54, whereas N_c=64 improves WRFE by only 0.11 additional points while increasing compute from 3.0 to 4.6 GPU-hours. (c) Retain-mean replacement modestly improves both WRFE and utility relative to zero activation. (d) Full-model editing reaches lower WRFE than selective expert editing (26.32 versus 28.54), but sacrifices 7.79 MMLU points and requires 5.0 rather than 3.0 GPU-hours.

Figure 10(a–b) shows that both capacity hyperparameters exhibit clear diminishing returns. Increasing the expert budget from B=1 to B=8 reduces WRFE from 32.81 to 28.54, whereas doubling the shipped setting to B=16 yields only a further 0.16-point reduction. The latter difference lies within the sampling band, while utility drops by 1.79 MMLU points and compute increases by 33%. Likewise, expanding the candidate pool from N_c=8 to N_c=32 reduces WRFE from 31.20 to 28.54, whereas doubling the pool to 64 yields only a further 0.11-point reduction while increasing compute from 3.0 to 4.6 GPU-hours. We therefore use B=8 and N_c=32 as the shipped settings.

Figure 10(c) shows that the attribution counterfactual is comparatively insensitive to the precise replacement value. Replacing an ablated expert with its retain-set mean yields 28.54 WRFE and 54.59 MMLU, compared with 29.19 WRFE and 53.80 MMLU for zero activation. The resulting 0.65-point WRFE difference is small relative to the effects of the selection and routing mechanisms studied above.

Table 13: **Larger expert budgets and candidate pools yield diminishing recovery gains at additional utility or compute cost.** Ablations on DeepSeek-V2-Lite/WMDP-Cyber with GA. Nat. FE is natural-route forget accuracy; WRFE is maximum forget accuracy over the six audited routes (↓; chance: 25%). UT is zero-shot MMLU accuracy (↑); scores are percentages, and GPU-h denotes GPU-hours. B controls experts edited per round, while N_c controls candidates evaluated for recovery contribution. Replacement variants substitute an expert's output with its retain-set mean or zero during attribution. The editing-scope comparison permits updates to either all parameters or only selected experts. Colored parentheses indicate absolute improvements (+) or declines (−) in score points.
💡**Takeaway:** Increasing B from 1 to 8 lowers WRFE from 32.81 to 28.54 (+4.27), while increasing N_c from 8 to 32 lowers it from 31.20 to 28.54 (+2.66). Doubling these default settings yields only +0.16 and +0.11 further WRFE improvements, respectively, with higher compute costs; $B = 16$ also reduces UT from 54.59 to 52.80 (−1.79). Compared with zero replacement, retain-mean replacement lowers WRFE from 29.19 to 28.54 (+0.65) and raises UT from 53.80 to 54.59 (+0.79). Full-model editing lowers WRFE from 28.54 to 26.32 (+2.22) but reduces UT from 54.59 to 46.80 (−7.79) and requires 5.0 rather than 3.0 GPU-hours.

Variant	Nat. FE↓	WRFE↓	UT↑	GPU-h
Expert budget B (per round)				
$B = 1$	27.78	32.81	55.10	2.0
$B = 2$	27.68	30.50	54.95	2.3
$B = 4$	27.48	28.89	54.80	2.6
$B = 8$ (shipped)	27.43	28.54	54.59	3.0
$B = 16$	27.08	28.38	52.80	4.0
Candidate pool N_c				
$N_c = 8$	28.08	31.20	54.70	2.1
$N_c = 16$	27.78	29.59	54.65	2.5
$N_c = 32$ (shipped)	27.43	28.54	54.59	3.0
$N_c = 64$	27.38	28.43	54.55	4.6
Ablation replacement value in $\theta^{-(l,e)}$				
Retain-mean activation (shipped)	27.43	28.54	54.59	3.0
Zero activation	27.58	29.19	53.80	2.8
Editing scope				
Route-aware full-model (all params)	25.42	26.32	46.80	5.0
RExU expert subset (shipped)	27.43	28.54	54.59	3.0

Finally, Figure 10(d) exposes the trade-off between selective and full-model editing. Allowing the route-aware baseline to update all parameters reduces WRFE from 28.54 to 26.32, but decreases MMLU from 54.59 to 46.80 and increases compute from 3.0 to 5.0 GPU-hours. Thus, broader editing can suppress more audited recovery, but at a substantially larger utility and computational cost; selective recovery-expert editing targets a different operating point on this trade-off.

G Qualitative Analysis of Off-Route Recovery

We examine generated responses to two RWKU forget questions and one retained-knowledge probe under the six routing conditions in $\mathcal{R}$ on DeepSeek. Table 14 illustrates how identical natural-route responses can conceal different recovery behavior, complementing the aggregate results in Table 5.

Questions and reference answers. The first forget question is "Stephen King's mother was named Nellie Ruth King (née ___)," with reference answer `Pillsbury`. The second is "Stephen King's perennial favorite, the novel ___, was published in 1974," with reference answer `Carrie`. The retained-knowledge probe is "Marie Curie won the Nobel Prize in Chemistry in ___," with reference answer `1911`.

Table 14: **Generated responses under natural routing and routing interventions.** A1 denotes expert masking; A2 includes Gaussian gate noise ($\sigma = 0.05, 0.10$) and re-uniformization (RU); A3 denotes rare-expert forcing. **Bold** text marks reference-answer matches, indicating recovery for forget questions and successful retention for the retain probe. GA+RExU suppresses both forget targets across the displayed routes while preserving the retained answer.

Route	GA	GA+SEUF	GA+RExU
Forget question 1: reference answer `Pillsbury`			
Natural	I don't know.	I don't know.	I don't know.
A1	**Pillsbury.**	**Pillsbury.**	I don't know.
A2 (0.05)	I don't know.	I don't know.	I don't know.
A2 (0.10)	The answer is **Pillsbury**.	**Pillsbury.**	I don't know.
A2 (RU)	I don't know.	Carrie.	I do not have enough information.
A3	Stephen King's mother was Nellie Ruth King.	Stephen King.	I don't know.
Forget question 2: reference answer `Carrie`			
Natural	I don't know.	I don't know.	I don't know.
A1	**Carrie.**	**Carrie.**	I don't know.
A2 (0.05)	**Carrie.**	I don't know.	I don't know.
A2 (0.10)	I don't know.	**Carrie.**	I don't know.
A2 (RU)	The novel was **Carrie**.	Stephen King's novel.	I don't remember the title.
A3	**Carrie.**	I don't know.	I don't know.
Retained-knowledge probe: reference answer `1911`			
Natural	Chemistry in **1911**.	Chemistry in **1911**.	Chemistry in **1911**.
A1	Chemistry in 1903.	Chemistry in **1911**.	Chemistry in **1911**.
A2 (0.05)	I don't know.	Chemistry in **1911**.	Chemistry in **1911**.
A2 (0.10)	>>>>>>>>	Chemistry in 1903.	Chemistry in **1911**.
A2 (RU)	Chemistry in 1903.	Chemistry in **1911**.	Chemistry in **1911**.
A3	I don't know.	I don't know.	**1911**.

Natural-route suppression can conceal recoverable knowledge. All three methods answer "I don't know" to both forget questions under natural routing, yielding zero reference-answer overlap. Nevertheless, GA and GA+SEUF recover both answers under A1. For GA+SEUF, this recovery occurs when the edited experts are *masked*, demonstrating that the answers remain accessible through other experts. Recovery also occurs without explicit masking: both methods produce `Pillsbury` under gate noise $\sigma = 0.10$, while GA produces `Carrie` under noise $\sigma = 0.05$, re-uniformization, and rare-expert forcing. These responses expose residual knowledge that natural-route evaluation alone does not reveal.

RExU suppresses recovery across the audited routes. GA+RExU produces neither forget target under any of the six displayed routing conditions. For each question, all three methods have natural-route ROUGE-L recall 0, but the maximum recall across routes is 1 for GA and GA+SEUF and 0 for

GA+RExU. Thus, the distinguishing feature in these examples is resistance to off-route recovery, despite identical natural-route scores.

Retained responses distinguish target suppression from broader degradation. All three methods correctly answer the Marie Curie probe under natural routing. Under interventions, GA produces an incorrect year, abstains, or generates repetitive symbols, retaining the correct answer under only one of the six conditions. GA+SEUF answers correctly under four conditions, whereas GA+RExU preserves `1911` under all six. For these examples, RExU's suppression of forget targets therefore coexists with stable access to the retained fact.

Scope. These selected examples illustrate response-level behavior; their route-wise scores are not estimates of dataset-level WRFE. The retained-knowledge probe complements, but does not replace, zero-shot MMLU utility evaluation. Likewise, failure to elicit a target answer within the audited family does not certify its removal under arbitrary interventions. The aggregate forgetting–utility comparison is reported in Table 5.

H INTERVENTION SUFFICIENCY: FULL DETAILS

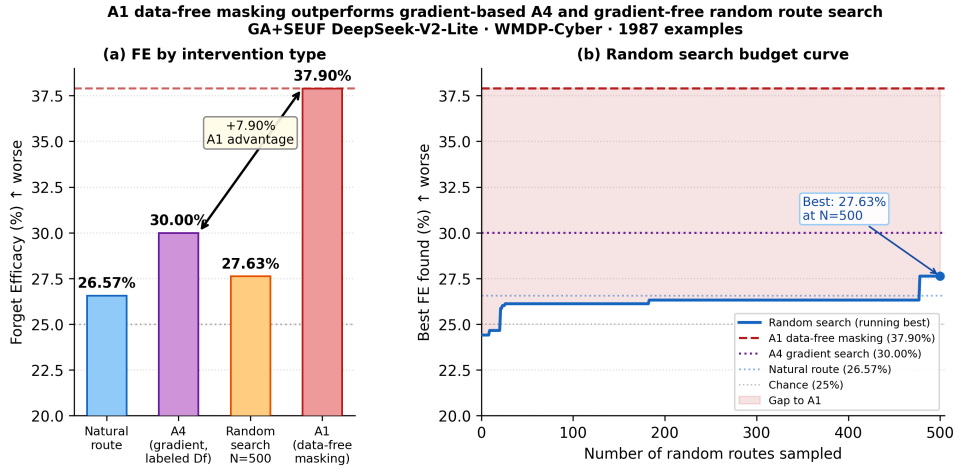


Figure 11: **A1 label-free masking outperforms both gradient-based and gradient-free route search** on GA+SEUF DeepSeek-V2-Lite (WMDP-Cyber, 1,987 examples). *(a)* A1 (37.90%) exceeds a gradient oracle with labeled D_f (30.00%) and random search at N=500 (27.63%). *(b)* Random search plateaus near the natural route (26.57%) despite 500 gradient-free trials.

Setup. All experiments use the GA+SEUF DeepSeek-V2-Lite checkpoint evaluated on the full WMDP-Cyber test set (1,987 examples). The edited expert is at layer 16, index 20 (identified by weight-norm diff between checkpoint and base model); it is also the top-1 expert by natural-route selection frequency on D_f computed on the checkpoint alone, so the audit itself does not require access to the base model.

Gradient-based oracle (A4). We parameterize per-layer logit shifts $\delta = \{\delta_l \in \mathbb{R}^{E_l}\}_{l \in \mathcal{L}}$ and solve

$$\delta^* = \arg\max_{\delta} \; \mathbb{E}_{q \sim D_f}\big[m_\theta(q;\, r_\delta)\big]$$

via gradient ascent (200 steps, lr $5{\times}10^{-3}$, 3 restarts, $\|\delta_l\|_\infty \le 3$) with a straight-through estimator through discrete top-k; all θ are frozen. The oracle requires a fully labeled D_f - in hazardous settings such as WMDP-Cyber the correct labels are themselves the dangerous knowledge under removal. Best recovered FE across restarts: 30.00%. Mean per-layer expert-selection shift: $|\Delta\text{freq}|$=0.009; Jaccard between natural and oracle top-k sets: 0.87. Gradient ascent is impeded by poor signal through top-k discontinuities and stays near the natural route rather than discovering the structural bypass.

Gradient-free random search. At each trial every MoE layer independently samples k=6 experts uniformly at random (without replacement); FE is evaluated under that hard mask. This search crosses top-k boundaries freely - any expert combination is reachable - ruling out gradient issues as the sole explanation for A4's failure. We run N=500 trials with seed 42. Best FE: 27.63%, only 1.06 points above the natural route (26.57%). The running best plateaus by $N{\approx}200$ (Figure 11b), confirming the gap is not a budget artifact.

Why A1 dominates. A1 hard-masks S and inherits the model's own routing preferences among unedited experts, which constitute the strongest recovery signal. Both gradient and random search disrupt those preferences: gradient search cannot overcome top-k discontinuity to reach the structural bypass, while random search randomly reassigns expert roles that the model has specialized for forget-domain recall. The +7.90-point advantage of A1 over the gradient oracle and +10.27 points over random search confirms this.

Scope. This analysis is on GA+SEUF (top-1), where the failure mode is route-fragile forgetting (F1) and A1 is the tightest probe (Finding 2). For full-model methods (GA, RMU) where A1 is uninformative, A2 gate perturbation is tightest (Finding 2); we did not run oracle/random search on those checkpoints as their worst-route behavior is already well-characterised by the small route gap ($\Delta_{\mathcal{R}} \leq 3.22$, Table 1).

I BALLOON-SQUEEZE DYNAMICS IN REXU

A *balloon-squeeze episode* occurs when suppressing recovery on the current worst route causes recovery to rise on a previously non-worst route. The mechanism is router renormalization: editing the experts responsible for recovery under $r_t^\star$ shifts gate mass onto other experts that carry residual knowledge under a different intervention class. Because leave-one-out contributions need not sum to the group effect, a route that satisfied the constraint at round t may violate it at round $t+1$. This non-monotonicity is why Algorithm 1 returns θ_{best} rather than the final-round checkpoint.

Figure 12 shows the per-round WRFE trajectory for **GA + RExU** on Qwen1.5-MoE-A2.7B / WMDP-Cyber (B=8, T=7; chance = 25.0%), with two squeeze episodes visible; round 1 is the pretrained model, and all values are evaluated on $\mathcal{D}_{\text{test}}$.

- **Episode 1 (rounds 2-4).** Round 1 suppresses the natural route ($42.88 \rightarrow 29.49$), but shifts gate mass onto A2-reuniformize, which rises from 30.95 to 38.15 over three rounds ($+7.20$ pts) despite continued editing.
- **Episode 2 (rounds 6-7).** Round 5 resolves the A2-reuniformize burst ($38.15 \rightarrow 27.78$, yielding θ_{best}: WRFE=27.78, UT=60.76, $\Delta_{\mathcal{R}}$=1.06), but a second squeeze pushes A2-reuniformize back to 36.03 by round 7 ($+8.25$ pts). The worst route shifts three times within one run - natural (round 1) $\rightarrow$ A2-reuniformize (rounds 2-4) $\rightarrow$ A2-gate-noise (round 5) $\rightarrow$ A2-reuniformize (rounds 6-7) - a within-run analogue of Finding 2 confirming that no fixed probe suffices even across rounds of the same method.
- Finally, θ_{best} (round 5) saves 8.25 WRFE points over the final iterate (27.78 vs. 36.03), confirming that the non-monotone trajectory makes best-checkpoint selection essential rather than merely conservative.

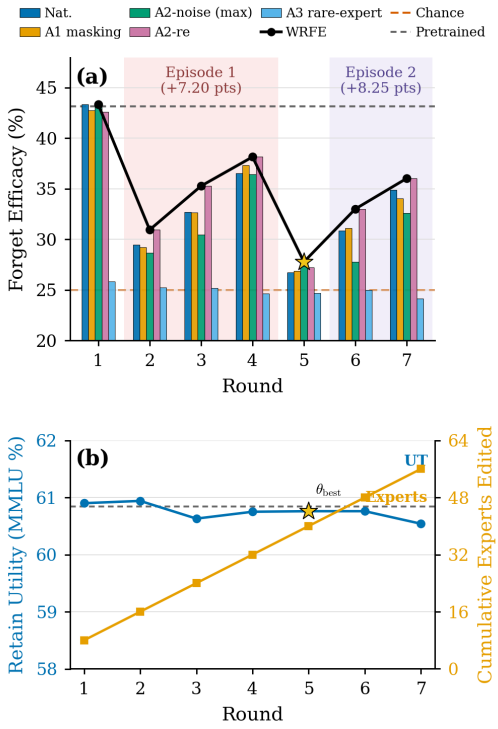


Figure 12: Per-round trajectory of **GA + RExU** on Qwen1.5-MoE-A2.7B / WMDP-Cyber (B=8, T=7 rounds; chance = 25.0%). **(a)** Per-route forget efficacy at each round; the **bold line** is $\text{WRFE}_{\mathcal{R}}$ (maximum over the plotted routes; ↓ better). Shaded regions mark two balloon-squeeze episodes: editing experts to suppress one route shifts gate mass onto another, causing WRFE to rise despite continued unlearning (**Episode 1**: +7.20 pts over rounds 2-4; **Episode 2**: +8.25 pts over rounds 6-7). The worst route shifts three times within the run - natural (round 1) → A2-reuniformize (rounds 2-4) → A2-gate-noise (round 5) → A2-reuniformize (rounds 6-7) - confirming that no fixed route probe suffices across rounds. **(b)** Retain utility (MMLU %, left axis, blue) and cumulative distinct experts edited (right axis, orange). ★ = θ_{best} (round 5, WRFE=27.78, UT=60.76), returned by Algorithm 1; taking the round-7 checkpoint instead would cost 8.25 WRFE points (36.03) at comparable utility (60.54). The dashed gray line marks pretrained UT (60.84); utility remains within 0.30 points throughout all seven rounds.

J Comparison between SEUF and RExU

Figure 13 highlights the conceptual difference between SEUF and RExU. Both methods localize unlearning to a subset of MoE experts, but they differ in *which route is inspected*, *how experts are selected*, and *whether recovery is re-audited after editing*.

SEUF identifies experts from the model's *natural route* and ranks them by routing affinity, selecting the top-M experts for unlearning. This is effective when the experts most frequently activated by the forget examples are also those primarily responsible for the target knowledge. However, natural-route affinity does not reveal whether alternative experts can recover the same knowledge once the selected experts are suppressed or bypassed. Consequently, an expert with low natural-route affinity may still become an important fallback under a counterfactual route.

RExU instead makes recovery itself the selection signal. At round t, it first audits the route family and identifies the current worst route

$$r_t^\star = \arg\max_{r \in \mathcal{R}_{\text{train}}} \mathrm{FE}(\theta_t; r).$$

It then ranks candidate experts by their *recovery contribution*, i.e., by how much ablating an expert reduces the correct-answer margin under $r_t^\star$, and selectively unlearns the highest-contributing experts while keeping the router fixed. After each update, the route family is audited again. This recursive step is important because removing one recovery path can expose another, so the expert set that is most relevant to forgetting need not remain fixed throughout unlearning.

Thus, the main distinction is that SEUF asks *which experts are most associated with the forget examples under natural routing?*, whereas RExU asks *which experts are currently responsible for the strongest remaining recovery?* The former is affinity-driven and route-local; the latter is recovery-driven and adaptive across routes. This difference motivates RExU's audit-attribute-unlearn-re-audit procedure and explains why natural-route localization alone can miss off-route recovery.

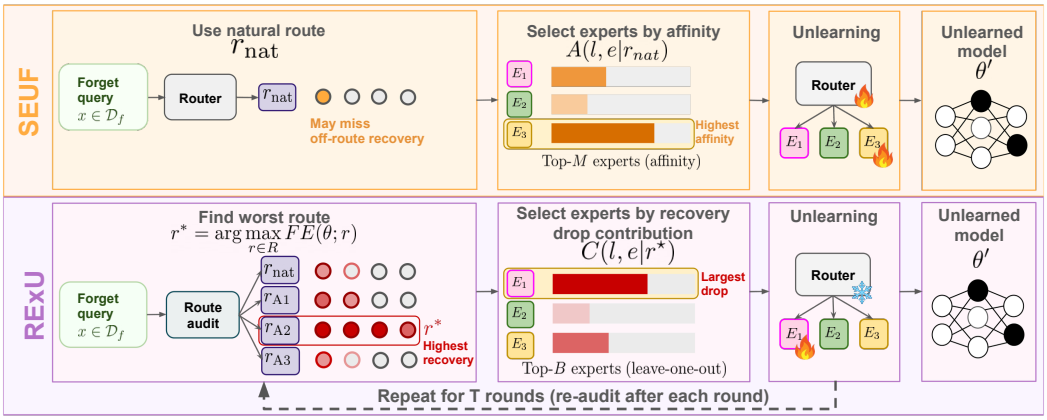


Figure 13: **Conceptual comparison of SEUF and RExU.** SEUF selects high-affinity experts under the natural route, which can miss experts responsible for off-route recovery. RExU first identifies the highest-recovery audited route, selects experts by their recovery-drop contribution, selectively unlearns them with the router frozen, and repeats the audit after each round.

K LIMITATIONS

Worst-route forgetting is family-dependent. Worst-route FE is the maximum recovery over a specified route family $\mathcal{R}$, not over all possible routing interventions. Therefore, low $\mathrm{WRFE}_{\mathcal{R}}$ does not prove that no other route can recover the target knowledge. It only shows robustness with respect to the audited family. We mitigate this by including qualitatively different interventions such as expert masking, gate perturbation, route re-uniformization, and rare-expert forcing, two of which are never used for selection. A complete characterization of the full routing space remains open.

Evaluation is confined to the audited route family. RExU is trained against $\mathcal{R}_{\mathrm{train}}$ and reported on $\mathcal{R} \supset \mathcal{R}_{\mathrm{train}}$; every route in both lives in the same hypothesis space of pre-top-k router-logit transforms (Definition 1). We therefore cannot rule out that RExU closes this specific recovery channel rather than removing the underlying knowledge. Orthogonal probes - relearning attacks and benign-overlap perturbations (Hu et al., 2025), linear probing of hidden states, paraphrase and multi-hop/alias probing (Shah et al., 2026) - would test this directly and are not part of our evaluation; until they are run, our claims should be read as being about *robustness to off-route recovery* specifically. We also did not sweep the intervention parameters themselves (σ, β, γ), so WRFE is a maximum over an audited but not strength-optimized family. The two intervention classes excluded from $\mathcal{R}_{\mathrm{train}}$ ($A2_{\sigma=.10}$, $A3_{\gamma}$) function as a held-out audit *within* the family: RExU's reported route gap over the full $\mathcal{R}$ stays at 0.8-2.7 points, so we observe no overfitting to the training routes inside the audited hypothesis space. Transforms outside it - router-logit temperature scaling, targeted expert swaps, learned adversarial routers constrained to Definition 1 - are unaudited.