ROUTING CAN HIDE WHAT UNLEARNING DID NOT REMOVE: A FORCED-ROUTING PROBE FOR MIXTURE-OF-EXPERTS

Anonymous authors
Paper under double-blind review

ABSTRACT

In Mixture-of-Experts (MoE) language models, unlearning can lower forget-domain accuracy in two ways: by removing knowledge stored in expert parameters, or by changing which experts the router activates. Evaluation under the model's own routing conflates the two, scoring a removed capability and one the router merely hides alike. We introduce a **forced-routing evaluation framework** that separates them by also reading the forget domain through the experts that natural routing bypasses. With it, we show that natural-routing metrics can substantially overestimate unlearning, as capabilities that appear forgotten can remain recoverable through alternative expert paths. We further propose a **recoverability-based expert selection criterion**, which chooses experts by how much target capability remains recoverable through them rather than by how often or how strongly the router selects them. The two criteria select largely disjoint experts. Under representation-based unlearning, recoverability-based selection leaves less capability recoverable than SEUF's affinity-based selection at matched forgetting depth, with comparable retain utility. Unlearning in sparse models should therefore be evaluated by what the parameters retain, not only by what the default computation path exposes. Anonymous code is available at https://anonymous.4open.science/r/ForcedRoutingProbe.

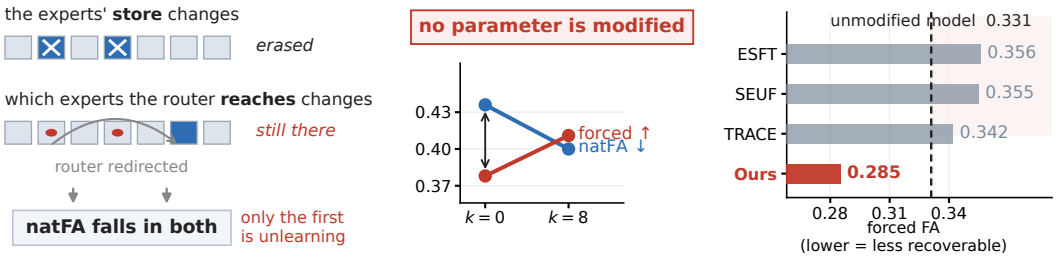
1 INTRODUCTION

Mixture-of-Experts (MoE) architectures have become a prominent way to scale language models by activating only a small fraction of parameters for each token (Jiang et al., 2024; Dai et al., 2024; Muennighoff et al., 2025). Like dense language models, MoEs can memorize and reproduce private, copyrighted, or hazardous information (Jang et al., 2022; Patil et al., 2023; Li et al., 2024). Machine unlearning aims to remove such target knowledge while preserving the model's remaining capabilities (Liu et al., 2025b). Recent MoE-specific approaches exploit sparsity by identifying a small subset of experts associated with the forget data and updating only those experts (Zhuang et al., 2025; Xie et al., 2026).

However, sparsity introduces a fundamental ambiguity into both training and evaluation. A decrease in forget-domain performance can arise because target knowledge has been removed from expert parameters, or because the router has shifted computation away from experts that still retain it. Natural-routing evaluation conflates these two effects: it measures not only *what the model retains*, but also *what its default computation path exposes*. This is particularly problematic for MoE unlearning, where existing methods are evaluated under the model's own routing policy and expert selection itself is typically based on routing statistics such as gate scores or activation frequencies. A model may therefore appear to forget while the underlying capability remains accessible through experts that natural routing no longer favors.

Therefore, building on the expert forcing of Zhu et al. (2026), we introduce a **forced-routing evaluation** to separate these effects. At the layers modified by unlearning, we bypass the experts selected by the natural router and instead activate high-scoring experts from the remaining routing tail. This provides a complementary reading of the same trained model: natural routing measures the capability exposed by the default path, whereas forced routing probes capability that remains accessible

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) Two sources of lower natFA **(b)** Routing-only control **(c)** Forced-routing comparison

Figure 1: **Natural vs. forced-routing evaluation.** Routing changes can mimic forgetting under natural evaluation, while forced routing exposes residual capability.

outside that path. A routing-only intervention on the unmodified model validates the distinction: changing which experts are exposed, without modifying any expert parameter, changes the standard forget metric. As shown in Figure 1, after unlearning, the same discrepancy remains—models that appear similarly forgotten under natural routing can retain substantially different amounts of capability under forced routing, and the two evaluations can induce opposite orderings over the same trained models.

This perspective also suggests a different principle for MoE unlearning. Existing methods select experts according to *routing affinity*, which measures where the router prefers to send forget-domain inputs. We instead ask where the target capability remains recoverable. Our **recoverability-based selection** explicitly routes through each candidate expert and scores how much forget capability remains accessible through it, yielding an interventional measure of retention rather than an observational measure of routing preference. The resulting expert sets differ sharply from those selected by affinity-based criteria. Under representation-based unlearning, our experiments show that recoverability-based selection leaves less residual capability accessible under forced routing at matched forgetting depth while preserving comparable retain utility; under a likelihood-based objective the ordering does not reproduce (§5.3). Together, these results show that routing is not merely an implementation detail of sparse unlearning: it can change both how forgetting is measured and which experts are chosen to forget. They motivate a shift from *routing preference* to *capability recoverability* when auditing and targeting knowledge in conditionally activated models.

Our contributions are threefold:

- **We isolate a routing-induced ambiguity in how MoE unlearning is measured.** Router shortcuts during training are known (Zhuang et al., 2025; Zhu et al., 2026); on an unmodified model, natural-routing forget accuracy falls with no expert parameter touched, showing that apparent forgetting and parameter-level forgetting need not coincide.
- **We turn expert forcing into an evaluation protocol for MoE models.** Applied to every item and read beside natural routing, it probes capability outside the model's preferred computation path, exposing residual knowledge that standard natural-routing evaluation can miss, and can reverse the apparent ordering of unlearning methods.
- **We propose recoverability-based expert selection.** Rather than selecting experts by routing affinity, we target experts according to the capability they retain, yielding stronger removal of recoverable capability at matched forgetting depth under representation-based unlearning.

2 Related Work

Unlearning in large language models, and expert selection for MoE. Machine unlearning was formulated for classical models as the removal of a training sample's influence (Cao & Yang, 2015; Bourtoule et al., 2021) and for LLMs as the removal of a capability (Liu et al., 2025b). Targeted forgetting methods fall into a few families: gradient ascent and its regularised variants (Jang et al., 2022; Yao et al., 2023), preference-style objectives such as NPO (Zhang et al., 2024), weight-arithmetic edits (Ilharco et al., 2022; Eldan & Russinovich, 2023), and representation-space objectives such as RMU (Li et al., 2024), scored along a forget–utility trade-off on TOFU (Maini et al., 2024), MUSE (Shi et al., 2025) and WMDP (Li et al., 2024). Both the methods and those metrics were

developed for dense transformers, where one forward pass exercises the whole network. Methods designed for MoE architectures add one step, picking a small set of experts and updating only those, and differ in how the set is chosen. SEUF (Zhuang et al., 2025) scores an expert by its mean gate score averaged over all tokens and counted as zero where it is not selected. It anchors the router because it finds routing drifting to non-target experts during unlearning, lowering the forget loss while the intended experts go unmodified, a shortcut it names *fake unlearning*. That anchor acts during training; the reading that follows is taken under whatever routing the trained model ships. ESFT (Wang et al., 2024) uses the mean gate score over all tokens. TRACE (Xie et al., 2026) uses the hard selection frequency on forget data, taking the upper outliers of the pooled distribution. All three estimate *how strongly the router prefers an expert on forget data* and select largely overlapping sets. We instead score an expert by whether the forget target remains answerable when the model is forced through that expert alone, which reads a largely disjoint region of the same distribution rather than re-estimating the same quantity. Each of these methods, and the numbers each reports, is read under natural routing.

Does unlearning remove or merely obfuscate? Apparent forgetting in dense models is often recoverable. Łucki et al. (2025) restore WMDP capability without modifying any weights, by orthogonalising residual-stream directions and by isolating the weights that carry the entire unlearning effect. Parallel work agrees from other directions: extracted answers survive objectives designed to suppress them (Patil et al., 2023), and the evaluation protocol alone changes whether a model looks unlearned (Lynch et al., 2024). Capability returns under light fine-tuning (Deeb & Roger, 2024; Qi et al., 2023; Lermen et al., 2023) unless suppression is trained adversarially (Sheshadri et al., 2024), and the same gap is documented for concept erasure and directional interventions (Belrose et al., 2023; Zou et al., 2023; Arditi et al., 2024; Meng et al., 2022). *All of these recovery channels live in representation space and require white-box optimisation or additional training.* Mixture-of-Experts models expose a further, *architectural* channel, the router, which an adversary exercises at inference with neither optimisation nor data. Robustness to prompts does not close it: Zhuang et al. (2025) report that a 5,000-step GCG attack recovers nothing from their unlearned models, yet under expert forcing Zhu et al. (2026) find the same method's forget accuracy on WMDP rising from 0.25 to 0.33. Their forcing is a post-hoc check accompanying a main result, in two variants: the top five non-selected experts on forget queries whose routing shifted, and, in an appendix protocol, the pre-unlearning path. Our probe is closest to the former, but is applied to every item rather than only to those whose routing shifted, so it needs neither the original weights nor any shift in routing; we also read it against the unmodified model under the same probe and promote it to the primary evaluation criterion, with a comparison protocol built around it.

3 Preliminaries

3.1 Mixture-of-Experts routing

We study the sparsely gated mixture-of-experts layer (Shazeer et al., 2017; Lepikhin et al., 2020; Fedus et al., 2022) as it appears in deployed open-weight models (Jiang et al., 2024; Muennighoff et al., 2025; Yang et al., 2024; Dai et al., 2024): decoder-only, with each MoE layer ℓ holding N *routed* experts $\{E_1^{(\ell)}, \dots, E_N^{(\ell)}\}$ and n_s *shared* experts. A gate $W_g^{(\ell)} \in \mathbb{R}^{N \times d}$ scores the routed experts for each token representation $h \in \mathbb{R}^d$ as $s^{(\ell)}(h) = \sigma(W_g^{(\ell)} h) \in \mathbb{R}^N$, with σ applied element-wise. The scores are *not* normalised across experts: this architecture uses a per-expert sigmoid gate with auxiliary-loss-free load balancing (DeepSeek-AI, 2024; 2025). The router selects the top-k set $\mathcal{S}_{\text{nat}}^{(\ell)}(h)$ using load-balance-corrected scores, and the layer output is

$$y^{(\ell)}(h) \;=\; \gamma \sum_{j \in \mathcal{S}^{(\ell)}(h)} \hat{w}_j \, E_j^{(\ell)}(h) \;+\; \sum_{i=1}^{n_s} E_i^{(\ell),\text{sh}}(h), \qquad \hat{w}_j = \frac{s_j^{(\ell)}(h)}{\sum_{j' \in \mathcal{S}^{(\ell)}(h)} s_{j'}^{(\ell)}(h)}, \tag{1}$$

where γ rescales the routed term relative to the shared one and is not fitted by us: it is the `routed_scaling_factor` field of the released configuration, $\gamma = 2.446$ for the model we use. Two properties of Eq. 1 drive everything that follows. **The routed term is *gated***, so which experts contribute is a function of $s^{(\ell)}$ and is modifiable without touching any parameter. **The shared term is *not* gated**, so those experts contribute to every token regardless of routing, and no routing policy can remove them and no routing-based selection rule can reach them.

3.2 UNLEARNING OBJECTIVES

An *arm* is one training configuration in which the selection signal is the only thing that varies. We distinguish two objective families, because the family determines *where* the forget term is defined and therefore how an update is distributed over the experts it may touch. *Representation family.* RMU (Li et al., 2024) defines it in activation space, on the residual-stream representation $a_{\ell^*}(x)$ at a designated layer ℓ^* against a frozen copy $\tilde{a}_{\ell^*}$, with u a fixed random unit vector and c a steering magnitude:

$$\mathcal{L}_{\text{RMU}} = \big\|a_{\ell^*}(x_f) - c\,u\big\|_2^2 + \alpha\big\|a_{\ell^*}(x_r) - \tilde{a}_{\ell^*}(x_r)\big\|_2^2, \tag{2}$$

with x_f and x_r drawn from a forget and a retain corpus. *Likelihood family.* NPO (Zhang et al., 2024) instead defines it on the model's own output distribution, through the log-likelihood margin $d(x_f) = \log p_\theta(x_f) - \log p_{\text{ref}}(x_f)$, averaged over tokens, against a frozen reference at temperature $\beta > 0$:

$$\mathcal{L}_{\text{NPO}} = \tfrac{2}{\beta}\,\log\big(1 + e^{\beta\, d(x_f)}\big), \qquad \frac{\partial \mathcal{L}_{\text{NPO}}}{\partial d} = 2\,\sigma\big(\beta\, d\big), \tag{3}$$

whose gradient decays to zero as $d \to -\infty$, so the objective withdraws its own gradient once a sample has been unlearned. The two families are not commensurable in absolute value, a squared activation distance against a log-likelihood margin in nats, so we never difference across them and compare only orderings within a family.

One consequence is used later. The gradient reaching routed expert $E_j^{(\ell)}$ is scaled by its gate weight $\hat{w}_j$. The target $c\,u$ of Eq. 2 is externally chosen and does not shrink with the expert's own contribution, whereas the forget term of Eq. 3 is a function of the likelihood the model assigns, to which a rarely selected expert contributes little. Under a likelihood-family objective a shared learning rate is therefore not the same intervention across arms, which is why forgetting depth is matched explicitly rather than assumed from a shared step budget.

4 READING THE FORGET DOMAIN TWICE, AND A SELECTION SIGNAL

4.1 THE PHENOMENON: ROUTING ALONE MOVES THE FORGET NUMBER

Write $\text{FA}(\theta, \pi)$ for accuracy on the forget benchmark when weights θ are served under routing policy π, and $\pi_{\text{nat}}(\theta)$ for the policy the gate shipped with θ induces. *natFA* is the reading MoE unlearning is currently scored by: the model as served. With θ^0 the released weights, θ^u those an unlearning objective returns and $\pi^0 = \pi_{\text{nat}}(\theta^0)$, the fall in natFA is a sum of two terms, $\text{FA}(\theta^0, \pi^0) - \text{FA}(\theta^u, \pi^0)$, the *store* term, what the experts hold, and $\text{FA}(\theta^u, \pi^0) - \text{FA}(\theta^u, \pi_{\text{nat}}(\theta^u))$, the *routing* term, which experts are reached. Only the store term is unlearning, and **natFA reports only the sum**; the routed contribution is gated, so nothing in the objective drives the routing term to zero. The split is a definition, not a result, and we never measure the two terms apart: its use is that one control sets the store term to zero by construction.

The positive control. Reading θ^0 under a policy other than π^0 makes the store term vanish identically, leaving natFA free to move only through the routing term. At each intervened layer of the *unmodified* model we fix the k highest-affinity experts and offset their load-balancing correction bias by a large negative constant. They therefore cannot enter the top-6 set for any token, while routing elsewhere is untouched; the offset enters the top-6 comparison only and reaches no expert

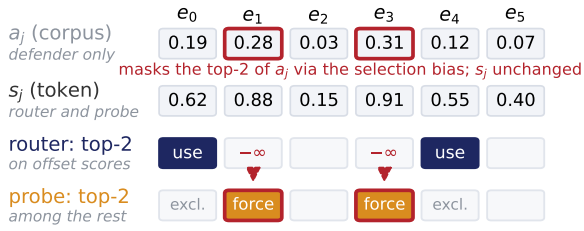


Figure 2: **Mask and probe on one token** (router top-2, k=2 masked, m=2; illustrative scores): the masked experts are the first the probe recovers.

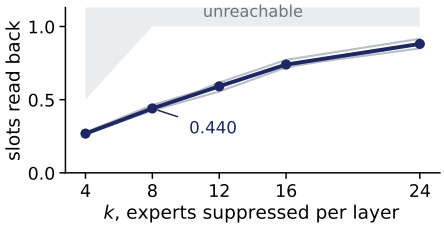


Figure 3: **Read-back**: probe slots held by suppressed experts; thin lines: $\ell = 5, 6, 7$; grey is above the ceiling $\min(k, m)/m$.

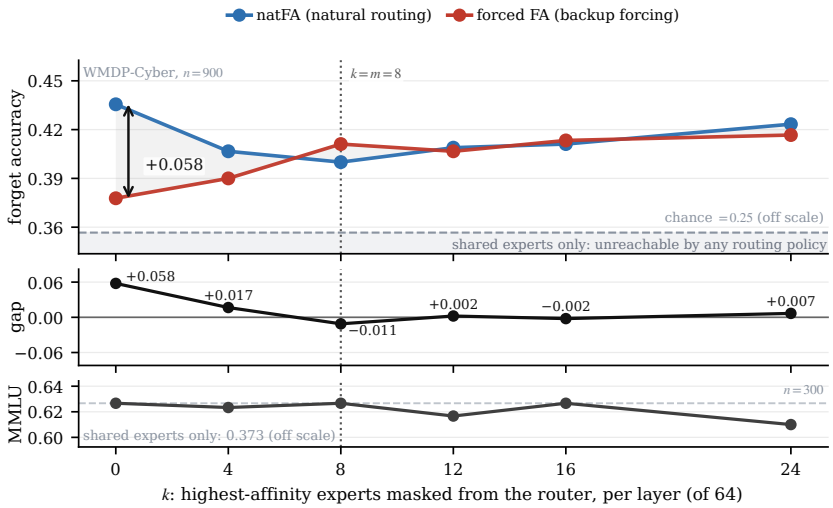


Figure 4: **Routing moved, parameters untouched.** Masking the k highest-affinity experts from the router. **Top**, the two rulers; **middle**, their difference GAP, printed at each k; **bottom**, MMLU.

output. No expert parameter is touched. The forget domain is then read twice at every k: under natural routing, and under a probe that serves each token from the next-best m experts instead (*forced FA*; both fixed below). The probe ranks by $s_j(h)$ and excludes the 6 experts the router actually selects, so a suppressed expert leaves natural routing while staying in the probe's pool at its original score (Figure 2). **The quantity this control tracks is their difference**, GAP = natFA − forced FA: the margin by which the router's own choice beats the experts it declines. A method that removes knowledge shrinks both readings and leaves GAP where it was; a method that only reroutes shrinks GAP instead. Only here do the two readings move in opposite directions; trained arms are compared at matched natFA, where GAP adds nothing and is not reported.

Corollary 1: a natural reading is not a parameter reading. Sweeping $k \in \{0, 4, 8, 12, 16, 24\}$ gives Figure 4. natFA falls at all five non-zero k, forced FA rises at all five, GAP goes from $+0.058$ to -0.011, and retain accuracy does not move. Since no expert parameter was touched, every one of those changes is attributable to the routing policy alone, which rules out reading natFA as a statement about what the parameters retain. At $k = 8$, natFA falls from 0.436 to 0.400. Forced FA rises from 0.378 to 0.411. On a second backbone whose router differs from ours on four properties at once, the control does not reproduce. Its natFA leg does not move, and although GAP collapses, a counterfactual probe that is additionally denied the suppressed experts recovers nothing beyond the unmasked baseline at any k. On that backbone the collapse is therefore a property of the probe's candidate pool rather than evidence of concealment, and we report it as a failure to generalise (App. B.2).

> **Insight 1**: *A natural-routing reading of the forget domain is not a statement about what the parameters retain: on the unmodified model, with no expert parameter touched, a pure routing change moves it while the forced reading does not follow.*

What this does not claim. We claim the collapse of GAP, not the sign of any individual residual, every gap at $k \geq 4$ being at most 0.017 in magnitude. At *every* k the masked experts stay in the probe's pool and are read back (Figure 3), filling 0.268 of its slots at $k = 4$, 0.440 at $k = 8$ and 0.881 at $k = 24$, below the ceiling a per-token rule would force. The upper end of the sweep is therefore the part closest to a restatement, and no claim here rests on a single k.

4.2 THE INSTRUMENT: TWO RULERS AND THE RULES FOR COMPARING THEM

Threat model. The defender ships any parameters and any routing behaviour; the adversary has white-box access at inference time and may *override the routing decision*, but may *not* retrain and cannot see the forget corpus or the original weights. Unlearning succeeds only if *no* admissible routing policy recovers the capability. This adversary is deliberately weak, so failing against it implies failing against the dense-model adversaries of Łucki et al. (2025); our probe searches a *strict subset* of even what it may do (App. E).

Three policies. With the architecture above, a *routing policy* π replaces the selected set $\mathcal{S}(h)$ at a designated set of layers $\mathcal{L}$, fixed experimentally and not part of the definition. Below, $\pi : \mathcal{X}(h)$ reads *serve each token from* $\mathcal{X}(h)$, identically at every $\ell \in \mathcal{L}$:

$$\pi_{\text{nat}} \colon \mathcal{S}_{\text{nat}}(h) \qquad \pi_{\text{bak}}(m) \colon \mathcal{B}_m(h) = \text{top-}m_{\, j \notin \mathcal{S}_{\text{nat}}(h)} s_j(h) \qquad \pi_{\text{sgl}}(\ell, j) \colon \{j\} \text{ at } \ell \tag{4}$$

π_{nat} is the deployed policy and $\mathcal{B}_m(h)$ the long tail the current router avoids; under π_{sgl} layer ℓ routes every token through expert j alone with weight 1 while all other layers route naturally; π_{sgl} is an instrument we borrow for selection.

Two invariants. Shared experts are never gated and never modified: no routing policy removes their contribution and no selection scheme updates them, so every accuracy sits above a reference level set by what they alone retain, which we measure rather than subtract. This describes the architecture we study, not MoE models in general (App. E). **Interventions are evaluation-time only**: training always routes naturally, so the trainable set is the only variable compared, and metrics are read as deployed with shared experts active.

> **Insight 2**: *Every forced reading is a **lower bound** on what an adversary recovers, never a certificate of removal: our probe searches a strict subset of the routing policies the threat model permits.*

The scorer is a plug-in. Each reading factorises into a routing policy and a scoring function X, and the intervention touches only the gate, so any X may be substituted:

$$\text{NAT-}X = X\big(\mathcal{D}_f;\, \pi_{\text{nat}}\big), \qquad \text{FORCED-}X = X\big(\mathcal{D}_f;\, \pi_{\text{bak}}(m)\big). \tag{5}$$

The operator in $\pi_{\text{bak}}(m)$ follows the top-non-selected forcing of Zhu et al. (2026); what this work contributes is applying it to every item, without their routing-shift condition, and the protocol of reading the pair side by side. We instantiate X as multiple-choice accuracy, since our forget target is a QA benchmark; two other scorers give the same direction, though not the same magnitude (Table 1). Utility RA is read on a held-out benchmark $\mathcal{D}_r$ under π_{nat}.

Corollary 2: two arms may be compared only at matched depth. An arm driven further along its own trade-off curve shows a lower forced FA regardless of which experts were selected, so raw endpoint values are not comparable across methods. Every comparison here is either matched on natFA or reported as a bound.

Five rules govern every comparison we report. Selection signals are computed on a calibration split disjoint from the evaluation split. Methods are compared within overlapping natFA bins. Random-expert controls are read only over the natFA range the method attains, and an arm that fails the *liveness gate* is withheld. Every arm is evaluated on an untargeted forget domain. The shared-expert level is used as a scale and never as a null.

4.3 A SELECTION SIGNAL READ UNDER THIS PROTOCOL

Everything above is independent of how experts are chosen; what follows is one case study conducted under Corollary 2. Under a compact budget we may update only B of N experts per layer. Existing MoE methods answer with *routing preference* in three variants: the mean gate score a_j^{unm}, the same mean with the score zeroed where the expert is not selected a_j^{msk}, and the hard activation frequency p_j^f. All three read π at natural routing. We instead answer with *recoverability*, how much of the forget target can be read out of a single expert when the router is bypassed: using π_{sgl}, define for layer ℓ and expert j

$$\Delta_j^{(\ell)} \;=\; \text{FA}(\mathcal{D}_{\text{cal}};\, \pi_{\text{sgl}}(\ell, j))\,, \tag{6}$$

where $\mathcal{D}_{\text{cal}}$ is a calibration split of the forget benchmark disjoint from the evaluation split $\mathcal{D}_f$. A high $\Delta_j^{(\ell)}$ means the forget target remains answerable when the model is forced through expert j alone. LEAK selects the B experts with the largest Δ per layer.

> **Insight 3**: *Recoverability and routing preference are not two estimates of one quantity. Across the 24 experts each rule selects, the three affinity variants share 17–19 with one another and 1–2 with ours (Fig. 5).*

All four rules instantiate one template; ours is the only one that intervenes on π. We validate only the accuracy instantiation: a corpus-likelihood one ranks experts on the forget and the retain corpus

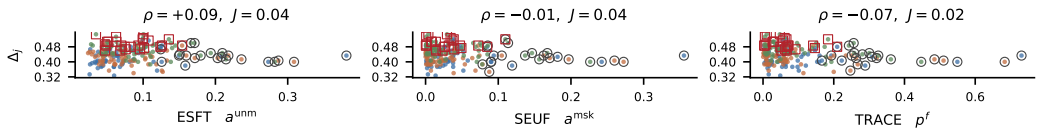


Figure 5: **Recoverability is not routing preference.** One point per routed expert per intervened layer, coloured by layer; each panel plots Δ_j against one published affinity signal. Outlined markers are the two selected sets: circles the affinity rule's, squares ours.

almost identically ($\rho = +0.73$), ordering general competence rather than target-specific knowledge (App. C.4). Where the forget target has no calibration set our selection step therefore has no validated scorer, and the affinity family remains applicable where ours does not. That restriction is confined to selection, since the second reading admits scorers needing neither labels nor options.

We do not propose Δ as a practical recipe: it costs 192 forward passes where the affinity family reads its values from one, requires a labelled calibration set that family does not, and has split-half reliability 0.46. Its role is evidential. A signal that ignores routing preference selects experts leaving measurably less recoverable capability at matched depth under the representation-family objective, though the ordering does not carry to the likelihood family. Routing preference therefore cannot determine where forget knowledge is recoverable from: one objective family in which it fails is enough to refuse it the status of a general criterion.

5 Experiments

5.1 Setup

We unlearn the WMDP cyber corpus from Moonlight-16B-A3B-Instruct with RMU, intervening at layers $\mathcal{L} = \{5, 6, 7\}$ and unfreezing $B = 8$ experts per layer, 24 in total, so that the selection signal is the only free variable across arms. The router is frozen throughout (App. A.1), so no arm can lower natFA by moving a routing decision: what a forced reading recovers here is held by experts the objective never updated. The layer set is inherited from the RMU configuration rather than chosen by us, and falls inside the band locate-then-edit work identifies independently (Meng et al., 2022), and is the same set Zhuang et al. (2025) adopt for RMU on MoE backbones. Forced FA is read under $\pi_{\text{bak}}(m)$ at $m = 8$ against two reference lines we measure rather than assume: the unmodified model returns 0.4115 on the same items and the shared experts alone 0.3371, so a routing-only intervention there can traverse at most 0.074. Each run is also read over the sweep $m \in \{1, 2, 4, 6, 8, 12, 16, 24, 32\}$ (Table 2, Figure 6). The corpora, the evaluation split and its registration, the scorer and its two known limits (Table 1) and the liveness gate are in App. A, and the five rules stated there fix every column, test and convention. Unless a caption says otherwise, trained arms are plateau-window means over 3 seeds, 5 for the two arms compared at matched depth, arms failing the liveness gate are withheld, and readings under different probe instruments are never differenced. Among trained arms, significance is claimed only for that pairing, with p a one-sided exact permutation test paired by seed; it reaches its floor of 0.031 only when every seed agrees in sign. Elsewhere we claim only that two groups of runs separate completely (Dror et al., 2018).

Table 1: **Three scorers on one item set**, $n = 150$.

scorer	natural	forced	shared only
accuracy	0.407	0.327	0.333
option CE	1.179	1.461	1.635
free-form NLL	6.803	7.009	7.448

Table 2: **Forced FA against** m, from single checkpoints; each cell is the mean of three runs.

	probe budget m								
	1	2	4	6	8	12	16	24	32
Leak	0.331	0.246	0.273	0.268	0.281	0.295	0.307	0.323	0.319
SEUF	0.330	0.268	0.294	0.333	0.363	0.364	0.364	0.373	0.372

5.2 Two rulers on six selection signals

At equal budget, the two rulers disagree. Six selection rules run under identical hyper-parameters. In Table 3 the three affinity rules drive natFA to chance, yet under forcing the same models return

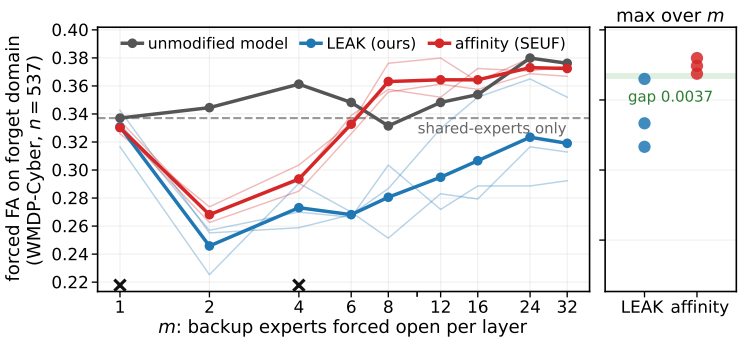


Figure 6: **The worst value over the probe sweep is the one that counts.** Forced FA against m: thick lines are means, thin lines individual runs; crosses mark the two budgets at which the arms are not completely separated; right, each run's maximum. Details: App. D.4.

above the 0.331 the unmodified model returns under the same probe: on those arms the probe measures no removal at all. Our arm reverses that pattern on both rulers, forgetting *less* than every baseline under natural routing while leaking least under forcing. A deeper unlearning run cannot explain this, because the mismatch has the wrong sign: our arm is the *shallowest* of the five surviving arms. Because the depths are not matched here, this reading is a bound and not an ordering; the ordering is established at matched depth below. Our arm's forced FA falls below the shared-expert level, which is a scale and not a null: a parameter update that corrupts the routed path can drive a reading beneath it. Random 1 and SEUF coincide under natural routing, 0.256 against 0.256, and separate under forcing, 0.315 against 0.355; random selection can lower forced FA and pays for it in utility, our arm reading lower under forcing and higher on retain accuracy.

Read at matched depth: the ordering resolves. Reading each arm at the 4 natural-routing depths of the grid gives panel (a) of Table 4, where our arm leaks less under forcing ($p = 0.031$). The difference grows with depth rather than appearing at one point, and is absent only at the shallowest grid value, where neither arm has yet moved by more than 0.021 from 0.412.

Restricted to shared grid values, the difference weakens. Because the arms do not cover the grid equally, we also report the restriction to the 2 grid values that both arms always reach. That restriction discards the depths where the two differ most and keeps the one at which neither has moved. There the difference is -0.028, and the runs no longer agree on its sign. The unequal reach is a property of the arms rather than a nuisance we can design away.

A longer run keeps the ordering and costs utility. Continuing our arm to 900 steps preserves the ordering on the second ruler at matched depth, over all six runs rather than the three that read best, and costs utility: retain accuracy falls to 0.533–0.597 against the baselines' 0.610–0.617.

The two points are mismatched in opposite directions: at 450 steps retain accuracy is matched and the depth is not, at 900 steps the reverse. A confound that explains the first reading, that we simply trained harder, predicts the opposite of what we observe there. One that explains the second, that we simply damaged the model, is excluded at the first point. No single confound accounts for both. Random 0 collapses and is withheld by the liveness gate. On the untargeted forget domain our arm's drop on WMDP-Bio is 0.036, inside the gate of 0.05 but 6–36 times larger than the affinity family's,

Table 3: **Six selection signals at a matched training budget**, read by both rulers.

Arm	Selection signal	natFA ↓	forced FA ↓	RA ↑
Reference rows (unmodified model, no training)				
Unmodified model	–	0.412	0.331	0.627
Shared experts only	–	0.337	0.337	–
Trained arms, 450 steps, every run passes the liveness gate				
LEAK (ours)	recoverability Δ	0.340	**0.285**	0.616
SEUF	masked affinity a^{msk}	0.256	0.355	0.617
ESFT	unmasked affinity a^{unm}	0.255	0.356	0.612
TRACE	selection frequency p^f	0.259	0.342	0.610
Random 1	uniform draw	0.256	0.315	0.587
Trained arm, 450 steps, no run passes the liveness gate				
Random 0	uniform draw	*withheld*	*withheld*	*withheld*

Table 4: **Read at matched depth, pooled**; per depth: Table 7. natFA is matched by design (interpolated in (a), one probe step in (b)); bold marks the better arm where the two separate.

Objective	Arm	natFA	forced FA ↓	RA ↑
(a) representation (RMU)	LEAK (ours)	0.370	**0.303**	0.620
$m = 8$	SEUF	0.370	0.353	0.625
(b) likelihood (NPO)	LEAK (ours)	0.360	0.384	0.601
max over 9 budgets	SEUF	0.353	0.377	**0.627**

a ratio whose width reflects how close to zero those denominators are. natFA remains the right number to report: it is the depth variable on which the comparison is matched.

The worst probe budget is the one that counts. The probe budget is the adversary's knob rather than a hyper-parameter we may tune, so what matters is the *worst* value a run attains over the sweep $m \in \{1, 2, 4, 6, 8, 12, 16, 24, 32\}$ (Table 2, Figure 6). Of the two budgets that do not separate the arms, $m = 1$ is the one whose sign is not ours, and at $m = 4$ the sign is ours but the ranges overlap. Against masked affinity, the only baseline for which per-budget states were stored, our worst run reads just 0.0037 below their best, two items out of 537. Quoting $m = 8$ alone would overstate the result, so we always quote it alongside the sweep.

Insight 4: *The two rulers order the same trained models in opposite directions, so a comparison is only licensed at matched forgetting depth; ranking arms by the natural reading alone inverts the conclusion.*

5.3 A SECOND OBJECTIVE FAMILY

We repeat the comparison under a likelihood-family objective. The representation-family ordering does not reproduce there: the forced reading does not separate the arms in our runs, and our arm loses more retain accuracy at matched depth ($p = 0.031$). The mechanism was predicted in §3: an expert's effective step size tracks its share of gate mass on the forget corpus, and our signal selects the low-flow tail by construction. Xie et al. (2026) derive the same scaling at the level of hard activation frequency, for the retain side of a different objective; this is its forget-side consequence. App. D.5 gives the full reading and the step-budget confound it carries.

6 CONCLUSION

We study machine unlearning in Mixture-of-Experts models and show that natural-routing evaluation can conflate changes in expert parameters with changes in routing behavior. A routing-only intervention on the unmodified model changes forget accuracy without modifying any expert parameter, demonstrating that a lower natural-routing accuracy is not, by itself, evidence that the target capability has been removed. We therefore introduce a forced-routing evaluation that probes the forget domain through experts outside the model's natural routing path and exposes residual capability that standard evaluation can miss. We further revisit expert selection for MoE unlearning. Existing methods select experts using routing affinity, whereas our recoverability-based criterion measures how much of the forget target remains accessible when the model is forced through an expert. The two criteria identify substantially different experts, and under representation-based unlearning, recoverability-based selection leaves less capability recoverable under forced routing at matched forgetting depth. Overall, our results show that MoE unlearning should not be evaluated from natural routing alone, and that routing preference does not necessarily identify the experts from which target capability is most recoverable.

AI USE STATEMENT

In this work, we used generative AI tools (large language model assistants) for language and literature-related assistance, including polishing and editing text written by the authors, translating papers and personal notes, and checking English phrasing and terminology. We did not use generative AI tools to generate synthetic datasets, run experiments, or produce experimental results. The authors reviewed and verified all AI-assisted content and take full responsibility for the final manuscript, including all text, results, and references.

Ethics statement

This work evaluates unlearning methods on WMDP, a benchmark of proxy questions about hazardous knowledge that is published for exactly this purpose. We introduce no new hazardous content and release no model whose safety properties differ from the public model we start from. Our central finding is that a widely used evaluation can overstate how much hazardous capability an unlearning method has removed. We report it because a defender who cannot measure removal cannot know whether a deployed model is safe; the routing override our probe uses requires no optimisation, no data and no training, so it is available to any party with the weights, and describing it does not create a capability that withholding it would deny.

Reproducibility statement

The split protocol, evaluation sets, scoring convention, hyper-parameters, liveness gate and reference-line measurements are given in full in App. A.1; the comparison rules that govern every number we report are stated in full in App. A.2. The calibration and evaluation halves are fixed by the seed reported there and their disjointness is verified by exact question-level fingerprints. Because our absolute values are not produced by the lm-evaluation-harness convention, we state the scorer explicitly in App. A.1 and compare only within it. Code for the routing intervention, the probe and the selection signals, together with the per-expert scores, is available at https://anonymous.4open.science/r/ForcedRoutingProbe. The operator is also specified in the text: Eq. 4 gives the three policies, and App. B.1 states which quantity the mask modifies and at which point of the forward pass it applies.

References

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction, 2024. URL https://arxiv.org/abs/2406.11717.

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. LEACE: Perfect linear concept erasure in closed form, 2023. URL https://arxiv.org/abs/2306.03819.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *IEEE Symposium on Security and Privacy (S&P)*, pp. 141–159, 2021. doi: 10.1109/SP40001.2021.00019.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *IEEE Symposium on Security and Privacy (S&P)*, pp. 463–480, 2015. doi: 10.1109/SP.2015.35.

Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models, 2024. URL https://arxiv.org/abs/2401.06066.

Aghyad Deeb and Fabien Roger. Do unlearning methods remove information from language model weights?, 2024. URL https://arxiv.org/abs/2410.08827.

DeepSeek-AI. DeepSeek-V2: A strong, economical, and efficient mixture-of-experts language model, 2024. URL https://arxiv.org/abs/2405.04434.

DeepSeek-AI. DeepSeek-V3 technical report, 2025. URL https://arxiv.org/abs/2412.19437.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1383–1392, 2018.

Ronen Eldan and Mark Russinovich. Who's harry potter? approximate unlearning in LLMs, 2023. URL https://arxiv.org/abs/2310.02238.

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. The language model evaluation harness, July 2024. URL https://zenodo.org/records/12608602.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic, 2022. URL https://arxiv.org/abs/2212.04089.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. Knowledge unlearning for mitigating privacy risks in language models, 2022. URL https://arxiv.org/abs/2210.01504.

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, 2024. URL https://arxiv.org/abs/2401.04088.

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, 2020. URL https://arxiv.org/abs/2006.16668.

Simon Lermen, Charlie Rogers-Smith, and Jeffrey Ladish. LoRA fine-tuning efficiently undoes safety training in Llama 2-chat 70b, 2023. URL https://arxiv.org/abs/2310.20624.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew B. Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Adam Khoja, Zhenqi Zhao, Ariel Herbert-Voss, Cort B. Breuer, Samuel Marks, Oam Patel, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Lin, Adam A. Hunt, Justin Tienken-Harder, Kevin Y. Shih, Kemper Talley, John Guan, Russell Kaplan, Ian Steneker, David Campbell, Brad Jokubaitis, Alex Levinson, Jean Wang, William Qian, Kallol Krishna Karmakar, Steven Basart, Stephen Fitz, Mindy Levine, Ponnurangam Kumaraguru, Uday Tupakula, Vijay Varadharajan, Ruoyu Wang, Yan Shoshitaishvili, Jimmy Ba, Kevin M. Esvelt, Alexandr Wang, and Dan Hendrycks. The WMDP benchmark: Measuring and reducing malicious use with unlearning, 2024. URL https://arxiv.org/abs/2403.03218.

Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin Xu, Enzhe Lu, Junjie Yan, Yanru Chen, Huabin Zheng, Yibo Liu, Shaowei Liu, Bohong Yin, Weiran He, Han Zhu, Yuzhi Wang, Jianzhou Wang, Mengnan Dong, Zheng Lin, Xin Yu, Xiaokun Yuan, Yutao Zhang, Yuxin Wu, Enming Yuan, Yongsheng Kang, Zhiqi Huang, and Zhilin Yang. Muon is scalable for LLM training, 2025a. URL https://arxiv.org/abs/2502.16982.

Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Yuguang Yao, Chris Yuhao Liu, Xiaojun Xu, Hang Li, Kush R. Varshney, Mohit Bansal, Sanmi Koyejo, and Yang Liu. Rethinking machine unlearning for large language models. *Nature Machine Intelligence*, 7(2):181–194, 2025b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Jakub Łucki, Boyi Wei, Yangsibo Huang, Peter Henderson, Florian Tramèr, and Javier Rando. An adversarial perspective on machine unlearning for AI safety. *Transactions on Machine Learning Research (TMLR)*, 2025.

Aengus Lynch, Phillip Guo, Aidan Ewart, Stephen Casper, and Dylan Hadfield-Menell. Eight methods to evaluate robust unlearning in LLMs, 2024. URL https://arxiv.org/abs/2402.16835.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. TOFU: A task of fictitious unlearning for LLMs, 2024. URL https://arxiv.org/abs/2401.06121.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT, 2022. URL https://arxiv.org/abs/2202.05262.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016. URL https://arxiv.org/abs/1609.07843.

Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, 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, Noah A. Smith, Pang Wei Koh, Amanpreet Singh, and Hannaneh Hajishirzi. OLMoE: Open mixture-of-experts language models. In *International Conference on Learning Representations (ICLR)*, pp. 62061–62121, 2025.

Vaidehi Patil, Peter Hase, and Mohit Bansal. Can sensitive information be deleted from LLMs? objectives for defending against extraction attacks, 2023. URL https://arxiv.org/abs/2309.17410.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning aligned language models compromises safety, even when users do not intend to!, 2023. URL https://arxiv.org/abs/2310.03693.

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, 2017. URL https://arxiv.org/abs/1701.06538.

Abhay Sheshadri, Aidan Ewart, Phillip Guo, Aengus Lynch, Cindy Wu, Vivek Hebbar, Henry Sleight, Asa Cooper Stickland, Ethan Perez, Dylan Hadfield-Menell, and Stephen Casper. Latent adversarial training improves robustness to persistent harmful behaviors in LLMs, 2024. URL https://arxiv.org/abs/2407.15549.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. MUSE: Machine unlearning six-way evaluation for language models. In *International Conference on Learning Representations (ICLR)*, pp. 27797–27818, 2025.

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 Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 784–801, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.46.

Jingyi Xie, Yijun Lin, Yinjiang Xiong, Zhikun Zhang, and Sai Li. Routing-aware expert calibration for machine unlearning in mixture-of-experts language models, 2026. URL https://arxiv.org/abs/2606.10338.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Rui Men,

Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yichang Zhang, Yu Wan, Yunfei Chu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report, 2024. URL https://arxiv.org/abs/2407.10671.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning, 2023. URL https://arxiv.org/abs/2310.10683.

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

Andy Zhu, Rongzhe Wei, Yupu Gu, and Pan Li. GRIP: Algorithm-agnostic machine unlearning for mixture-of-experts via geometric router constraints, 2026. URL https://arxiv.org/abs/2601.16905.

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 Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 8664–8678, Vienna, Austria, July 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.424.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to AI transparency, 2023. URL https://arxiv.org/abs/2310.01405.

A Experimental setup and measurement conventions

A.1 Setup

Model and data. We use Moonlight-16B-A3B-Instruct (Liu et al., 2025a): 27 layers, 64 routed and 2 shared experts per MoE layer, top-6 routing and scaling constant 2.446 (§3.1). The top-k selection of Eq. 1 ranks $s_j^{(\ell)}(h) + b_j^{(\ell)}$, where $b^{(\ell)}$ is the auxiliary-loss-free load-balancing correction bias carried in the released configuration; it shifts which experts are selected and does not enter the weights $\hat{w}_j$, and the control of §4.1 offsets exactly this term. Every policy of §4.2 replaces $\mathcal{S}^{(\ell)}(h)$ in both places it enters Eq. 1: $\hat{w}_j$ is renormalised over the experts actually served, so the weight 1 of π_{sgl} is $\hat{w}_j$ for a single expert, and γ still scales the routed term. The router of the second backbone does not renormalise, and App. B.2 reports its probe under both weightings. The steering layer of Eq. 2 is $\ell^* = 7$, the last layer of $\mathcal{L}$ (Table 5). WMDP contains 3668 items after deduplication. We split it at the midpoint under a fixed seed (20260714), compute selection signals on the first half and evaluate only on the second, with disjointness verified by exact question-level fingerprints (R1). From the 1834 items of the evaluation half we draw 1000 items, registered before any arm was trained: 537 cyber, the only forget domain reported as a primary metric; 366 bio, reserved as the untargeted domain of R4; and 97 chem, too few items to support a conclusion. Retain accuracy is read on 300 held-out MMLU items.

Training. An expert receives a gradient only if it is in the trainable set and is activated in that step's forward pass. Training always routes naturally (§4.2), so the trainable set is the only variable, and every arm unfreezes the same number of experts. Only the three projection matrices of the selected experts are updated; the router, the shared experts, attention and all other experts stay frozen. Table 5 lists the remaining settings. The forget objective never sees a question format and the scorer is a separate choice, so no arm is read by a ruler derived from its own criterion.

Probe, reference lines and scorer. Forced FA masks the natural top-6 set and activates the $m = 8$ highest-scoring of the remaining 58 experts; given m the probe has no free parameters. Disabling all routed experts at the intervened layers leaves 0.3371, against 0.4115 for the unmodified model on the

Table 5: Training settings, shared exactly by every arm.

Objective	RMU (Eq. 2), fixed random unit vector resampled per run
Forget corpus	WMDP cyber, first 600 blocks of 512 tokens
Retain corpus	WikiText-103 (Merity et al., 2016)
Trainable set	$B = 8$ experts at each layer of $\mathcal{L} = \{5, 6, 7\}$, 24 in total
Steps	450 steps, evaluated every 25
Optimiser	AdamW (Loshchilov & Hutter, 2019), constant learning rate 5×10^{-5}, weight decay 0.01, gradient clipping 1.0
Batch size, retain weight	2, $\alpha = 1200$
Steering layer ℓ^*	the last layer of $\mathcal{L}$, $\ell^* = 7$: Eq. 2 is read on the residual stream there, and its gradient reaches every unfrozen expert of the block
Steering magnitude	$c = 5.0$ times the mean activation norm at ℓ^* (7.34), giving $c = 36.7$
Precision	parameters and optimiser state in bfloat16

same items, so a routing-only intervention can traverse at most 0.074; that level is a reference and not a floor (R5). Accuracy is symbol argmax over the four option tokens, zero-shot and without length normalisation. That differs from the lm-evaluation-harness convention (Gao et al., 2024), so our absolute values are not comparable to numbers reported under it. On the unmodified model, moving between two scoring conventions changes forget accuracy by more than the largest difference any of our conclusions rests on; every comparison is therefore within one scorer, and numbers from different scoring pipelines are never differenced.

The liveness gate. R3 admits an arm only if its retain accuracy at the last recorded step exceeds 0.50. The threshold is a property of the protocol, fixed before any arm was trained and never tuned per arm; an arm below it is withheld and reported as withheld, not scored. It separates the runs without ambiguity rather than by a margin: the fifteen surviving runs end at 0.587–0.627 retain accuracy, and the one arm that fails ends at 0.200–0.220 in all its runs, which is why Random 0 is the only withheld row of Table 3. The survival rate of every random arm is reported with its accuracies. This gate reads retain accuracy and is a different instrument from the untargeted-domain gate of R4, which bounds an arm's drop on the bio domain at 0.05; the two are reported separately and never combined.

Two facts about the runs. One random-selection arm collapses in every run; we report the collapse rather than discarding the arm, because a collapsed run would otherwise rank first on forgetting depth. The main runs predate a fix to the order in which the steering magnitude is computed; the magnitude was verified to be identical across all eighteen runs, so the fix changes no reported value.

A.2 Comparison rules and measurement conventions

Table 6: The five rules that govern every comparison reported in the paper, and the failure each prevents.

	Rule	Prevents
R1	Selection signals are computed on a calibration split disjoint from the evaluation split.	A signal being scored on the items it is later credited with answering.
R2	Methods are compared within overlapping natFA bins.	An interval supported only by interpolation being read as measured.
R3	Random-expert controls are read only over the natFA range the method attains; an arm failing the liveness gate is withheld.	A control that unlearns more deeply reaching a lower forced FA by sliding along the same curve.
R4	Every arm is evaluated on an untargeted forget domain.	Indiscriminate damage being scored as targeted forgetting.
R5	The shared-expert level is used as a scale and never as a null.	A level that a corrupted routed path can be driven below being read as a floor.

Columns of Table 7. Every difference between the two arms quoted in the text is ours minus SEUF, formed before rounding, so it can differ in the last digit from the difference of two rounded table entries. The two panels are comparable in direction only. A pooled row averages within each

Table 7: **Read at the same forgetting depth, do the two arms differ on the second ruler?** Each row fixes a natural-routing depth and reads our arm and masked affinity (SEUF) there: **(a)** representation, **(b)** likelihood. Bold marks the better arm where the two separate; the pooled rows are those of Table 4.

	forced FA		retain acc.	
Matched depth (natFA)	ours	SEUF	ours	SEUF
RMU, probe budget $m = 8$				
0.400	0.333	0.344	0.627	0.628
0.375	0.306	0.350	0.620	0.626
0.350	0.286	0.363	0.618	0.619
0.325	0.272	0.364	0.608	0.621
pooled	**0.303**	0.353	0.620	0.625
pooled, fully covered rows only	0.319	0.347	–	–
NPO, maximum over 9 probe budgets				
0.375	0.390	0.375	0.614	0.621
0.350	0.381	0.378	0.603	0.627
0.325	0.387	0.379	0.565	0.633
pooled	0.384	0.377	0.601	**0.627**

run first, so its unit is the run, and the claim rests on the pooled rows rather than on any single depth; under RMU the pooled difference is -0.051, with a between-run standard deviation of 0.018.

Protocol asymmetries, and a recomputation. The likelihood-family runs log a per-cell probe margin, so a cell can be discarded when neither arm has moved; the RMU runs predate that field, which is why the shallowest RMU row is reported rather than filtered. Our arm reaches fewer runs at the two deepest RMU rows because a shared step budget buys it less depth, hence the restriction to fully covered rows in §5.2. The depth grid, cell width and coverage rule were fixed for §5.3 before the additional RMU runs were trained and were not searched afterwards. Binning recorded evaluation points by depth instead of interpolating returns -0.045 against -0.051, with every run again agreeing in sign; we report the interpolating estimator because the second family uses it. R2 restricts a comparison to the depth interval both arms measure; where two recorded points bracket a grid value, the reading at that value is interpolated between them, and the binned estimator above is the check on what that costs. The two panels of Table 4 also differ in probe budget: the representation panel is quoted at $m = 8$ while the likelihood panel is quoted at its worst value over 9 budgets. That asymmetry is a logging limitation and not a choice of statistic. The per-budget states a full sweep needs were stored for the likelihood runs and not for the RMU depth grid, so the RMU sweep exists only at the 450-step endpoints, where the depths are not matched (App. D.4); at those endpoints the worst-case separation against masked affinity narrows to 0.0037, which we report there rather than here.

Overlap between each arm's selection and the probe segment. Of the 24 experts an arm updates, 25% fall inside gate-score ranks 7 to 14 for our arm, 25% for unmasked affinity and 33% for masked affinity. The affinity rules occupy the probe segment as much as we do, or more, yet their forced FA does not move, so the overlap does not run in the direction a self-selection reading requires. For unmasked affinity the value is structural: its top 8 experts by gate score meet ranks 7 to 14 in exactly two experts per layer.

B THE POSITIVE CONTROL

B.1 ON THE MAIN BACKBONE

How the suppression interacts with the probe. The control subtracts a large constant from the gate's load-balancing correction bias, so a suppressed expert cannot enter the natural top-6 at any token. The probe recomputes gate scores from the gate weight alone, which the correction bias does not enter, so a suppressed expert leaves the natural selection while remaining in the probe's pool at its original score (Figure 2). The same run records no token at which a suppressed expert enters the natural top-6, so the natural leg does not read suppressed experts back. How often the probe does is

measured in Table 9: 0.440 of its slots at $k = 8$ against a ceiling of one, rising to 0.881 at $k = 24$. The reading at $k = 8$ is therefore neither independent of the suppression nor a restatement of it, and the upper end of the sweep is the part closest to a restatement.

Table 8: **The positive control.** The k highest-affinity experts of the *unmodified* model are masked through the gate's selection bias: **no expert parameter is modified**. Moonlight, $n = 900$.

Masking	natFA	forced FA
none, $k = 0$	0.436	0.378
$k = 8$	0.400	0.411

Reading Figure 4. The figure comes from a second scoring pipeline with $n = 900$ items rather than 537, so it is never differenced against §5.2, and its shared-expert line on the cyber domain is 0.357 rather than 0.337. That pipeline applies the same split seed and reads the evaluation half, filling a per-domain quota in shuffled order rather than drawing over the whole pool. It differs from the registered set in size and in the deduplication key applied before the shuffle, so its half is aligned with the registered one by construction rather than item for item, which is why the two carry different reference lines. R1 governs *selection*, and this control selects nothing and trains nothing: it reads one unmodified checkpoint under two routing policies, so no reading here is scored on items a signal was chosen from.

Three limits. We claim the collapse of the distance between the rulers, not the sign of any individual residual: every gap at $k \geq 4$ is at most 0.017 in magnitude. We do not attribute the collapse to a mechanism: from $k = 0$ to $k = 8$ the two readings move by similar amounts in opposite directions (-0.036 and $+0.033$), so neither leg alone accounts for it. And because suppressed experts are read back at every k, we rest no claim on any single k.

Table 9: Read-back fraction: of the $m = 8$ slots the probe activates, the fraction held by experts the control has suppressed, on the unmodified model at $\mathcal{L} = \{5, 6, 7\}$, averaged over tokens (at most 20,000 per layer) on 500 of the 1300 items the control forwards. The ceiling $\min(k, m)/m$ is the largest attainable value. The layer-averaged row is formed from the unrounded per-layer fractions, so it need not equal the mean of the three rounded rows printed below it. These are fractions of slots, not accuracies, and are never differenced against Table 8 or Table 3.

k	4	8	12	16	24
mean over layers	0.268	0.440	0.591	0.740	0.881
$\ell = 5$	0.262	0.426	0.556	0.720	0.879
$\ell = 6$	0.266	0.430	0.617	0.772	0.916
$\ell = 7$	0.276	0.463	0.600	0.729	0.850
ceiling $\min(k, m)/m$	0.500	1.000	1.000	1.000	1.000

Why the suppressed set is fixed per layer. The set is chosen once per layer from a corpus-level score. Suppressing each token's own top-k instead would make natural routing coincide with the backup path the probe reads, collapsing the two rulers into one; it would force the read-back fraction to one whenever $k \geq m$; it would leave no fixed expert set to compare with a training budget or with the published rules, all of which return a fixed set per layer; and it would act on every retain-domain token alike, which the liveness gate rejects as damage rather than concealment.

B.2 On a second backbone

The control trains nothing, so we repeat it on Qwen1.5-MoE-A2.7B-Chat, whose router differs from ours in four respects: a softmax over all experts instead of a per-expert sigmoid, no scaling constant on the routed term, all 24 layers sparse with 60 routed experts and top-4 selection, and a gated shared branch, so the shared-expert invariant of §4.2 does not hold. We intervene at layers $\{4, 5, 6\}$ (the same relative depth as ours), keep $m = 8$, sweep the same $k \in \{0, 4, 8, 12, 16, 24\}$ with the same items, prompt and scorer, and never difference the result against Table 8: item sets and pipelines

differ, so only directions are comparable. The run costs under one GPU-hour. Its split was re-derived on that machine and drops three duplicate questions (3665 items before the split rather than 3668; the cyber half is 537 items in both).

Two probe conventions. This router does not renormalise selected weights, so a backup expert either keeps its own gate weight (the model's convention, primary) or the m weights are renormalised to sum to one; both were fixed before the run. The model's own top-4 weights sum to about 0.25 of the gate distribution (tenth to ninetieth percentile 0.15–0.39), so renormalising multiplies the routed contribution roughly fourfold, which is why the model's own weighting is primary.

Table 10: The positive control on Qwen1.5-MoE-A2.7B-Chat.

k	natFA	forced FA (own weights)	forced FA (renorm.)
0	0.389	0.283	0.279
4	0.389	0.391	0.331
8	0.387	0.374	0.343
12	0.358	0.387	0.359
16	0.369	0.367	0.365
24	0.367	0.367	0.376

Reference lines. Removing the routed branch at the three layers leaves 0.279, removing the shared branch leaves 0.326, and natural routing returns 0.389. The routed path is worth 0.110 here, more than the 0.074 it is worth on ours, so the sweep is not limited by a lack of room; the two lines are not additive. Before any suppression the probe returns 0.283, two items above the shared-expert line.

What reproduces and what does not. At $k = 0$ the rulers are +0.106 apart, and at every $k \in \{4, 8, 12, 16, 24\}$ they are within 0.030 of each other. Under renormalised weights the distance at $k = 0$ is +0.110 and closes more slowly, being still 0.058 at $k = 4$. The two legs do not reproduce in proportion. Forced FA rises at all five non-zero k, by +0.084 to +0.108 (+0.052 to +0.097 renormalised). natFA is unchanged at $k = 4$ and one item lower at $k = 8$, and falls by at most 0.032, at $k = 12$, so we do not claim that the natFA leg transfers. Mean gate scores on this router are close to flat, the most preferred expert reaching only 1.25 to 1.61 times the uniform level over 307,200 forget-corpus tokens, so removing the top-k leaves the natural path enough substitutes.

Probe depth. Over $m \in \{1, 2, 4, 8, 16, 24\}$ the distance at $k = 0$ stays between $+0.104$ and $+0.128$, and at $k = 8$ between -0.004 and $+0.013$; at the depth that maximises forced FA it is $+0.104$ and -0.004, so the collapse is not a property of the depth we carried over. Before suppression forced FA spreads by only 0.024 over the sweep and sits at the shared-expert line, so the routed tail carries nothing readable at any depth we tried; after it, a probe opening a single expert per token recovers 95% of the routed contribution, against 5% before. The two sweeps form a cross rather than a grid: k is swept at $m = 8$ and m at $k \in \{0, 8\}$.

The collapse on this backbone is a property of the probe's pool. Suppressed experts fill 0.167 of the probe's slots at $k = 4$ and 0.733 at $k = 24$. Repeating the forced reading with them also removed from the pool, a decomposition of the rise rather than the threat model of §4.2, gives a rise of +0.097 to +0.108 with the pool open and of at most 0.006 with it closed, while natFA moves by at most 0.019; the closed-pool reading sits between 0.2626 and 0.2700, below this backbone's shared-expert line of 0.2793. Neither leg of the control reproduces here, and we report this as a failure to generalise rather than as a qualified success.

The same decomposition on the main backbone. There the open-pool rise is +0.012 to +0.039, and with the pool closed it survives in part: it is positive at all 5 values of k and reaches +0.030 at $k = 16$, where 60% of the open-pool rise at $k = 8$ is not read-back. At the other values of k it is at most +0.022, and the largest k returns to the unmasked baseline. We rely on the contrast with the second backbone, where the same quantity is zero at every k, and not on the magnitude on either.

C THE SELECTION SIGNAL

C.1 THE SELECTION TEMPLATE, AND WHY THE PROBE IS SCORED BY THE ROUTER

All four rules instantiate one template. With $\pi(\ell, j)$ a routing policy, s a per-token readout and R an aggregator over token positions, every selection rule we compare has the form

$$S_j^{(\ell)} \;=\; R\Big[\, s\big(\mathcal{D};\, \pi(\ell, j)\big)\Big], \tag{7}$$

with $\mathcal{D}$ forget text and s a gate-derived readout for the three baselines (§4.3), and $\mathcal{D}_{\text{cal}}$ and the argmax over the option symbols for ours. The three baselines leave π at natural routing, so they are defined on any input, need no labels or options, and report which experts the router *prefers*. Ours is the only rule that intervenes on π, which is what establishing where the target knowledge *is* requires, and an intervention is mutually exclusive across experts. The aggregator matters as well: R is an identity at one scored position for our rule and a mean over all positions for the other three, and App. C.4 shows what a corpus-averaged aggregator does when combined with an interventional π. Despite its name, Δ is an accuracy and not a difference, since selection uses only the within-layer ordering.

The probe ranks candidate experts with the model's own gating logits rather than with any method's signal, so it is identical for every rule under comparison; a probe defined through a method's own scoring function would read each arm with a ruler derived from its own criterion. Given m it has no free parameters: the masked set is the natural top-6 at that token and the activated set the m highest-scoring experts among the remaining 58. Its overlap with each arm's selection is in App. A.2.

C.2 COST, RELIABILITY AND REDUNDANCY OF THE SIGNAL

Cost. Computing Δ takes one forced sweep of the calibration set per expert, $|\mathcal{L}| \cdot N = 192$ in all, and no training; it is cached per (model, forget target) pair. The affinity family reads every value from one sweep, since a single forward pass records every gate score, whereas forcing layer ℓ through expert j excludes forcing it through j'; the ratio of forward passes is therefore 192 whatever calibration set is fixed. Our selection took $192 \times 150 = 28{,}800$ single-item evaluations, approximately 1.4 hours on one accelerator at a measured 0.178 s per item (range 0.176–0.183 s over four runs, 0.179 s in an independent timing on a larger set). We timed neither an unlearning run nor affinity scoring, so we do not convert the ratio into time. The cost is a reason not to adopt Δ as a recipe (§4.3), not a reason to doubt what it establishes.

Competence and reliability. Forced single-expert accuracy on the forget domain correlates with the same experts' competence on the retain domain (Spearman $\rho = 0.45$, $n = 64$); controlling for it leaves a substantial association with forget-domain competence (partial $\rho = 0.66$). We report this as exploratory: the criterion registered before the experiment used the contrast between retain and forget loss, which is close to zero here, and we report both. Accuracy and loss come from the same forward passes over the same 150 items, so part of their association reflects the shared sample; the retain-domain correlation, on a disjoint item set, is not affected. Δ_j reads one expert at a time, so what it scores is the forget capability an expert carries *on its own*. Capability that exists only in the co-activation of several experts is invisible to it, and a set assembled from single-expert scores need not be the best set of that size. The redundancy measurement below bounds how much this can cost on this backbone: a single routed expert already recovers most of the routed contribution, so the single-expert reading is not close to empty here, but on an architecture whose experts are individually weaker the same signal would need a set-level form. The split-half reliability of the selector is 0.46, and two independent readings of Δ share 7 of the 24 experts they each select (Jaccard 0.171, 2.6× the chance value), so a different item sample would give a materially different top-eight set. A less reliable selector should be harder to tell from a random one, and the selected group nonetheless separates from random selection in every run.

Redundancy. On a held-out set of 150 questions the routed path contributes 0.147 accuracy over the shared-expert-only path. Forcing that computation through any single one of the 64 routed experts recovers 59% of the contribution on average (mean 0.466, standard deviation 0.036; we report the mean, since selection inflates the maximum), forcing the 24 highest-scoring alternatives

instead of the natural selection recovers 59%, and the main evaluation split gives the same fraction (58%) under a different scoring pipeline.

C.3 Selection overlap and traffic share

Rank correlations here are Spearman's ρ, pooled over the 3×64 routed experts of the intervened layers as in Figure 5; each signal selects 24 experts, and Δ_j is the calibration-half scoring of Eq. 6 that every trained arm of §5.2 uses, with the per-expert scores released with the code. The three affinity variants correlate with one another at $\rho = 0.785$–0.941 and overlap at Jaccard 0.55–0.66, sharing 17–19 of their 24 selected experts; ours correlates with them at $\rho = +0.088$, -0.010 and -0.075 (averaging within a layer instead gives -0.015, -0.074 and -0.111) and overlaps them at 0.02–0.04, sharing 1–2, below the 0.067 expected and the 0.07 observed for two independent selections of the same size. On the forget corpus the experts the affinity family chooses are selected 2.98–3.18$\times$ as often as the uniform baseline 0.0938, and ours 0.76$\times$, inside the range of random selection (four draws spanning 0.71–1.11$\times$); since TRACE thresholds at an upper outlier of that distribution, its rule cannot in principle reach the experts our signal selects. On the evaluation corpus the affinity selections absorb 42.7–49.7% of all routed traffic in the intervened layers and ours 7.6%; 71% of our selections lie in gate-score ranks 15–64, whereas 62.5–75% of the affinity selections are the router's own top-6. The ratio of forget- to retain-corpus selection frequency is 4.8–6.0 for the affinity family and 0.7–1.1 for both our selection and random selection, with no overlap between the groups.

C.4 A label-free instantiation of Δ

We tested one instantiation of the template other than accuracy and report it as a negative result: an expert is scored by the mean token-level cross-entropy of the forget corpus under $\pi_{\mathrm{sgl}}(\ell, j)$, which needs no labels and no answer options, with the layer set, the per-layer budget and every other setting fixed, selecting 24 experts. Because a low loss and a high accuracy both mean that an expert supports the forget target, the matching rule keeps the *smallest* losses; the literal rule is reported alongside.

The variant's selection overlaps ours at Jaccard 0.26, about a quarter (0.02 under the literal rule, against 0.067 for two independent draws of the same size): more than chance, and far from a substitute. Its Spearman correlation with Δ, computed within a layer and averaged over the 3 intervened layers, is -0.33 (-0.58, -0.16, -0.24); pooling the layers instead gives -0.55, a between-layer artefact. Scored on the retain corpus it agrees with Δ equally well ($\rho = -0.33$, Jaccard 0.26) and recovers the same ten of the thirty-eight experts in the union of the two selections, and its forget and retain scorings nearly coincide ($\rho = +0.73$, per layer $+0.75$, $+0.69$, $+0.75$; Jaccard 0.41), so it ranks a property of the expert rather than of the target. We attribute this to the aggregator. A corpus-averaged cross-entropy is dominated by general fluency, which lowers the loss on every block whether or not it carries the target, whereas a multiple-choice item is scored at one position where fluency is common to all four options and cancels. Any corpus-averaged aggregator should therefore carry the same generic component; we did not test a third instantiation and state this as a prediction. The result narrows our scope where §4.3 says it does, since the label-free scorer that would have removed the need for a calibration set is the one ruled out here. It does not weaken §5.2, which requires only that *some* signal ignoring routing preference locates recoverable capability better than routing preference does.

D Additional results

D.1 Replacing the evaluation scorer

Because the intervention touches only the gate, the scorer X is a plug-in: the same reading can use the cross-entropy of the option tokens, the conditional likelihood of a free-form answer with no options at all, or, for targets such as chat logs or copyrighted text, a verbatim-memorisation scorer. On an identical intervention and item set, the forced path is degraded relative to the natural path under three scorers (Table 1; the gap is $\mathrm{natural} - \mathrm{forced}$ for the accuracy row and $\mathrm{forced} - \mathrm{natural}$ for the two loss rows, where larger is worse): multiple-choice accuracy, the cross-entropy of the

four option tokens, and the conditional negative log-likelihood of a free-form answer in a single-turn dialogue. Under a corpus continuation scorer the positive control's suppression degrades 34.7% of the natural path's available range against 8.4% of the forced path's, an asymmetry of 4.1 times. Neither direction is therefore an artefact of the multiple-choice format. The magnitude is not shared: under the free-form scorer the gap is 0.32 of the natural-to-shared-only distance, against 1.09 under multiple-choice accuracy (about 29%), and we do not claim the scorers are interchangeable. These are pure routing interventions on the unmodified model; every post-training number in the paper is a multiple-choice accuracy, and whether the trained arms separate under a scorer without answer options is untested. On the selection side, the substitute we tried is a four-way cross-entropy over the answer tokens, a differentiable form of the same judgement, and the one label-free selector we tried is that of App. C.4.

D.2 Reading without the shared experts

Every accuracy in the main text keeps the shared experts active, as at serving time (§4.2); here they are disabled at the intervened layers so that only the routed path contributes. On the unmodified model the routed path alone returns 0.279 on the cyber forget domain, the shared path alone 0.337, and the two together 0.412: neither reaches the pair and the two do not sum to it, which is why the shared-expert level is used as a scale and never as a null (R5). On the trained models the routed path returns 0.265 for our arm and 0.238 for masked affinity, changes of -0.014 and -0.041 from the unmodified routed path. The ordering here is the natural-routing one, as expected: the affinity family updates the experts carrying most of the routed traffic (App. C.3), which removing the shared path exposes, whereas our arm updates experts the router rarely visits. These values come from the step-449 checkpoint rather than the plateau window of Table 3 and are never differenced against it, and the random arms available at this reading are two further draws (0.251–0.258), reported for completeness rather than as controls.

D.3 Continuation to 900 steps

The arms are not at equal depth at 450 steps, so we continue our arm to 900 steps in two variants of each run that differ only in master-weight precision: arm A keeps the bfloat16 parameters and optimiser state of the main runs, and arm B holds a float32 master copy, which removes rounding-induced freezing. The baselines are not continued, being already at chance on the natural ruler at 450 steps. We report all six runs rather than the three that read best. Arm B attains depth matching, with natFA 0.251, 0.250 and 0.255, at or below every baseline; arm A does not (0.272, 0.274, 0.248). The highest forced FA among the six runs is 0.322 (arm A), read at the 900-step endpoint. Read at their own 450-step endpoint, the lowest of the nine affinity runs is 0.317, three items out of 537 below our highest; over the plateau window of Table 3 the lowest affinity run is 0.332, above our highest. At the level of runs the ordering is complete under the window reading and complete apart from that one pair under the endpoint reading, which we state because it is the less favourable; at the level of arms it is unchanged under both, the lowest affinity arm mean being 0.342. The cost is utility: retain accuracy falls to 0.533–0.597 over the six runs (0.553–0.597 for arm A, 0.533–0.573 for arm B), against 0.610–0.617 for the baselines at 450 steps.

D.4 The sweep over the probe budget, in full

Seven of the nine non-zero budgets separate our runs from the masked-affinity runs completely, that arm being the only baseline with stored per-budget states; differences are reported as the affinity group minus ours, so a positive value means our runs read lower. At $m = 1$ the difference is -0.0006, the only budget whose sign is not ours: a single backup expert is too close to the reference line for any method to be distinguished and the ranges of the runs overlap completely. At $m = 4$ it is +0.0205, with our sign but overlapping ranges. Taking each run's maximum over m (ours 0.3166, 0.3333 and 0.3650; masked affinity 0.3687, 0.3743 and 0.3799; Figure 6, right), the two groups separate completely, the strongest outcome a design of three runs against three can show. The separation is fragile: its margin is 0.0037, two items out of 537, so it rests on an ordering of six numbers, not on their spacing. Reporting the worst case also costs us: quoting $m = 8$ alone would put the gap at 0.0826 and $m = 24$ alone at 0.0497, so the maximum shrinks our reported advantage by a factor of about 22. Single points are also hard to read near the reference line: the unmodified model returns 0.331 at $m = 8$, below the shared-expert line of 0.3371, and 0.380 at $m = 24$.

Two further properties of the worst case. Two of our three runs attain their maximum at $m = 1$, one of them attaining the same value again at $m = 24$; the two values, 0.3166 and 0.3333, lie below the reference line and far below the lowest affinity maximum of 0.3687, so the effect on the comparison is conservative. Within each arm, moving the probe from $m = 8$ to $m = 24$ leaves forced FA almost unchanged for all three affinity runs but raises it for two of ours, and the gap between the arms narrows from 0.0826 to 0.0497; we offer no mechanism for this.

Reading Figure 6. The panel plots step-449 checkpoints, the final step of the 450-step run under zero-indexed numbering, of the two arms with stored per-budget states, so it carries no natFA and no RA and cannot be read for depth matching. $m = 0$ is omitted because every curve returns 0.3371 bit-for-bit there. A single checkpoint is not a plateau-window mean, so the figure and Table 3 differ at $m = 8$ and are never differenced; the gaps quoted above come from the same checkpoints.

D.5 The likelihood family in full

We repeat the comparison with Eq. 3, holding the probe, the split and the budget fixed. Both arms use the optimiser, the precision, the clipping and the constant learning rate 5×10^{-5} of Table 5, and keep the retain term of Eq. 2 at the same ℓ^* with the same α; in Eq. 3 $\beta = 1$, the margin d is averaged over the tokens of a block, and one step accumulates 4 forget blocks of 512 tokens, each paired with one retain block. Training stops when the median of d over 32 held-out probe blocks, disjoint from the training blocks, reaches -3 nats per token, and the forced reading is taken whenever natFA crosses one of 4 grid values between 0.400 and 0.325, probed every 3 steps; all ten runs stopped at that criterion, and none became worse than uniform on the forget corpus. The affinity arm passes 0.400 between two probes in every run, leaving that value without a paired reading, so Table 7 reports the remaining 3.

The ordering does not transfer. Panel (b) of Table 7 reports this family in the form of §5.2. The forced reading does not resolve: our arm is *higher* than the affinity arm, but the runs split on that sign; read as the strongest single run per arm, the affinity arm is lower, and the same direction holds at 8 of the 9 probe budgets taken separately. Depth is also matched more coarsely than in the representation panel: each forced reading is taken at a single probe step within half a grid step (0.0125) of its grid value rather than interpolated onto it, and the affinity arm, which moves further between probes (see the speed ratio below), is read slightly deeper: natFA 0.353 against 0.360 for ours (Table 4). By Corollary 2 the deeper arm reads lower under forcing, and in these runs the affinity arm's retain accuracy rises with depth (Table 7), so on both columns the offset favours the affinity arm. We declare it rather than correct it, which would need a forced reading at every probe. The utility reading resolves against us: at matched depth the retain-accuracy difference, ours minus affinity, is -0.026, with every run agreeing in sign. The ordering of §5.2 therefore fails to reproduce and is paid for in utility; it is not reversed at any level these runs resolve. That the two families order the arms in opposite directions licenses the statement that the choice of selection signal and of objective family are not independent, not a comparison of the two families' absolute numbers, which we never make.

Decomposing the arm speed ratio. The two arms need 70 and 29 steps to reach the same margin, a ratio of 2.4×. The median ratio of retain-term to forget-term gradient norm is 0.61 for our arm and 0.14 for the affinity arm, and the cosine between the two gradients stays within a few hundredths of zero, so once the sum is clipped to unit norm the forget-direction fraction of a step is $1/\sqrt{1+r^2}$; its ratio between the arms is 1.16×. Dividing leaves 2.1× unexplained by the anchor, against a gate-mass ratio of 2.25× measured on the same two expert sets without reference to any training run. This is the mechanism of §3.2: an expert's effective step size tracks its share of gate mass on the forget corpus, and our signal selects the low-flow tail by construction (App. C.3). Cumulative parameter displacement per step differs between the arms by far less than the speed ratio, as a per-parameter-normalised optimiser predicts, and in the deep regime the probe distribution is bimodal, so the mean margin reported during training is a mixture rather than a state. One confound is ours to declare: both arms use the same learning rate, so our arm reaches each depth after roughly 2.4× as many steps, and its extra retain cost at matched depth cannot be separated from those extra steps by this experiment. Calibrating the learning rate per arm would separate them; we did not run it.

E LIMITATIONS

Two limitations of our current study also suggest directions for future work.

First, our evaluation is limited to a small set of MoE architectures, forget domains, and expert budgets. Future work could extend the study to larger and structurally different MoE models, broader forget domains, and multimodal MoE models, where routing depends on both visual and textual representations. Scaling the expert coverage would also help test whether the observed ordering generalizes beyond the current setting.

Second, our probe explores only a restricted set of recovery strategies by fixing the intervened layers and varying the backup-expert budget. Future work could consider adaptive routing attacks and stronger recovery procedures that jointly search over layers and experts. Another direction is to study post-unlearning adaptation, such as LoRA or lightweight fine-tuning, to examine whether residual forget capability can be recovered through subsequent training.