TARGETED ROUTING-AWARE CALIBRATION FOR MACHINE UNLEARNING IN MIXTURE-OF-EXPERTS LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Machine unlearning aims to remove the undesired data from large language models (LLMs) while preserving model utility. As Mixture-of-Experts (MoE) LLMs become prevalent, there is a growing need for effective unlearning methods that account for their distinctive architecture. In this work, we observe that a small subset of experts has substantially higher activation frequencies on forget data than the other experts, while generic retain data activates these experts much less frequently. These observations motivate considering both which experts to update and how to direct retain regularization toward them. We propose **TRACE**, **T**argeted **R**outing-**A**ware **C**alibration for **E**xperts, a two-stage framework for MoE unlearning. TRACE first selects experts with high activation frequencies on a small calibration subset of forget data. During unlearning, it reweights the retain loss to prioritize retain samples whose activation frequencies on the selected experts more closely match those of the forget data, providing targeted retain regularization. Experiments across multiple MoE LLMs show that TRACE preserves substantially higher utility at comparable forgetting levels, including a 4.55-point MMLU improvement over the strongest external baseline on Qwen1.5-MoE-A2.7B-Chat and a 3.96-point improvement on DeepSeek-V2-Lite-Chat.

1 INTRODUCTION

Large language models (LLMs) (Yang et al., 2025; DeepSeek-AI et al., 2025; OpenAI et al., 2024) have achieved remarkable success and are increasingly deployed across a wide range of applications. However, LLMs may memorize undesired information, including harmful (Li et al., 2024), copyrighted (Eldan & Russinovich, 2023), or private (Jin et al., 2024; Maini et al., 2024) content, thereby necessitating machine unlearning to remove the influence of forget data while preserving the model's utility. To avoid the prohibitive cost of retraining from scratch, existing unlearning approaches typically optimize a forgetting objective together with regularization on auxiliary retain data, which serves to mitigate the utility degradation induced by forgetting (Yao et al., 2024; Zhang et al., 2024; Liu et al., 2022; Fan et al., 2025).

As Mixture-of-Experts (MoE) architectures (Mu & Lin, 2026; DeepSeek-AI et al., 2025; Jiang et al., 2024; Lepikhin et al., 2020; Fedus et al., 2022) become prevalent in modern LLMs, effective unlearning in this setting has become increasingly important. However, most existing unlearning methods have been developed and evaluated primarily for dense Transformer architectures (Vaswani et al., 2017) and their effectiveness for MoE-based LLMs remains underexplored. Unlike dense Transformers, which employ a single feed-forward network at each layer, MoE architectures replace it with multiple experts and a router that dynamically selects the top-K experts for each token. As a result, directly applying unlearning methods designed for dense LLMs to MoE models may fail to account for the expert specialization and routing mechanisms unique to MoE architectures.

Sparse top-K routing shapes MoE unlearning in two coupled ways: it determines which experts participate in processing forget data and how retain regularization reaches those experts. Existing expert-targeted approaches (Zhuang et al., 2025) emphasize locating the expert with highest affinity to forget data, leaving open how actual routing patterns should guide both expert selection and utility preservation. Addressing these questions requires examining expert participation on both

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

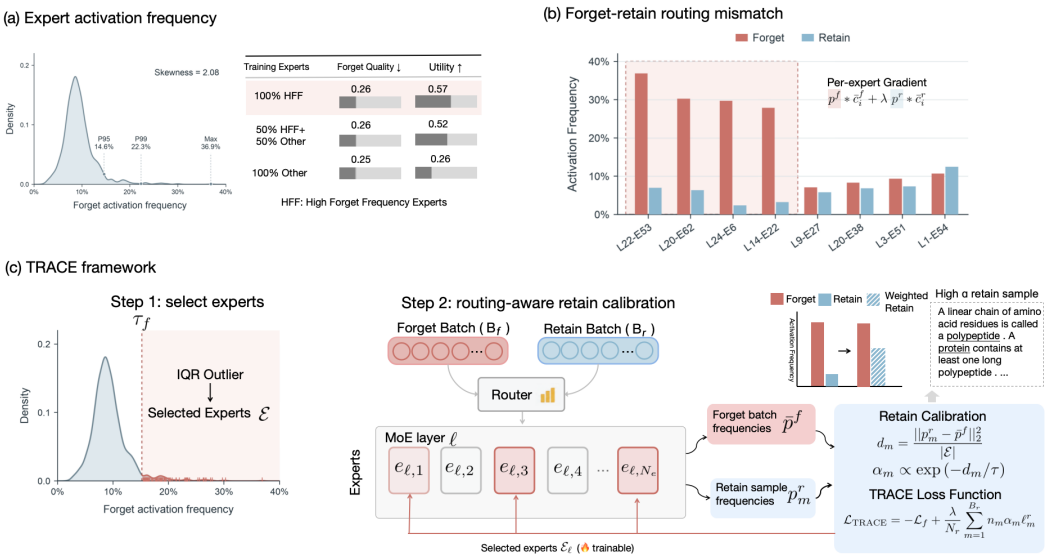


Figure 1: **Motivation and overview of TRACE.** (a) Expert activation frequencies on forget data exhibit a long-tailed distribution. Targeting high forget frequency (HFF) experts preserves higher utility at comparable forgetting quality. Other experts are sampled uniformly without replacement from the non-HFF expert set. (b) Generic retain data activates these experts much less frequently, revealing a forget–retain routing mismatch. (c) TRACE first selects HFF experts $\mathcal{E}$. During unlearning, it updates only the selected experts and redistributes retain loss weights toward samples whose activation frequencies on these experts are closer to those of the current forget batch, providing more targeted retain regularization.

the forget and retain sides. We identify a **forget–retain routing mismatch** through an analysis of expert activation frequencies. Forget data disproportionately activates a subset of experts but generic retain data often activates the same experts much less frequently, creating a mismatch between their exposure to forgetting and retain regularization. Thus, experts heavily involved in processing forget data may receive limited retain regularization, which selecting the appropriate experts alone does not resolve.

Motivated by these observations, we propose TRACE, **T**argeted **R**outing-**A**ware **C**alibration for **E**xperts, for MoE unlearning (Figure 1(c)). TRACE first detects high forget frequency experts using activation statistics collected on a small calibration set of forget data. It then redistributes retain loss weights to prioritize samples whose activation frequencies on the selected experts are closer to those of the forget data, providing more targeted retain regularization. By coupling expert selection with retain loss reweighting, TRACE enables effective unlearning while better preserving model utility.

Our contributions are summarized as follows:

- We identify a pronounced heterogeneity in expert activation frequencies and uncover a forget–retain routing mismatch in MoE LLMs, where experts frequently activated by forget data are substantially less activated by generic retain data.
- We propose TRACE, a two-stage framework that uses expert activation statistics to guide both expert selection and retain-loss reweighting for utility preservation in MoE LLMs.
- We demonstrate across multiple MoE LLMs on WMDP and MUSE-BOOKS that TRACE consistently preserves higher utility than existing baselines at comparable forgetting levels, while remaining effective even with generic retain data.
- Our ablation studies further show that routing-aware retain-loss reweighting improves utility preservation, particularly on subjects closely related to the forget domains.

2 Related Work

2.1 Machine unlearning

A growing line of work has investigated how to formulate, evaluate, and improve unlearning performance for LLMs (Geng et al., 2025; Liu et al., 2024b; Zhang et al., 2024; Fan et al., 2025; Jia et al., 2025; Lu et al., 2022; Liu et al., 2024a; Wang et al., 2023; Fang et al., 2026; Barbulescu & Triantafillou, 2024; Chen & Yang, 2023). Recent benchmarks instantiate LLM unlearning under different removal targets, including privacy information (Maini et al., 2024; Jin et al., 2024), copyrighted or memorized content (Shi et al., 2024; Eldan & Russinovich, 2023), and hazardous domain knowledge (Li et al., 2024). These benchmarks evaluate not only unlearning effectiveness, but also utility preservation. Existing methods typically address these objectives through different mechanisms, including gradient-based optimization (Zhang et al., 2024; Fan et al., 2025; Yu et al., 2023; Zhang et al., 2023; Rafailov et al., 2024), localization-informed editing (Jia et al., 2025; Fang et al., 2026; Patil et al., 2023; Meng et al., 2023; Wu et al., 2023), and input-based methods (Liu et al., 2024a; Madaan et al., 2023; Pawelczyk et al., 2024; Muresanu et al., 2025). Specifically, gradient-based methods directly optimize model parameters with forget and retain objectives, localization-informed methods first identify model components associated with target knowledge and then unlearn, while input-based methods modify the model behavior through the input context. However, most of them are designed and evaluated on dense transformer models, without considering sparse MoE LLMs.

2.2 MoE LLMs and MoE unlearning

MoE LLMs have become an important architecture for scaling large language models by increasing model capacity without a proportional increase in inference cost (DeepSeek-AI et al., 2024; 2025; Du et al., 2022; Jiang et al., 2024; Fedus et al., 2022; Lepikhin et al., 2020). Unlike dense models that activate the same set of parameters for every input, MoE LLMs use a router to assign each token to a sparse subset of experts. Recent work has begun to exploit the modular structure of MoE LLMs for unlearning. SEUF (Zhuang et al., 2025) selects the expert with the largest affinity score to the forget data and updates the selected expert together with its corresponding router. ESFT (Wang et al., 2024), originally proposed for expert-specialized fine-tuning, selects top-ranked experts by cumulative affinity per layer and has also been adopted as an expert-selection baseline for MoE unlearning. Prior MoE unlearning methods primarily focus on where to localize forgetting and, in the case of SEUF, how to stabilize routing during unlearning. In contrast, TRACE identifies a complementary retain-side problem: sparse routing can prevent generic retain data from sufficiently regularizing the very experts being updated for forgetting.

3 Expert Activation Frequencies and Retain Regularization

In this section, we examine how expert activations inform expert selection and retain regularization in MoE unlearning. Let $\mathcal{D}_f$ and $\mathcal{D}_r$ denote the forget and retain sets, respectively. A broad class of unlearning methods optimizes a forget objective regularized by a retain loss:

$$\mathcal{L}(\boldsymbol{\theta}) = -\mathcal{L}_f(\boldsymbol{\theta}; \mathcal{D}_f) + \lambda \mathcal{L}_r(\boldsymbol{\theta}; \mathcal{D}_r),$$

where $\mathcal{L}_f$ and $\mathcal{L}_r$ are next token negative log-likelihood losses, and $\lambda > 0$ trades off forgetting the targets in $\mathcal{D}_f$ against preserving the model utility in $\mathcal{D}_r$.

Consider a MoE LLM with top-K routing. Let $e_{j,\ell}$ denote the j-th expert of layer ℓ, and let N_f and N_r denote the numbers of valid token positions in the forget and retain sets, respectively. Let $\mathcal{S}_i^f$ and $\mathcal{S}_i^r$ denote the sets of experts selected for the i-th forget and retain token positions. We define the forget and retain activation frequencies of the expert $e_{j,\ell}$ as

$$p_{j,\ell}^f = \frac{\sum_{i=1}^{N_f} \mathbf{1}\left[e_{j,\ell} \in \mathcal{S}_i^f\right]}{N_f} \qquad p_{j,\ell}^r = \frac{\sum_{i=1}^{N_r} \mathbf{1}[e_{j,\ell} \in \mathcal{S}_i^r]}{N_r}. \tag{1}$$

These frequencies measure how often each expert participates in processing the corresponding data.

As shown in Figure 1(a), forget-side activation frequencies exhibit a long-tailed distribution, with a small subset of experts having substantially higher activation frequencies than the other experts.

This observation motivates focusing unlearning updates on experts frequently involved in processing forget data. The expert-selection comparison in Figure 1(a) further shows that updating high forget frequency experts preserves higher utility than updating the same number of other experts at comparable forgetting quality.

Figure 1(b) shows that generic retain data activates experts with high forget activation frequencies much less frequently than forget data. We refer to this difference in expert participation as a **forget–retain routing mismatch**. Under the assumptions detailed in Appendix B, the GradDiff gradient for expert $e_{j,\ell}$ can be written as

$$\nabla_{\boldsymbol{\theta}_{j,\ell}} \mathcal{L} = -p^f_{j,\ell} \bar{\mathbf{c}}^f_{j,\ell} + \lambda p^r_{j,\ell} \bar{\mathbf{c}}^r_{j,\ell}, \tag{2}$$

where $\bar{\mathbf{c}}^f_{j,\ell}$ and $\bar{\mathbf{c}}^r_{j,\ell}$ denote the average local gradient contributions per activation of expert $e_{j,\ell}$ on the forget and retain data, respectively. Holding the average retain contribution fixed, a lower retain activation frequency reduces the retain gradient component. This motivates prioritizing retain samples whose activation frequencies on the selected experts are closer to those of the current forget batch. Section 4.2 presents routing-aware retain calibration, which implements this idea through normalized sample weights to mitigate utility degradation during unlearning.

4 TRACE: Targeted Routing-Aware Calibration for Experts

Guided by the two implications above, we propose TRACE, a targeted routing-aware expert calibration framework for MoE unlearning. The goal is to focus unlearning on experts most associated with the forget data while providing these experts with sufficient retain regularization. To achieve this, TRACE consists of two stages. First, it selects experts using routing activation statistics. Second, it calibrates retain regularization by reweighting sample-wise retain losses.

4.1 Select Experts

We first identify experts that are disproportionately activated by the forget data. For each expert $e_{j,\ell}$, we compute its forget-side activation frequency $p^f_{j,\ell}$ using equation 1. We then select a set of experts whose forget activation frequencies are unusually high. Specifically, we use the interquartile range (IQR) rule (Heckert et al., 2002) to define an adaptive activation threshold. Let $Q_1(\cdot)$ and $Q_3(\cdot)$ denote the first and third quartiles over the collection of expert activation frequencies $\{p^f_{j,\ell}\}$, and $\mathrm{IQR}(z) = Q_3(z) - Q_1(z)$. We define the threshold as

$$\tau_f = Q_3\left(p^f_{j,\ell}\right) + \lambda_f \,\mathrm{IQR}\left(p^f_{j,\ell}\right), \tag{3}$$

where λ_f controls the strictness of outlier selection. The selected expert set is

$$\mathcal{E} = \{(j,\ell) : p^f_{j,\ell} \geq \tau_f\}. \tag{4}$$

This adaptive rule allows TRACE to select multiple forget-relevant experts when the forget data activates a distributed expert subset, while avoiding unnecessary updates to experts weakly associated with the forget data.

4.2 Routing-Aware Retain Reweighting

After selecting $\mathcal{E}$, TRACE redistributes retain loss weights according to the distance between each retain sample's expert activation frequencies and those of the current forget batch. All activation frequency vectors below are restricted to $\mathcal{E}$.

For each retain batch, let B_r denote the number of retain samples and n_m the number of valid tokens in sample m, with $N_r = \sum_{m=1}^{B_r} n_m$.

Expert activation frequencies. We compare the expert activation frequencies of each retain sample with those computed over the current forget batch. For retain sample m, let $p^r_{m,j,\ell}$ denote the activation frequency of expert $e_{j,\ell}$ on that sample. The corresponding forget activation frequency

$p^f_{j,\ell}$ is computed over all samples in the current forget batch, as defined in equation 1. Restricting these frequencies to the selected experts $\mathcal{E}$, we obtain

$$\mathbf{p}^r_m = \left(p^r_{m,j,\ell}\right)_{(j,\ell)\in\mathcal{E}}, \qquad \bar{\mathbf{p}}^{\,f} = \left(p^f_{j,\ell}\right)_{(j,\ell)\in\mathcal{E}}. \tag{5}$$

Distance-based sample weights. We measure the routing discrepancy between retain sample m and the current forget batch by

$$d_m = \frac{1}{|\mathcal{E}|}\left\|\mathbf{p}^r_m - \bar{\mathbf{p}}^{\,f}\right\|_2^2. \tag{6}$$

For temperature $\tau > 0$, we assign the sample weight

$$\alpha_m = \frac{N_r \exp(-d_m/\tau)}{\sum_{m'=1}^{B_r} n_{m'} \exp(-d_{m'}/\tau)}. \tag{7}$$

Within a batch, a smaller distance gives a larger α_m, with smaller τ placing greater emphasis on samples with closer activation frequencies. The normalization ensures $\frac{1}{N_r}\sum_{m=1}^{B_r} n_m\alpha_m = 1$. Thus, TRACE redistributes retain-loss weights while maintaining a token-weighted mean of one.

Sample-wise retain objective. Let $\ell^r_m(\boldsymbol{\theta})$ denote the negative log-likelihood loss of retain sample m. The weighted retain objective is

$$\mathcal{L}^{(\alpha)}_r(\boldsymbol{\theta}) = \frac{1}{N_r}\sum_{m=1}^{B_r} n_m \,\mathrm{sg}(\alpha_m)\,\ell^r_m(\boldsymbol{\theta}), \tag{8}$$

where sg denotes the stop-gradient operator. The factor n_m/N_r preserves the length normalization of the original retain objective. The resulting unlearning objective is

$$\begin{aligned} &\mathcal{L}_{\mathrm{TRACE}}(\boldsymbol{\theta}) = -\mathcal{L}_f(\boldsymbol{\theta}) + \lambda\mathcal{L}^{(\alpha)}_r(\boldsymbol{\theta}). \\ &\text{subject to } \ \boldsymbol{\theta}_{j,l} = \boldsymbol{\theta}^{(\mathrm{ptr})}_{j,l} \ \ \forall (j,l) \notin \mathcal{E}, \end{aligned} \tag{9}$$

where $\boldsymbol{\theta}^{(\mathrm{ptr})}$ denotes the pre-trained model parameters, which are fixed during unlearning. The overall procedure is summarized in Algorithm 1.

5 Experiments

We evaluate TRACE on WMDP (Li et al., 2024) and MUSE-BOOKS (Shi et al., 2024) across multiple MoE architectures. We assess unlearning performance from both forgetting effectiveness and model utility.

5.1 Experimental Setups

Datasets. To demonstrate the effectiveness of our method, we conduct experiments on two benchmarks: WMDP and MUSE-BOOKS. WMDP aims to remove hazardous knowledge, including content related to Cybersecurity (Cyber) and Biosecurity (Bio), with WikiText (Merity et al., 2016) used as the retain dataset. MUSE-BOOKS comprises text extracted from the Harry Potter series, focusing on removing the copyrighted content, while the corresponding retain dataset is sampled from Harry Potter Fandom Wiki.

MoE LLMs. We evaluate different unlearning methods on three MoE LLMs: Qwen1.5-MoE-A2.7B-Chat (Team, 2024), DeepSeek-V2-Lite-Chat (DeepSeek-AI et al., 2024), and Qwen3-30B-A3B (Yang et al., 2025). Following Zhuang et al. (2025), we evaluate Qwen1.5-MoE-A2.7B-Chat and DeepSeek-V2-Lite-Chat for WMDP. For MUSE-BOOKS, we use Qwen3-30B-A3B. To ensure that the model has learned the relevant knowledge from both the forget and retain sets before unlearning, we first fine-tune it on these datasets. The resulting checkpoint serves as the original model for all unlearning methods. Specifically, Qwen1.5-MoE-A2.7B-Chat contains 14.3B total parameters, with 2.7B activated during inference; DeepSeek-V2-Lite contains 15.7B total parameters, with 2.4B activated; and Qwen3-30B-A3B contains 30.5B total parameters, with 3.3B activated.

Metrics. We evaluate each method in terms of both forgetting quality and retaining utility on two benchmarks. For WMDP, we report answer accuracy on WMDP-Cyber and WMDP-Bio to evaluate forgetting quality, with 25% serving as the random-guessing reference. We report answer accuracy on MMLU (Hendrycks et al., 2021) to assess general utility. Following WMDP (Li et al., 2024), we additionally report accuracy on four MMLU subjects in the ablation: college biology, virology, college computer science, and computer security. These subjects provide a focused assessment of general knowledge related to the forget domains, as WMDP reports substantial degradation in virology and computer security despite largely preserved overall MMLU accuracy. For MUSE-BOOKS, we evaluate forgetting quality using VerbMem and KnowMem metrics on the forget set. VerbMem measures verbatim memorization, whereas KnowMem measures knowledge memorization. Both memorization metrics are evaluated using ROUGE-L. For utility, we report KnowMem on the retain set, also measured by ROUGE-L.

Baselines. We compare TRACE with GA, GradDiff (Maini et al., 2024), and Negative Preference Optimization (NPO) (Zhang et al., 2024). We also compare with SEUF (Zhuang et al., 2025) and ESFT (Wang et al., 2024), which restrict unlearning updates to selected experts. SEUF selects the single expert with the highest affinity to the forget set and updates it together with its corresponding router, that has the maximum average affinity score on the calibration set of forget data. In contrast, ESFT ranks experts by their average affinity scores within each layer and selects experts until their cumulative affinity exceeds a predefined threshold p.

5.2 Overall Performance

WMDP. Table 2 reports the main results. TRACE attains MMLU accuracies of 57.74% on Qwen1.5-MoE-A2.7B-Chat and 55.00% on DeepSeek-V2-Lite-Chat, the highest among the evaluated unlearning methods, while reducing WMDP averages to 26.56% and 25.90%, respectively. Relative to the original models, the MMLU decreases are 1.80 and 1.72 percentage points. Compared with ESFT, the external baseline with the highest MMLU accuracy in this table, TRACE improves MMLU by 4.55 points on Qwen1.5-MoE-A2.7B-Chat and 3.96 points on DeepSeek-V2-Lite-Chat. These correspond to relative improvements of 8.55% and 7.76%. The WMDP averages are also lower than ESFT's 28.27% and 25.94%. Compared with GA, GradDiff, and NPO, TRACE preserves substantially more general utility.

MUSE-BOOKS. Table 1 shows that TRACE achieves zero forget VerbMem and a forget KnowMem score of 2.60, while preserving the highest retain KnowMem score among the evaluated unlearning methods at 69.52. Compared with SEUF, the external baseline with the highest retain KnowMem, TRACE improves retained utility by 2.82 points while reducing forget KnowMem from 15.73 to 2.60. Although GA and GradDiff achieve lower forget KnowMem scores, their retain KnowMem drops to 0.00 and 43.92, respectively. TRACE also improves over TRACE-Select, reducing forget KnowMem from 9.05 to 2.60 and increasing retain KnowMem from 68.16 to 69.52, while keeping forget VerbMem at zero.

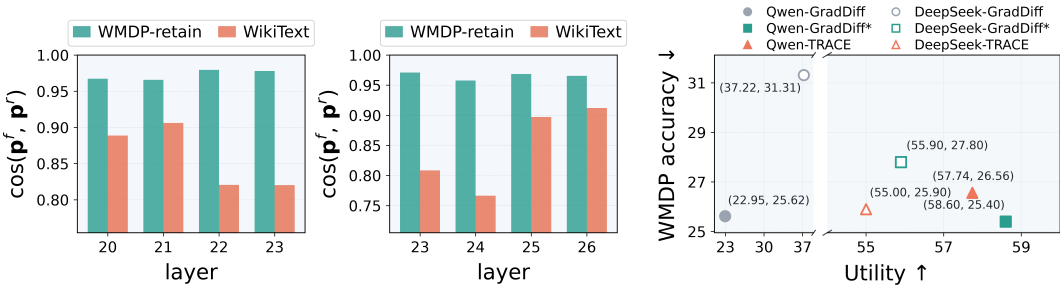
Table 1: Results and ablation on MUSE-BOOKS with Qwen3-30B-A3B. f/r: forget/retain. Bold marks the best result in each column.

Method	Forgetting		Utility
	VerbMem$_f$ ↓	KnowMem$_f$ ↓	KnowMem$_r$ ↑
original	24.60	40.79	74.41
GA	**0.00**	0.00	0.00
GradDiff	**0.00**	**1.10**	43.92
NPO	**0.00**	8.93	64.92
SEUF	2.08	15.73	66.70
ESFT	0.32	17.56	65.71
TRACE-Select	0.00	9.05	68.16
TRACE	**0.00**	2.60	**69.52**

Comparison with Domain-Specific Retain Data. We additionally compare with GradDiff*, which uses task-curated WMDP-retain data as the retain set instead of WikiText. Figure 2(a,b) shows that WMDP-retain more closely matches the forget data in expert activation frequencies, with higher cosine similarity than WikiText in the last four MoE layers of both models. As shown in Figure 2(c), TRACE achieves MMLU accuracy close to GradDiff*, trailing by 0.86 percentage points on Qwen and 0.90 on DeepSeek. Its WMDP average is higher on Qwen (26.56% versus 25.40%) and lower on DeepSeek (25.90% versus 27.80%). These results suggest that TRACE can

Table 2: Main results on WMDP. All values are accuracy percentages; Avg. is the mean of Bio and Cyber. The random-guessing reference for WMDP is 25%. Bold denotes the best performance among unlearning methods. TRACE-Select uses the same selected experts as TRACE without retain reweighting.

Method	Qwen1.5-MoE-A2.7B-Chat				DeepSeek-V2-Lite-Chat			
	Cyber	Bio	Avg.	MMLU ↑	Cyber	Bio	Avg.	MMLU ↑
Original	42.43	65.36	53.90	59.54	39.46	58.60	49.03	56.72
GA	29.64	26.47	28.06	32.17	29.29	27.02	28.16	33.07
GradDiff	26.57	24.67	**25.62**	22.95	29.94	32.67	31.31	37.22
NPO	26.87	24.74	25.81	24.81	22.46	25.31	**23.89**	24.66
SEUF	25.45	26.70	26.08	49.15	30.50	26.00	28.25	45.53
ESFT	29.99	26.55	28.27	53.19	26.90	24.98	25.94	51.04
TRACE-Select	30.09	24.74	27.42	57.17	25.97	24.66	25.31	53.93
TRACE	27.83	25.29	26.56	**57.74**	26.27	25.53	25.90	**55.00**



(a) Qwen1.5-MoE-A2.7B-Chat (b) DeepSeek-V2-Lite-Chat (c) Forgetting and utility

Figure 2: Comparison with task-curated retain data. (a,b) Cosine similarity between forget and retain activation frequencies in the last four MoE layers of Qwen1.5-MoE-A2.7B-Chat and DeepSeek-V2-Lite-Chat, respectively. (c) MMLU accuracy versus WMDP accuracy. GradDiff and TRACE use WikiText, whereas GradDiff* uses WMDP-retain. All accuracy values are reported as percentages.

preserve high utility using readily available generic retain data, reducing the need to collect and curate domain-specific retain datasets and the associated costs.

5.3 Ablation Study

Expert selection. Table 2 shows that TRACE-Select preserves higher MMLU accuracy than GA, GradDiff, NPO and the expert selection baselines SEUF and ESFT on both models. Compared with ESFT, TRACE-Select improves MMLU by 3.98 percentage points on Qwen1.5-MoE-A2.7B-Chat and 2.89 on DeepSeek-V2-Lite-Chat, while achieving lower WMDP-average accuracy on both models. These results suggest that selecting experts with unusually high forget activation frequencies and identifying these experts through IQR outlier detection across all experts helps effective unlearning.

Retain calibration. We further examine retain calibration at comparable forgetting levels in Table 3. On Qwen1.5-MoE-A2.7B-Chat at step 200, TRACE improves MMLU by 0.86 percentage points over TRACE-Select, while slightly reducing WMDP-average accuracy from 27.42% to 27.30%. On DeepSeek-V2-Lite-Chat, TRACE at step 200 achieves 54.41% MMLU accuracy compared with 54.04% for TRACE-Select at step 250, with both WMDP averages around 25.6%. The utility gains are more pronounced on several MMLU subjects related to the forget domains. On Qwen1.5-MoE-A2.7B-Chat, TRACE improves college biology, virology, college computer science, and computer security by 8.34, 7.84, 3.0, and 5.0 percentage points, respectively. On DeepSeek-V2-Lite-Chat, the gains are 1.39, 2.0, and 3.0 points for college biology, college computer science,

Table 3: Retain calibration at comparable WMDP-average accuracy. Scores are accuracy percentages. CB, CCS, and CS denote college biology, college computer science, and computer security.

Method	Step	WMDP			Utility ↑				
		Bio	Cyber	Avg.	MMLU	CB	Virology	CCS	CS
Qwen1.5-MoE-A2.7B-Chat									
TRACE-Select	200	24.74	30.09	27.42	57.17	47.91	19.87	43.00	44.00
TRACE	200	25.92	28.69	**27.30**	**58.03** (+0.86)	**56.25** (+8.34)	**27.71** (+7.84)	**46.00** (+3.0)	**49.00** (+5.0)
DeepSeek-V2-Lite-Chat									
TRACE-Select	250	24.58	26.52	25.55	54.04	56.94	19.27	44.00	31.00
TRACE	200	24.74	26.42	25.58	**54.41** (+0.37)	**58.33** (+1.39)	19.27	**46.00** (+2.0)	**34.00** (+3.0)

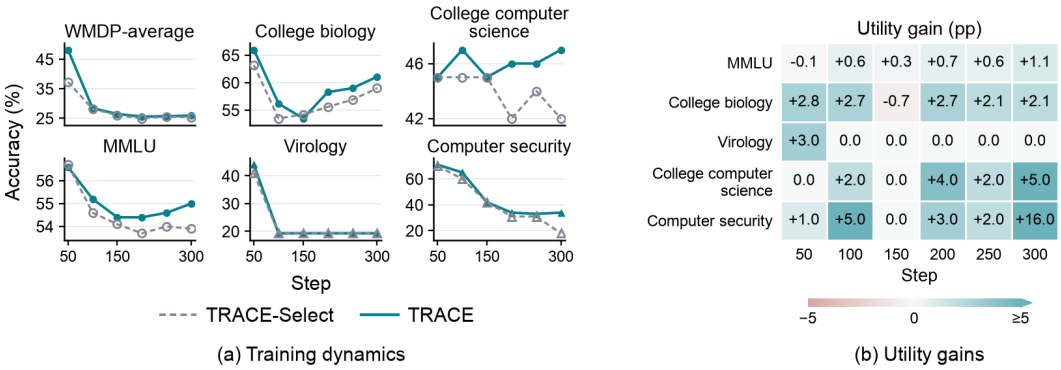


Figure 3: Ablation of retain reweighting on DeepSeek-V2-Lite-Chat. (a) Training dynamics of TRACE and TRACE-Select; WMDP-average is the mean of Bio and Cyber accuracy. (b) Utility gains of TRACE over TRACE-Select in percentage points.

and computer security, respectively, while virology remains unchanged. These results suggest that retain calibration helps mitigate the loss of general knowledge closely related to the forget domains.

Training dynamics. We report the training dynamics in Figure 3 to examine how retain calibration affects forgetting and utility during training. On DeepSeek-V2-Lite-Chat, TRACE initially forgets more slowly, but maintains higher MMLU accuracy from step 100 onward. At later checkpoints, the two methods reach comparable WMDP accuracy, while TRACE still preserves higher MMLU accuracy. The full trajectories show that the utility benefit of retain calibration appears across multiple training stages, complementing the checkpoint comparisons in Table 3. The training dynamics on Qwen1.5-MoE-A2.7B-Chat are presented in Figure 4.

5.4 Sensitivity Analysis

Sensitivity to the expert selection threshold. We evaluate $\lambda_f \in \{1.5, 3\}$, two standard choices for IQR outlier detection. Setting $\lambda_f = 1.5$ selects both mild and extreme outliers in forget activation frequencies, whereas $\lambda_f = 3$ selects only extreme outliers. This stricter criterion reduces the number of selected experts from 52 to 17. On Qwen1.5-MoE-A2.7B-Chat, changing λ_f from 1.5 to 3 changes WMDP average accuracy by only 0.24 percentage points and MMLU accuracy by 0.04 percentage points, suggesting that TRACE is relatively insensitive to the choice between these two standard thresholds.

Table 4: Sensitivity to λ_f on Qwen1.5-MoE-A2.7B-Chat. Experts denotes the number of selected experts.

λ_f	Experts	Bio ↓	Cyber ↓	Avg. ↓	MMLU ↑
1.5	52	25.29	27.83	26.56	57.74
3.0	17	25.06	28.54	26.80	57.78

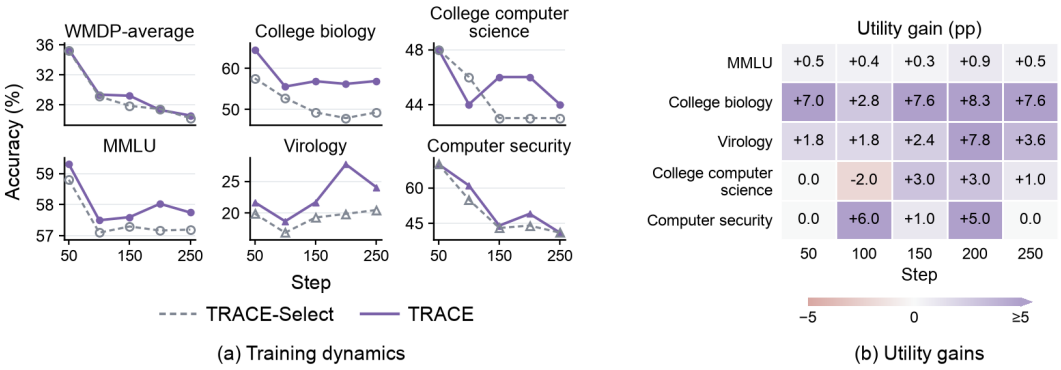


Figure 4: Ablation of retain reweighting on Qwen1.5-MoE-A2.7B-Chat. (a) Training dynamics of TRACE and TRACE-Select; WMDP-average is the mean of Bio and Cyber accuracy. (b) Utility gains of TRACE over TRACE-Select in percentage points.

Sensitivity to temperature τ. We evaluate $\tau \in \{0.001, 0.002, 0.01, 0.05, 0.1\}$ on Qwen1.5-MoE-A2.7B-Chat using WMDP, while keeping the other settings fixed. A smaller τ concentrates retain weights on samples with smaller routing distances, while a larger τ makes the weights more uniform. Figure 5 shows relatively stable utility across the tested range. MMLU accuracy varies by only 0.65 percentage points, while WMDP average accuracy ranges from 26.22% to 27.99%.

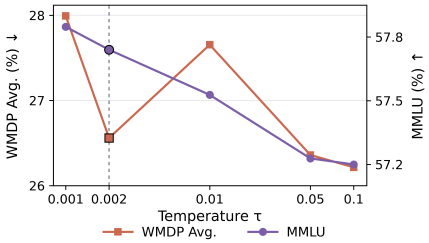


Figure 5: Temperature sensitivity on Qwen1.5-MoE-A2.7B-Chat.

5.5 Efficiency Analysis

TRACE collects expert activation frequencies through forward hooks during the existing forget and retain forward passes, requiring no additional forward pass. Given these frequencies, computing the squared distances and normalized retain weights takes $O(B_r|\mathcal{E}|)$ time. We measure training time per optimizer step on NVIDIA A800 GPUs, using a microbatch size of 4 and 4 gradient accumulation steps. Compared with TRACE-Select, retain calibration adds 2.4 seconds per optimizer step, corresponding to a 17.5% overhead, including routing-statistics collection, weight computation, and weighted retain-loss evaluation.

6 Conclusion

We study machine unlearning for MoE LLMs and identify a forget–retain routing mismatch: experts frequently activated by forget data may receive insufficient regularization from generic retain data. To address this mismatch, we propose TRACE, which selects experts with unusually high activation frequencies on forget data and applies routing-aware retain calibration during unlearning. TRACE redistributes retain-loss weights toward samples whose routing patterns on the selected experts are closer to those of the forget data, thereby providing more targeted regularization. Experiments on WMDP and MUSE-BOOKS across multiple MoE LLMs show that TRACE improves the utility–forgetting trade-off over existing baselines and better preserves model utility at comparable forgetting levels. Our results further show that routing-aware retain calibration is particularly beneficial for preserving knowledge related to the forget domains.

AI use statement

During the preparation of this work, we used generative AI tools, including Codex, to assist with manuscript polish, literature searches, mathematical formulations, and figure layout design. The authors take full responsibility for the final manuscript, experimental procedures, reported results, and conclusions, including all content produced with AI assistance.

REFERENCES

George-Octavian Barbulescu and Peter Triantafillou. To each (textual sequence) its own: Improving memorized-data unlearning in large language models, 2024. URL https://arxiv.org/abs/2405.03097.

Jiaao Chen and Diyi Yang. Unlearn what you want to forget: Efficient unlearning for llms, 2023. URL https://arxiv.org/abs/2310.20150.

DeepSeek-AI, Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Hao Yang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jin Chen, Jingyang Yuan, Junjie Qiu, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruizhe Pan, Runxin Xu, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Size Zheng, T. Wang, Tian Pei, Tian Yuan, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Liu, Xin Xie, Xingkai Yu, Xinnan Song, Xinyi Zhou, Xinyu Yang, Xuan Lu, Xuecheng Su, Y. Wu, Y. K. Li, Y. X. Wei, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Zheng, Yichao Zhang, Yiliang Xiong, Yilong Zhao, Ying He, Ying Tang, Yishi Piao, Yixin Dong, Yixuan Tan, Yiyuan Liu, Yongji Wang, Yongqiang Guo, Yuchen Zhu, Yuduan Wang, Yuheng Zou, Yukun Zha, Yunxian Ma, Yuting Yan, Yuxiang You, Yuxuan Liu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhewen Hao, Zhihong Shao, Zhiniu Wen, Zhipeng Xu, Zhongyu Zhang, Zhuoshu Li, Zihan Wang, Zihui Gu, Zilin Li, and Ziwei Xie. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model, 2024. URL https://arxiv.org/abs/2405.04434.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li,

Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. Deepseek-v3 technical report, 2025. URL https://arxiv.org/abs/2412.19437.

Nan Du, Yanping Huang, Andrew M. Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, Barret Zoph, Liam Fedus, Maarten Bosma, Zongwei Zhou, Tao Wang, Yu Emma Wang, Kellie Webster, Marie Pellat, Kevin Robinson, Kathleen Meier-Hellstern, Toju Duke, Lucas Dixon, Kun Zhang, Quoc V Le, Yonghui Wu, Zhifeng Chen, and Claire Cui. Glam: Efficient scaling of language models with mixture-of-experts, 2022. URL https://arxiv.org/abs/2112.06905.

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

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

Ce Fang, Zhikun Zhang, Min Chen, Qing Liu, Lu Zhou, Zhe Liu, and Yunjun Gao. Kuda: Knowledge unlearning by deviating representation for large language models, 2026. URL https://arxiv.org/abs/2602.19275.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity, 2022. URL https://arxiv.org/abs/2101.03961.

Jiahui Geng, Qing Li, Herbert Woisetschlaeger, Zongxiong Chen, Fengyu Cai, Yuxia Wang, Preslav Nakov, Hans-Arno Jacobsen, and Fakhri Karray. A comprehensive survey of machine unlearning techniques for large language models, 2025. URL https://arxiv.org/abs/2503.01854.

N. Heckert, James Filliben, C Croarkin, B Hembree, William Guthrie, P Tobias, and J Prinz. Handbook 151: Nist/sematech e-handbook of statistical methods, 2002.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021. URL https://arxiv.org/abs/2009.03300.

Jinghan Jia, Jiancheng Liu, Yihua Zhang, Parikshit Ram, Nathalie Baracaldo, and Sijia Liu. Wagle: Strategic weight attribution for effective and modular unlearning in large language models, 2025. URL https://arxiv.org/abs/2410.17509.

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.

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

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.

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.

Bo Liu, Qiang Liu, and Peter Stone. Continual learning and private unlearning, 2022. URL https://arxiv.org/abs/2203.12817.

Chris Yuhao Liu, Yaxuan Wang, Jeffrey Flanigan, and Yang Liu. Large language model unlearning via embedding-corrupted prompts, 2024a. URL https://arxiv.org/abs/2406.07933.

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, 2024b. URL https://arxiv.org/abs/2402.08787.

Ximing Lu, Sean Welleck, Jack Hessel, Liwei Jiang, Lianhui Qin, Peter West, Prithviraj Ammanabrolu, and Yejin Choi. Quark: Controllable text generation with reinforced unlearning, 2022. URL https://arxiv.org/abs/2205.13636.

Aman Madaan, Niket Tandon, Peter Clark, and Yiming Yang. Memory-assisted prompt editing to improve gpt-3 after deployment, 2023. URL https://arxiv.org/abs/2201.06009.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary Chase Lipton, and J Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=B41hNBoWLo.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt, 2023. URL https://arxiv.org/abs/2202.05262.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016.

Siyuan Mu and Sen Lin. A comprehensive survey of mixture-of-experts: Algorithms, theory, and applications, 2026. URL https://arxiv.org/abs/2503.07137.

Andrei I. Muresanu, Anvith Thudi, Michael R. Zhang, and Nicolas Papernot. Fast exact unlearning for in-context learning data for llms, 2025. URL https://arxiv.org/abs/2402.00751.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen

Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

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.

Martin Pawelczyk, Seth Neel, and Himabindu Lakkaraju. In-context unlearning: Language models as few shot unlearners, 2024. URL https://arxiv.org/abs/2310.07579.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2024. URL https://arxiv.org/abs/2305.18290.

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. 2024. URL https://arxiv.org/abs/2407.06460.

Qwen Team. Qwen1.5-moe: Matching 7b model performance with 1/3 activated parameters", February 2024. URL https://qwenlm.github.io/blog/qwen-moe/.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017. URL https://arxiv.org/abs/1706.03762.

Lingzhi Wang, Tong Chen, Wei Yuan, Xingshan Zeng, Kam-Fai Wong, and Hongzhi Yin. Kga: A general machine unlearning framework based on knowledge gap alignment, 2023. URL https://arxiv.org/abs/2305.06535.

Zihan Wang, Deli Chen, Damai Dai, Runxin Xu, Zhuoshu Li, and Y. Wu. Let the expert stick to his last: Expert-specialized fine-tuning for sparse architectural large language models, 2024. URL https://arxiv.org/abs/2407.01906.

Xinwei Wu, Junzhuo Li, Minghui Xu, Weilong Dong, Shuangzhi Wu, Chao Bian, and Deyi Xiong. Depn: Detecting and editing privacy neurons in pretrained language models, 2023. URL https://arxiv.org/abs/2310.20138.

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

Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning, 2024. URL https://arxiv.org/abs/2310.10683.

Charles Yu, Sullam Jeoung, Anish Kasi, Pengfei Yu, and Heng Ji. Unlearning bias in language models by partitioning gradients. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 6032–6048, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.375. URL https://aclanthology.org/2023.findings-acl.375/.

Jinghan Zhang, Shiqi Chen, Junteng Liu, and Junxian He. Composing parameter-efficient modules with arithmetic operations, 2023. URL https://arxiv.org/abs/2306.14870.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning, 2024. URL https://arxiv.org/abs/2404.05868.

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 (Volume 1: Long Papers)*, pp. 8664–8678, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.424. URL https://aclanthology.org/2025.acl-long.424/.

APPENDIX

A ALGORITHM OF TRACE

Algorithm 1 TRACE: Targeted Routing-Aware Calibration for Experts

Require: $\boldsymbol{\theta}^{(\mathrm{ptr})}$, $\mathcal{D}_f$, $\mathcal{D}_r$, $\widetilde{\mathcal{D}}_f$, λ_f, λ, $\tau > 0$
Ensure: Unlearned parameters $\boldsymbol{\theta}$
1: Initialize $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta}^{(\mathrm{ptr})}$.
2: Compute expert activation frequencies on the calibration set $\widetilde{\mathcal{D}}_f$.
3: Select high forget frequency experts using the IQR criterion to form $\mathcal{E}$.
4: **for** each unlearning step t **do**
5: Sample a forget batch $\mathcal{B}_f$ and a retain batch $\mathcal{B}_r$.
6: Compute activation frequency vectors $\mathbf{p}_m^r$ for each retain sample and $\bar{\mathbf{p}}^{\,f}$ over $\mathcal{B}_f$, both restricted to $\mathcal{E}$.
7: Compute distances d_m and normalized retain loss weights α_m.
8: Update only the parameters of experts in $\mathcal{E}$ using $\mathcal{L}_{\mathrm{TRACE}}$ defined in equation 9.
9: **end for**
10: **return** $\boldsymbol{\theta}$

B IMPLICIT PER-EXPERT SCALING IN GRADDIFF

We derive the gradient of an individual expert under the GradDiff objective. For $s \in \{f, r\}$, write the loss on the current batch as $\mathcal{L}_s(\boldsymbol{\theta}) = N_s^{-1} \sum_{i=1}^{N_s} \ell_i^s(\boldsymbol{\theta})$, where ℓ_i^s is the loss at a valid token position. We assume that the counted positions include all expert invocations that contribute to this loss and consider ordinary backpropagation through the selected top-K computation graph.

Contributions through expert computations. Let $\mathbf{h}_{i,\ell}^s$ be the input to the MoE module at position i in layer ℓ, and let $f_{j,\ell}(\cdot; \boldsymbol{\theta}_{j,\ell})$ denote the feedforward network of expert $e_{j,\ell}$ with parameters $\boldsymbol{\theta}_{j,\ell}$. The combined output of the routed experts is

$$\mathbf{u}_{i,\ell}^s = \sum_j \mathbf{1}[e_{j,\ell} \in \mathcal{S}_i^s]\, r_{i,j,\ell}^s\, f_{j,\ell}(\mathbf{h}_{i,\ell}^s; \boldsymbol{\theta}_{j,\ell}), \tag{10}$$

where $r_{i,j,\ell}^s$ is the routing weight. For expert parameters that are not shared across layers or with the router, $\mathbf{h}_{i,\ell}^s$ and the routing weights at this layer do not depend on $\boldsymbol{\theta}_{j,\ell}$. The local Jacobian through an active expert is therefore

$$\mathbf{J}_{i,j,\ell}^s = r_{i,j,\ell}^s \frac{\partial f_{j,\ell}(\mathbf{h}_{i,\ell}^s; \boldsymbol{\theta}_{j,\ell})}{\partial \boldsymbol{\theta}_{j,\ell}}. \tag{11}$$

The backward signal at this output is

$$\mathbf{a}_{i,\ell}^s = \nabla_{\mathbf{u}_{i,\ell}^s} \sum_{k=1}^{N_s} \ell_k^s(\boldsymbol{\theta}). \tag{12}$$

This signal includes all downstream paths through subsequent attention and layers. In particular, losses at later positions in the same sequence can contribute to $\mathbf{a}_{i,\ell}^s$. Define the contribution through this expert invocation as

$$\mathbf{c}_{i,j,\ell}^s = \left(\mathbf{J}_{i,j,\ell}^s\right)^\top \mathbf{a}_{i,\ell}^s, \tag{13}$$

and set it to zero when the expert is not selected. The chain rule then gives

$$\nabla_{\boldsymbol{\theta}_{j,\ell}} \mathcal{L}_s = \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbf{1}[e_{j,\ell} \in \mathcal{S}_i^s]\, \mathbf{c}_{i,j,\ell}^s. \tag{14}$$

Factorization by activation frequency. Using $p^s_{j,\ell}$ from the main text, the number of counted positions that activate this expert is $N_s p^s_{j,\ell}$. Define

$$\bar{\mathbf{c}}^s_{j,\ell} = \frac{\sum_{i=1}^{N_s} \mathbf{1}[e_{j,\ell} \in \mathcal{S}^s_i]\, \mathbf{c}^s_{i,j,\ell}}{N_s p^s_{j,\ell}}, \tag{15}$$

with $\bar{\mathbf{c}}^s_{j,\ell} = \mathbf{0}$ when $p^s_{j,\ell} = 0$. Thus, $\bar{\mathbf{c}}^s_{j,\ell}$ is the average contribution of active expert computations to the unnormalized batch loss, and $\nabla_{\boldsymbol{\theta}_{j,\ell}} \mathcal{L}_s = p^s_{j,\ell} \bar{\mathbf{c}}^s_{j,\ell}$. For GradDiff, this yields

$$\nabla_{\boldsymbol{\theta}_{j,\ell}} \mathcal{L} = -p^f_{j,\ell} \bar{\mathbf{c}}^f_{j,\ell} + \lambda p^r_{j,\ell} \bar{\mathbf{c}}^r_{j,\ell}. \tag{16}$$

The decomposition separates activation frequency from the average contribution through each expert invocation. Holding the average retain contribution fixed, a lower retain activation frequency reduces the retain gradient component. The actual balance between forget and retain gradients also depends on the magnitudes and directions of $\bar{\mathbf{c}}^f_{j,\ell}$ and $\bar{\mathbf{c}}^r_{j,\ell}$. This provides a perspective on our intuition that experts frequently activated by forget data may receive limited retain regularization when retain data rarely activates them, motivating routing-aware retain calibration.

Effect of routing-aware retain reweighting. The decomposition above also clarifies the role of the retain reweighting introduced in Section 4.2. Let

$$q_m = \frac{n_m}{N_r}$$

denote the fraction of retain tokens contributed by sample m, and let $p^r_{m,j,\ell}$ denote the activation frequency of expert $e_{j,\ell}$ within retain sample m. Under the TRACE-weighted retain objective, the effective activation frequency of expert $e_{j,\ell}$ becomes

$$\widetilde{p}^r_{j,\ell} = \sum_{m=1}^{B_r} q_m \alpha_m p^r_{m,j,\ell}.$$

Accordingly, the retain gradient with respect to this expert can be written as

$$\nabla_{\theta_{j,\ell}} \mathcal{L}^{(\alpha)}_r = \widetilde{p}^r_{j,\ell}\, \widetilde{\mathbf{c}}^r_{j,\ell},$$

where $\widetilde{\mathbf{c}}^r_{j,\ell}$ denotes the weighted average gradient contribution conditional on activating expert $e_{j,\ell}$.

Since α_m is larger for retain samples whose routing-frequency vectors are closer to that of the current forget batch, TRACE shifts the effective retain exposure toward samples that engage the selected experts in routing patterns more similar to the forget data. Thus, retain reweighting modifies not only the magnitude of the retain objective, but also how retain gradients are distributed across the selected experts.

Lemma B.1 (Routing discrepancy under TRACE reweighting)**.** *Let $q_m = n_m/N_r$ and let*

$$\alpha_m = \frac{\exp(-d_m/\tau)}{\sum_{m'=1}^{B_r} q_{m'} \exp(-d_{m'}/\tau)}.$$

Then

$$\sum_{m=1}^{B_r} q_m \alpha_m d_m \le \sum_{m=1}^{B_r} q_m d_m.$$

The inequality is strict unless all samples with positive q_m have the same routing distance.

This lemma shows that TRACE reweighting shifts the retain loss toward samples whose routing patterns are, on average, more similar to those of the forget data. Therefore, the reweighted retain objective provides a routing-matched form of regularization for the selected experts, rather than treating all retain samples uniformly. Importantly, the result concerns the average routing discrepancy and does not imply that the activation frequency of every selected expert increases after reweighting.

Proof. Since $d \mapsto \exp(-d/\tau)$ is decreasing,

$$\mathrm{Cov}_q\left(d_m, \exp(-d_m/\tau)\right) \le 0.$$

Therefore,

$$\sum_m q_m d_m \exp(-d_m/\tau) \le \left(\sum_m q_m d_m\right)\left(\sum_m q_m \exp(-d_m/\tau)\right).$$

Dividing both sides by $\sum_m q_m \exp(-d_m/\tau)$ gives the result. □

C Experiment Setup and Extended Experiments

C.1 Experimental Details.

Optimization and hyperparameter settings. We use AdamW for all experiments, with a batch size of 16 for WMDP and 4 for MUSE-BOOKS. For TRACE, retain sample weights are normalized within groups of 4 retain samples, with a token-weighted mean of one within each group. For GA, GradDiff, NPO, ESFT, and TRACE, we explore learning rates within $[5 \times 10^{-5}, 10^{-4}]$ and retain regularization coefficients λ within $[1, 2.5]$, where applicable. For WMDP, we consider training budgets of $[100, 300]$ optimization steps. For ESFT, we evaluate cumulative affinity thresholds $p \in \{0.1, 0.2\}$ and use $p = 0.1$ in the reported results. For TRACE, we fix the IQR coefficient $\lambda_f = 1.5$ across all main experiments and search the temperature over $\tau \in \{0.001, 0.002, 0.01, 0.05, 0.1\}$. The NPO parameter β is fixed at 0.01 for WMDP and 0.1 for MUSE-BOOKS. The configurations used for the main results are detailed below.

WMDP. The selected configurations for GA, GradDiff, and NPO use a learning rate of 5×10^{-5} and 100 optimization steps on both Qwen1.5-MoE-A2.7B-Chat and DeepSeek-V2-Lite-Chat. For SEUF, we use the authors' implementation and its accompanying configuration, with a learning rate of 2×10^{-4} and 100 optimization steps. ESFT and TRACE use a learning rate of 5×10^{-5}, with 250 steps on Qwen1.5-MoE-A2.7B-Chat and 300 steps on DeepSeek-V2-Lite-Chat. For TRACE, the selected temperature is $\tau = 0.002$ on Qwen1.5-MoE-A2.7B-Chat and $\tau = 0.01$ on DeepSeek-V2-Lite-Chat. For the reported results, we set the retain regularization coefficient to $\lambda = 2.5$.

MUSE-BOOKS. We first fine-tune Qwen3-30B-A3B on both the forget and retain sets using LoRA with rank $r = 16$, a learning rate of 5×10^{-5}, and two epochs. The resulting checkpoint serves as the common starting point for all unlearning methods. For the reported unlearning results, all methods use a learning rate of 5×10^{-5} and train for one epoch. GA, GradDiff, and NPO apply LoRA to all linear modules with rank $r = 16$. For TRACE, the selected temperature is $\tau = 0.05$. For the reported results, we set the retain regularization coefficient to $\lambda = 1$.

All experiments are conducted on machines equipped with either two NVIDIA A100 GPUs or two NVIDIA A800 GPUs.

Effect on retain activation frequencies. We define the weighted retain activation frequency as $\hat{p}^r_{j,\ell} = N_r^{-1} \sum_{m=1}^{B_r} n_m \alpha_m p^r_{m,j,\ell}$. Figure 6 shows that calibration increases this frequency for these high forget frequency experts, narrowing the gap with forget activation frequencies.

Sensitivity to the calibration set size. We vary the calibration set size by sampling $N \in \{5, 15, 25\}$ examples from each of the Bio and Cyber forget sets, resulting in $2N$ samples in total. As shown in Table 5, MMLU accuracy varies by only 0.67 percentage points across these settings, suggesting that utility preservation is relatively insensitive to calibration set size. WMDP accuracy does not improve monotonically with more calibration data: even five samples per domain achieve a WMDP average of 26.64% while retaining 57.19% MMLU accuracy.

D Selected Experts

Table 6 lists the selected expert sets $\mathcal{E}$ for Qwen1.5-MoE-A2.7B-Chat and DeepSeek-V2-Lite-Chat. The two sets contain 52 experts across 21 layers and 72 experts across 23 layers, respectively. Each

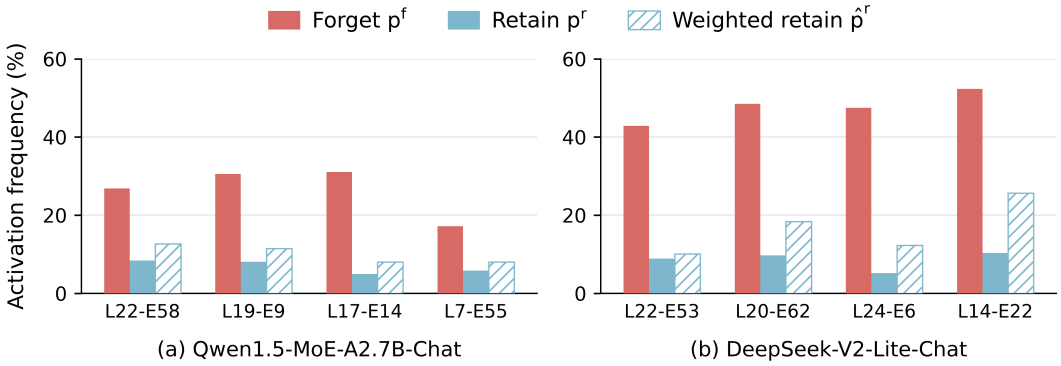


Figure 6: Forget, retain, and weighted retain activation frequencies for four high forget frequency experts from each model.

Table 5: Sensitivity to calibration set size on Qwen. N denotes the number of samples drawn from each forget domain. All scores are accuracy percentages.

Metric		$N = 5$	$N = 15$	$N = 25$
WMDP ↓	Bio	25.29	27.26	25.29
	Cyber	27.98	31.25	27.83
	Avg.	26.64	29.26	26.56
MMLU ↑		57.19	57.86	57.74

expert is identified by its layer ℓ and its index j within that layer. Layer and expert indices are reproduced from the selection files without renumbering. Only layers containing selected experts are listed.

E Retain Samples at Different Routing Distances

Tables 7 and 8 show the ten smallest and ten largest recorded retain sample distances from the first five training steps, with $\tau = 0.01$. Distances and weights refer to full samples. Bold text highlights biological and computer security terminology for illustration; the distances are computed from expert activation frequencies. References and weight normalization are specific to each local batch, including across records with the same step number. In these examples, all ten samples with smaller distances have $\alpha_m > 1$, while all ten samples with larger distances have $\alpha_m < 1$.

Table 6: Selected experts in Qwen1.5-MoE-A2.7B-Chat and DeepSeek-V2-Lite-Chat. n_ℓ denotes the number of selected experts in layer ℓ.

Qwen1.5-MoE-A2.7B-Chat			DeepSeek-V2-Lite-Chat		
Layer ℓ	Expert indices j	n_ℓ	Layer ℓ	Expert indices j	n_ℓ
1	54	1	4	36, 48	2
2	29, 42, 49	3	5	2, 46	2
3	20	1	6	4, 30	2
4	2, 21, 22	3	7	31, 47, 57	3
5	4, 20	2	8	9, 11, 13, 56	4
6	10, 30, 44, 56	4	9	0, 12, 63	3
7	0, 55	2	10	10, 28	2
8	20, 54, 56	3	11	42, 48, 62	3
9	1, 14	2	12	18, 52	2
10	0, 15	2	13	19, 26, 37	3
11	21, 53, 59	3	14	22	1
14	2	1	15	18, 26, 35, 38	4
15	10, 21	2	16	10, 20, 27, 40	4
16	44	1	17	2, 49, 50	3
17	14, 56	2	18	0, 4, 14, 17, 63	5
18	15	1	19	12, 32, 33	3
19	9, 41, 54	3	20	10, 32, 62, 63	4
20	12, 47	2	21	41, 44, 49	3
21	21, 33, 36	3	22	50, 53, 62	3
22	11, 23, 27, 51, 56, 58	6	23	12, 25, 30, 50	4
23	5, 16, 21, 23, 36	5	24	6, 16, 25, 27	4
			25	6, 14, 23, 27, 31, 51	6
			26	23, 50	2
Total	21 layers	52	Total	23 layers	72

Table 7: Retain samples with the ten smallest recorded distances.

Retain sample excerpt	Step	d_m ($\times 10^{-2}$)	α_m
L1. Like other stingrays, the **diamond stingray** is **aplacental viviparous**: the **embryos** are initially nourished by **yolk**, and later by **histotroph** ("**uterine milk**", rich in **proteins and lipids**) produced by the mother. …	5	1.485	3.501
L2. A linear chain of **amino acid residues** is called a **polypeptide**. A **protein** contains at least one long **polypeptide**. …	2	1.558	8.302
L3. Although usually trivalent, europium readily forms divalent compounds. This behavior is unusual to most lanthanides, which almost exclusively form compounds with an oxidation state of + 3. …	5	2.004	2.083
L4. Nuclear pores, which provide aqueous channels through the envelope, are composed of multiple **proteins**, collectively referred to as **nucleoporins**. …	5	2.298	1.553
L5. The noisy miner does not use a stereotyped courtship display; displays can involve 'driving' …	3	2.660	4.021
L6. The **prognosis** is worse in those with larger **burns**, those who are older, and those who are females. …	4	2.672	4.462
L7. The first **ceratopsian remains** known to science were discovered during the U.S. Geological and Geographical Survey of the Territories …	1	2.826	2.534
L8. In mid-September, a weak tropical wave moved off the coast of Northwest Africa. It moved westward and entered the eastern Pacific Ocean early on September 23. …	4	2.895	2.281
L9. In Charleston, South Carolina, the hurricane washed aground several ships and uprooted numerous trees, though damage to the city harbor was minimal. …	4	2.897	2.275
L10. The gods were believed to manifest in many forms. The Egyptians had complex conception of the human soul, consisting of several parts. …	1	3.320	1.838

Table 8: Retain samples with the ten largest recorded distances.

Retain sample excerpt	Step	d_m ($\times 10^{-2}$)	α_m
H1. For the fourth game, Takumi wrote the scenario and took on a supervisory role. He had wanted the series to end with the third game, as he felt Phoenix had been fully explored and …	2	6.684	0.049
H2. In 1905, after publishing four other books in the interim, Hornung brought back the character Stingaree, previously seen in Irralie's Bushranger. …	3	6.623	0.195
H3. In the January 2006 transfer window, he joined Grimsby Town on a two-year deal, three years after they first expressed an interest in signing him. …	2	6.226	0.078
H4. Wilde hesitated about submitting the script to Alexander, worrying that it might be unsuitable for the St James's Theatre, whose typical repertoire was relatively serious, and explaining that it had been written in response …	1	6.110	0.095
H5. The problem as I see it with the BDS movement is not the tactic. Who could not support Boycott, Divestment and Sanctions? …	3	6.052	0.345
H6. 150 people protested at the Church of Scientology building in Sydney, Australia, carrying signs and wearing costumes. … After the protest in Sydney, a surge in online Internet traffic due to individuals attempting to view pictures from the protest crashed hundreds of websites when a server was overloaded. …	3	6.026	0.354
H7. In March 2015, Vistara shifted to its new office at the One Horizon Center tower in Sector 43, Gurgaon, a satellite city of Delhi. …	2	5.940	0.104
H8. In the morning of December 31, the 4th Marines Battalion returned to the crash site with the entire force and the American graves were located and their corpses were dug up. …	2	5.898	0.108
H9. The ship was ordered on 22 March 1929 from Palmer's at Jarrow, under the 1928 Naval Programme. …	3	5.861	0.418
H10. Like Powderfinger's previous album Internationalist, Odyssey Number Five commented on social and political issues heavily, with the primary point of focus being Aboriginal affairs. …	2	5.770	0.123