

Forget-to-Focus: Targeted Unlearning for Domain Specialization in LLMs

Chaitali Bhattacharyya¹, Shinyoung Jang², Seohyun Kim¹, Junyoung Lee¹, Yeseong Kim¹

¹Department of Electrical Engineering, POSTECH, Pohang, Republic of Korea

²AX, POSTECH, Pohang, Republic of Korea

Abstract

Supervised fine-tuning is the standard approach for adapting pretrained large language models to specialized domains, yet it treats the pretrained checkpoint as a fixed and neutral initialization. In practice, pretrained models encode broad general-domain priors that can conflict with domain-specific learning. We propose FORGET-TO-FOCUS (F2F), a two-stage domain-specialization framework that first applies controlled unlearning to weaken interfering generalist behavior and then performs ordinary supervised fine-tuning from the resulting initialization. A key challenge is deciding what to forget. We introduce NEURON-GUIDED DATASET SELECTION (NG-DS), which constructs a model-elicited generalist proxy, identifies proxy-dominant feed-forward neurons using activation statistics, and selects forget examples that strongly activate generalist-dominant neurons while weakly activating domain-dominant neurons. We evaluate F2F on five LLM backbones spanning 0.6B–72B parameters across coding, medical question answering, and mathematical reasoning. Across coding, medical QA, and mathematical reasoning, F2F consistently improves over standard tuning and adaptation baselines. On HumanEval, it improves over SFT by +13.84 pp (percentage point) for Qwen3-0.6B, +7.86 pp for LLaMA3.1-8B, and +10.14 pp for LLaMA2-13B. On PubMedQA, it improves Qwen3-0.6B from 62.60 pp to 80.50 pp, showing that targeted preparatory forgetting provides a better initialization for domain specialization.

1 Introduction

Large language models (LLMs) are routinely adapted to specialized domains such as medical question answering, code generation, and mathematical reasoning by fine-tuning a general-purpose pretrained checkpoint on target-domain data (Parthasarathy et al., 2024; Hu et al., 2022;

Gururangan et al., 2020). This approach is simple and effective, but it treats the pretrained model as if it were a neutral starting point. In practice, a pretrained checkpoint carries broad general-domain priors, including correlations, styles, and heuristics acquired during pretraining. When these priors *interfere* with the target domain, fine-tuning must learn the target behavior *while also* pushing against behavior already encoded in the model. These two pressures are handled by the same gradient updates, which can leave specialization incomplete.

The common approach is to either invest more compute into fine-tuning (e.g., longer schedules and larger updates at the cost of adaptation pressure) or to constrain the update geometry using specialized fine-tuning methods. For example, LoRA constrains the update subspace (Hu et al., 2022), DAPT warms the model on domain text (Gururangan et al., 2020), and CurlLoRA improves stability under continued adaptation (Fawi, 2024). These methods are useful, but they all inherit the same pretrained state and rely on subsequent adaptation to absorb the conflict. Prior studies of pretraining and fine-tuning interactions (Sun and Dredze, 2025; Jiang et al., 2025; Zhang et al., 2022) suggest that pretrained knowledge is not uniformly helpful for downstream learning.

In this work, we exploit machine unlearning (Li et al., 2025) to minimize the interference. Unlike typical use cases of unlearning, i.e., removing specified information after deployment, often for privacy, safety, or compliance (Li et al., 2025; Geng et al., 2025; Yao et al., 2024; Zhang et al., 2024), we treat forgetting as *preparation* rather than a deletion mechanism. If part of a pretrained model competes with target-domain learning, then selectively weakening that part before fine-tuning can produce a more favorable initialization. It makes the target model reduce pretraining-induced interference enough to make later domain learning easier.

We propose **F2F (FORGET TO FOCUS)**, a two-

stage training method for domain specialization. In the first stage, controlled unlearning is applied to an identified forget set, optionally with a retain set or KL anchor to limit destructive drift. This produces a modified initialization $\hat{\theta}_0$ that is less aligned with pretraining-dominant behavior. In the second stage, ordinary supervised fine-tuning is performed on the target-domain data starting from $\hat{\theta}_0$. The final training objective is therefore unchanged from standard fine-tuning. The difference is the state from which fine-tuning begins.

The main challenge is how to decide what should be forgotten. Forgetting arbitrary general-domain text can be ineffective when the selected examples do not activate the interfering behavior. It can also be destructive when the selected examples overlap with domain-relevant knowledge. We therefore introduce **NG-DS (Neuron-Guided Dataset Selection)**, a model-aware selector for constructing forget sets. NG-DS compares feed-forward neuron activation frequencies on a pretraining proxy and a target-domain proxy, identifies pretraining-dominant and domain-dominant neurons, and ranks candidate examples by their relative activation of these two neuron groups. A high-scoring example activates neurons associated with pretraining-like behavior while weakly activating neurons associated with the target domain. This makes the forget set less dependent on surface similarity and better aligned with the model’s internal separation between generalist and domain-relevant behavior.

We evaluate F2F across five pretrained LLM backbones from 0.6B to 72B and three domains: coding, mathematics, and medical question answering. F2F+SFT consistently outperforms standard SFT and other adaptation baselines. On HumanEval, it improves Qwen3-0.6B from 31.71 to 45.55 and Qwen-2-72B-Instruct from 71.12 to 82.70 pass@1.

We make following contributions.

1. We propose F2F, a two-stage framework that treats unlearning as a pre-adaptation operation for domain specialization.
2. We introduce NG-DS, a neuron-guided forget-set selector that chooses examples activating pretraining-dominant neurons while avoiding domain-dominant neurons.
3. We show that F2F improves over SFT and adaptation baselines across diverse model families for three domains.
4. We provide representational analyses, showing that F2F reaches an internal geometry distinct from

ordinary tuning while making a better initialization for domain-specific adaptation.

2 Related Work

Domain adaptation of pretrained LLMs. Supervised fine-tuning (Parthasarathy et al., 2024; Devlin et al., 2019) adapts pretrained models by updating parameters on target-domain data. Domain-Adaptive Pretraining (DAPT) (Gururangan et al., 2020) first continues language-model training on in-domain text, while parameter-efficient methods such as LoRA (Hu et al., 2022) and Curl-LoRA (Fawi, 2024) restrict the adaptation update to improve efficiency or stability. Although these methods differ in objective and parameterization, they all start from the same pretrained checkpoint and rely on the adaptation stage to resolve conflicts between pretrained behavior and the target task. F2F is complementary: it modifies the initialization before downstream adaptation, after which standard SFT or parameter-efficient methods can be applied unchanged.

Machine unlearning in LLMs. Machine unlearning aims to remove the influence of selected data, facts, or behaviors from trained models, often for privacy, copyright, safety, or regulatory reasons (Hoofnagle et al., 2019; Li et al., 2025; Geng et al., 2025). Recent LLM unlearning methods include gradient ascent on the forget set (Yao et al., 2024), retain-set descent, KL anchoring to the original model, and Negative Preference Optimization (NPO) (Zhang et al., 2024). F2F uses this family of operations for a different purpose: rather than treating unlearning as deletion, we use it as a pre-adaptation signal that weakens behavior likely to interfere with target-domain specialization.

Plasticity, stability, and continual adaptation. Continual adaptation must balance plasticity to new data with retention of pretrained capabilities (Jiang et al., 2025). Prior work addresses this tradeoff through regularization, replay, parameter isolation, or selective forgetting. F2F instead reshapes the pretrained initialization through targeted unlearning before standard downstream adaptation.

Interaction between pretraining and fine-tuning. Negative transfer occurs when knowledge transferred from a source domain harms target-domain learning (Zhang et al., 2022). Recent studies of the pretrain-then-fine-tune pipeline show that pretrained knowledge is not uniformly beneficial:

some tasks benefit less from additional supervision when already expressed during pretraining (Sun and Dredze, 2025), while selective forgetting during fine-tuning can improve plasticity (Jiang et al., 2025). F2F builds on this view but applies a separate controlled unlearning stage before fine-tuning, allowing it to operate on released checkpoints without access to the original pretraining process. Unlike corpus-label or metadata-based approaches, it constructs forget sets from the model’s internal activation patterns.

Activation-based neuron specialization. Prior work has shown that Transformer neurons and feed-forward components encode interpretable patterns, factual knowledge, and task-specific skills (Geva et al., 2021; Dai et al., 2022; Wang et al., 2022). In contrast, NG-DS uses activation differences operationally to select examples for targeted unlearning, turning neuron specialization into a signal for pre-adaptation forget-set construction.

3 Methodology

We propose FORGET-TO-FOCUS (F2F), a two-stage framework for domain specialization. Direct fine-tuning adapts from the original pretrained initialization θ_0 , so it must learn the target behavior while simultaneously overriding broad general-domain behavior that conflicts with the target domain. F2F separates these two pressures. It first applies controlled unlearning to obtain a modified initialization $\tilde{\theta}_0$ in which selected pretraining-dominant behavior is weakened, and then performs ordinary supervised fine-tuning from $\tilde{\theta}_0$ for the target-domain data $\mathcal{D}$. The downstream objective is therefore unchanged from standard fine-tuning; the difference is the initialization from which the objective is optimized. The full pipeline is

$$\theta_0 \xrightarrow{\text{NG-DS}} \mathcal{F} \xrightarrow{\text{unlearning}} \tilde{\theta}_0 \xrightarrow{\text{domain fine-tuning}} \theta_{\text{F2F}}, \quad (1)$$

where $\mathcal{F}$ is the selected forget set and θ_{F2F} is the final domain-specialized model.

Figure 1 illustrates the intuition behind the F2F pipeline. F2F does not treat general-domain knowledge as uniformly harmful, but instead aims to weaken the subset of generalist behavior that obstructs target-domain adaptation before fine-tuning begins. This view imposes two requirements on the unlearning stage. First, forgetting must be controlled so that it reduces pretraining-induced interference without broadly degrading target-domain

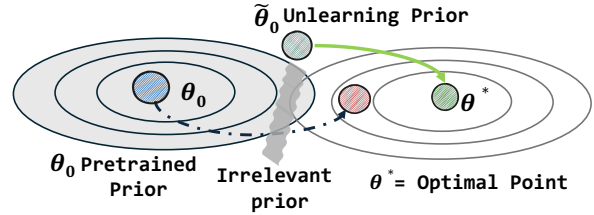


Figure 1: Schematic illustration of how pretraining priors create optimization barriers that slow convergence and induce suboptimal local minima when fine-tuning from θ_0 . Unlearning weakens these priors and can provide a cleaner starting point for fine-tuning.

competence. Second, the forget set must be model-aware since arbitrary general-domain examples may fail to activate the behavior that interferes with specialization, whereas examples that overlap with the target domain may erase useful knowledge.

To meet these requirements, we introduce NG-DS, which curates forget examples using internal activation statistics rather than surface similarity alone; we provide supporting diagnostics in Appendix D.1, showing that naive forget sets can exhibit non-trivial target-domain overlap and weak or negative alignment with target-domain descent directions.

3.1 Neuron-Guided Dataset Selection

NG-DS constructs $\mathcal{F}$ by selecting examples that strongly activate neurons associated with broad generalist behavior while weakly activating neurons associated with the target domain. It proceeds in three steps, (i) construct a model-elicited generalist proxy dataset $\mathcal{P}$ and a target-domain proxy $\mathcal{D}$, (ii) identify layers and neurons whose activation frequencies distinguish the two proxies, and (iii) rank candidate examples by their relative activation of the resulting neuron groups. This makes forget-set construction model-aware. Examples are selected because of how the current checkpoint internally represents them, not simply because their text appears far from the target domain.

Constructing a generalist proxy dataset. A key difficulty is how to get the pretraining knowledge for a given model. The original pretraining corpus of a released checkpoint is typically unavailable. More importantly, even if such a corpus were available, it would not be the most direct target for preparatory forgetting since our goal is not to reconstruct the historical data, but to identify *what the model internally knows*.

We therefore construct $\mathcal{P}$ as a model-elicited proxy for generalist behavior based on a MAGPIE-style procedure (Xu et al., 2024). MAGPIE observes that an instruction-tuned autoregressive model can synthesize a user instruction when conditioned only on the conversation template up to the user role. While MAGPIE uses this observation to synthesize alignment data, we repurpose it as a mechanism for eliciting the distributional structure expressed by the checkpoint itself. In this view, samples elicited in this way approximate broad, high-probability behaviors that the checkpoint itself can readily generate.

Formally, let t_{user} denote the pre-query template ending at the user role, and let t_{asst} denote the assistant-role template. We sample instructions and responses from the pretrained checkpoint θ_0 as

$$u_i \sim p_{\theta_0}(\cdot \mid t_{\text{user}}), \quad r_i \sim p_{\theta_0}(\cdot \mid t_{\text{user}}, u_i, t_{\text{asst}}), \quad (2)$$

and serialize the resulting instruction–response pairs into

$$\mathcal{P} = \{\text{serialize}(u_i, r_i)\}_{i=1}^{n_p}. \quad (3)$$

Because $\mathcal{P}$ consists of examples elicited from the checkpoint itself, it provides a suitable candidate pool for $\mathcal{F}$, i.e., selected examples target behaviors that remain active in the released model and may interfere with target-domain specialization.

Estimating proxy-dominant neurons NG-DS compares the pretrained checkpoint’s internal activations on $\mathcal{P}$ and $\mathcal{D}$ to identify neurons that preferentially respond to one proxy over the other. For neuron j in layer ℓ , let $a_{\ell j}(x)$ denote the MLP activation of the pretrained model on input x . A neuron is considered active when its activation exceeds a fixed threshold τ :

$$z_{\ell j}(x) = 1 [a_{\ell j}(x) > \tau]. \quad (4)$$

Let $\Pi_p = \mathcal{P}$ and $\Pi_d = \mathcal{D}$ with $s \in \{p, d\}$. We estimate proxy-specific activation frequencies as

$$f_s(\ell, j) = \mathbb{E}_{x \sim \Pi_s} [z_{\ell j}(x)]. \quad (5)$$

Using a dominance ratio $\alpha > 1$, we assign neurons to two proxy-dominant groups:

$$N_p^\ell = \{j : f_p(\ell, j) > \alpha f_d(\ell, j)\}, \quad (6)$$

$$N_d^\ell = \{j : f_d(\ell, j) > \alpha f_p(\ell, j)\}. \quad (7)$$

Here, N_p^ℓ contains neurons preferentially activated by the model-elicited generalist proxy, while N_d^ℓ contains neurons preferentially activated by the target-domain proxy. Neurons satisfying neither condition are treated as ambiguous and excluded from scoring. Because layers differ in how sharply they separate the two proxies, we compute a discriminative count score for each layer and retain the K layers with the largest absolute scores:

$$S_\ell = |N_p^\ell| - |N_d^\ell|, \quad \mathcal{L} = \text{TopK}_\ell(|S_\ell|). \quad (8)$$

Selecting the forget set Finally, NG-DS selects the forget set $\mathcal{F}$ from $\mathcal{P}$. For a candidate example $x \in \mathcal{P}$, we compute the fraction of active proxy-dominant neurons in each selected layer:

$$A_s^\ell(x) = \frac{1}{\max(|N_s^\ell|, 1)} \sum_{j \in N_s^\ell} z_{\ell j}(x). \quad (9)$$

We average these quantities over the selected layers and form a generalist-to-domain activation ratio:

$$\bar{A}_s(x) = \frac{1}{|\mathcal{L}|} \sum_{\ell \in \mathcal{L}} A_s^\ell(x), \quad (10)$$

$$q(x) = \frac{\bar{A}_p(x)}{\bar{A}_d(x) + \epsilon}. \quad (11)$$

A high value of $q(x)$ indicates that x strongly activates generalist-dominant neurons while weakly activating domain-dominant neurons. Such examples are suitable forget candidates because unlearning on them is more likely to suppress checkpoint-specific generalist behavior than target-domain competence. The final forget set is the set of top-scoring candidates:

$$\mathcal{F} = \text{TopK}_{x \in \mathcal{P}}(q(x)) \quad (12)$$

We provide a discussion for more detailed theoretical properties of NG-DS in Appendix A.

3.2 Forget-to-Focus

Given $\mathcal{F}$, F2F consists of controlled unlearning followed by standard domain fine-tuning.

Stage 1: controlled unlearning. The unlearning stage raises the language-modeling loss on $\mathcal{F}$ while using retain and KL terms to constrain destructive drift. The retain set $\mathcal{R}$ is drawn from the target domain as a small subset and serves as a stability anchor during unlearning; it is not intended to replace the later supervised fine-tuning stage. Let

$$\mathcal{L}_{\mathcal{F}}(\theta) = \mathbb{E}_{x \sim \mathcal{F}} [-\log p_\theta(x)] \quad (13)$$

be the forget-set language-modeling loss, and let

$$\mathcal{L}_{\mathcal{R}}(\theta) = \mathbb{E}_{(x,y) \sim \mathcal{R}} [-\log p_{\theta}(y | x)] \quad (14)$$

be the retain loss. We optionally anchor the model to its pretrained behavior using

$$\mathcal{L}_{\text{KL}}(\theta) = \mathbb{E}_{x \sim \mathcal{R}} [\text{KL}(p_{\theta_0}(\cdot | x) \| p_{\theta}(\cdot | x))]. \quad (15)$$

The unlearning objective is

$$\begin{aligned} \tilde{\theta}_0 = \arg \min_{\theta} \bigg[& -\lambda_f \mathcal{L}_{\mathcal{F}}(\theta) + \lambda_r \mathcal{L}_{\mathcal{R}}(\theta) \\ & + \lambda_{\text{KL}} \mathcal{L}_{\text{KL}}(\theta) \bigg]. \end{aligned} \quad (16)$$

The first term suppresses reliance on the selected forget set, while the retain and KL terms constrain destructive drift. Unless otherwise stated, our main experiments use the retain-aware $GA+GD$ variant; gradient-ascent-only unlearning (GA) is reported as an ablation.

Stage 2: domain fine-tuning. After T_u unlearning steps, we set $\tilde{\theta}_0 = \theta_{T_u}$ and fine-tune on the target domain:

$$\begin{aligned} \theta_{\text{F2F}} = \arg \min_{\theta} \mathbb{E}_{(x,y) \sim \mathcal{D}} [-\log p_{\theta}(y | x)], \\ \theta \text{ initialized at } \tilde{\theta}_0. \end{aligned} \quad (17)$$

Thus, F2F uses the same final objective as standard fine-tuning; its effect comes from preparing a checkpoint that is less dominated by pretraining-induced behavior that conflicts with the target domain. Appendix B provides the formal analysis behind F2F.

4 Evaluation

4.1 Experimental Setup

Models & Datasets We performed our experiments on models of different sizes, architectures, variants, and family: Qwen-2 72B-Instruct (Peng et al., 2023), LLaMA-2 13B (Touvron et al., 2023), LLaMA 3.1 8B-Instruct (Grattafiori et al., 2024), Gemma-2B-Instruct (Team et al., 2024) and Qwen-3-0.6B (Yang et al., 2025).

The evaluation covers three specialization domains, coding, medical question answering, and mathematical reasoning. For the coding domain, we trained on train set of OpenCoder (Huang et al., 2024) and evaluated on test set of HumanEval (Chen et al., 2021) and MBPP (Austin et al., 2021). In the medical domain, we utilized PubMedQA (Jin et al., 2019), PubMed

Guidelines (Chen et al., 2023a,b), and MedMCQA (Pal et al., 2022)’s training split as training datasets, and evaluated performance on the PubMedQA and MedMCQA test sets. For the mathematics domain, we used the NVIDIA’s OpenMathInstruct-1 dataset (Toshniwal et al., 2024) for training, while evaluation was carried out on the Hendrycks MATH (Hendrycks et al., 2021) and GSM8K (Cobbe et al., 2021) benchmarks. For the evaluation, we used LM-EVALUATION-HARNESS repository by Eleuther (Gao et al., 2024).

Hyperparameter Configuration Unless otherwise stated, NG-DS uses a fixed configuration across experiments: $\tau = 1.5$, $\alpha = 1.5$, $K = 6$, $\epsilon = 10^{-6}$, and $k = 1000$ for larger models or $k = 100$ for Qwen3-0.6B. For all experiments, we use $|\mathcal{P}| = 10,000$ candidates and 500 target-domain reference samples as the domain proxy. Additional forget-set and hyperparameter ablations are provided in Appendix D.

For unlearning, the base learning rate was 1×10^{-5} with the gradient ascent (GA) weight 1.0 and the gradient descent (GD) weight 0.5. For LLaMA models, we use a learning rate of 3×10^{-5} during unlearning, which stabilized convergence in our experiments. Qwen3-0.6B uses batch size 8 for 5 unlearning epochs; the remaining models use batch size 2. For Qwen-2 72B-Instruct, we use QLoRA with rank 16, dropout 0.05, and bfloat16 precision during unlearning. We use 100 forget samples for Qwen3-0.6B, 1000 forget samples for the remaining models, and a 1000-sample retain set.

For the fine-tuning stage, we use AdamW with learning rate 2×10^{-5} and effective batch size 128 via gradient accumulation over 32 steps. Qwen3-0.6B is fine-tuned for 8 epochs, while the remaining models are fine-tuned for 3 epochs to reduce overfitting on the relatively small domain datasets. For LLaMA-3.1-8B-Instruct, LLaMA-2-13B, and Qwen-2 72B-Instruct, we perform LoRA-based supervised fine-tuning using FP16 precision. For Qwen-2 72B-Instruct, we use 4-bit quantization for memory feasibility. All training runs are conducted on 80GB A100 GPUs.

4.2 Effect of F2F on Coding Performance

Table 1 shows that F2F improves coding performance across model families and scales. The retain-aware configuration, $Unl_{GA+GD} + SFT$, obtains the best result for every model-metric pair. For ex-

	Coding	Qwen 0.6B		Gemma 2B		LLaMA 8B-Instruct		LLaMA 13B		Qwen 72B	
		MBPP	HumanEval	MBPP	HumanEval	MBPP	HumanEval	MBPP	HumanEval	MBPP	HumanEval
F2F	(1) Base Model	22.60	19.50	19.80	16.46	49.00	33.54	27.22	33.60	67.21	70.12
	(1) + <i>SFT</i>	28.80	31.71	21.22	16.79	56.60	56.71	37.01	40.21	69.50	71.12
	(1) + <i>DAPT</i>	29.30	39.80	20.00	17.01	53.55	56.20	39.50	42.70	71.90	72.50
	(1) + <i>LoRA</i>	28.55	37.50	22.33	18.33	51.08	45.31	36.55	20.15	66.50	70.30
	(1) + <i>CurlLoRA</i>	31.00	40.91	17.50	17.09	57.40	52.93	40.50	42.00	69.00	68.20
	(1) + <i>ADePT</i>	30.81	39.93	22.90	20.38	53.35	47.62	38.72	31.17	68.78	72.45
	(1) + <i>DAPT+SFT</i>	31.89	41.11	22.03	22.63	61.99	61.81	52.02	47.67	74.55	79.78
	(2) <i>Unl_{GA}+GD</i>	32.77	23.04	12.00	4.10	47.80	59.08	31.42	34.80	71.31	75.40
	(2) + <i>SFT</i>	33.50	45.55	24.25	25.50	64.30	64.57	54.51	50.35	76.70	82.70
	(3) <i>Unl_{GA}</i>	24.10	20.50	5.00	4.20	26.80	5.40	4.20	29.70	64.25	69.22
	(3) + <i>SFT</i>	32.55	43.33	23.60	22.22	62.86	61.90	49.21	48.90	74.65	80.20

Table 1: MBPP and HumanEval pass@1 across different models (Qwen-2 72B-Instruct (Peng et al., 2023), LLaMA-2 13B (Touvron et al., 2023), LLaMA-3.1 8B-Instruct (Grattafiori et al., 2024), Gemma-2B-Instruct (Team et al., 2024), and Qwen-3-0.6B (Yang et al., 2025)) and tuning methods (higher is better). **Best**; **Second best**.

	Method	Coding		Medical		Mathematics	
		MBPP	HumanEval	PubMedQA	MedMCQA	Hendrycks	GSM8K
Qwen-0.6B	(1) <i>Unl_{GA}+GD</i>	32.77	23.04	61.66	32.35	40.07	0.02
	(1) + <i>Tuning</i>	33.50	45.55	80.50	47.22	60.11	17.20
	(2) <i>Unl_{GA}</i>	24.10	20.50	59.20	33.00	38.01	0.08
	(2) + <i>SFT</i>	32.55	43.33	74.39	42.30	53.33	15.33
	(3) <i>Baseline</i>	22.60	19.50	57.80	32.25	41.09	0.02
LLaMA-8B	(3) + <i>Tuning</i>	28.80	31.71	62.60	42.12	50.06	12.30
	(1) <i>Unl_{GA}+GD</i>	47.80	59.08	84.25	60.33	30.50	65.78
	(1) + <i>Tuning</i>	64.30	64.57	90.22	74.20	38.00	69.50
	(2) <i>Unl_{GA}</i>	26.80	5.40	81.00	61.30	17.30	58.30
	(2) + <i>SFT</i>	62.86	61.90	88.73	72.33	35.50	65.30
LLaMA-13B	(3) <i>Baseline</i>	49.00	33.54	75.20	59.43	19.90	62.85
	(3) + <i>Tuning</i>	56.60	56.71	85.31	64.20	25.51	66.70
	(1) <i>Unl_{GA}+GD</i>	31.42	34.80	78.59	46.55	32.01	7.75
	(1) + <i>Tuning</i>	54.51	50.35	94.75	78.33	53.30	25.60
	(2) <i>Unl_{GA}</i>	4.20	29.70	71.20	34.30	23.57	1.50
	(2) + <i>SFT</i>	49.21	48.90	89.10	71.00	48.30	23.30
	(3) <i>Baseline</i>	27.22	33.60	75.20	38.68	27.5	2.00
	(3) + <i>Tuning</i>	37.01	40.21	86.30	54.00	37.09	16.30

Table 2: Effect of F2F across domains. We compare baseline, unlearning-only, and unlearn-then-tune variants using the NG-DS forget set on Qwen3-0.6B, LLaMA3.1-8B-Instruct, and LLaMA2-13B across coding, medical, and mathematical benchmarks (higher is better). **Best**; **Second best**

ample, Qwen-0.6B improves from 31.71 with SFT to 45.55 with F2F on HumanEval, LLaMA-3.1-8B improves from 56.71 to 64.57, and Qwen-72B improves from 72.50 under DAPT to 82.70 with F2F. Similar gains are observed on MBPP, where F2F achieves the best performance for all reported models, including Qwen-0.6B improves from 31.00 under CurlLoRA to 33.50 and LLaMA-3.1-8B from 57.40 to 64.30.

Notably, F2F also consistently outperforms the recent ADePT baseline across all model metric pairs. Moreover, even compared with the stronger two-stage DAPT+SFT baseline, F2F provides consistent gains, improving HumanEval by 4.44, 2.87, 2.76, 2.68, and 2.92 points for Qwen-0.6B, Gemma-2B, LLaMA-3.1-8B, LLaMA-2-13B, and Qwen-72B, respectively.

These results support our key premise that specialization benefits from modifying the initialization before supervised adaptation. SFT, DAPT, LoRA, and CurlLoRA all begin from the original pretrained model, so target-domain optimization must both learn the desired behavior and compensate for incompatible general-domain behavior. F2F, in contrast, weakens selected pretraining-dominant behavior first, and then fine-tunes from the modified checkpoint.

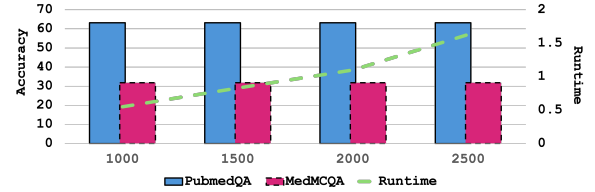


Figure 2: Runtime across different unlearning step counts, measured on a single A100 in GPU-hours.

The comparison between *Unl_{GA}+GD* and *Unl_{GA}* shows that our retain-aware approach is more reliable. GA-only unlearning can improve performance after SFT, but it is more unstable and sometimes causes large intermediate degradation although it is not a final deployed checkpoint. For instance, Gemma-2B drops to 5.00 on MBPP and 4.20 on HumanEval after GA-only unlearning.

After supervised tuning, performance recovers and surpasses the standard baselines, but they still remain consistently below the retain-aware *GA + GD* variant. Hence, retention prevents excessive degradation of useful capabilities, while unlearning suppresses behavior that interferes with specialization.

4.3 Effect of F2F on Other Domains

Table 2 evaluates whether the same pattern extends beyond coding. The results show that F2F consis-

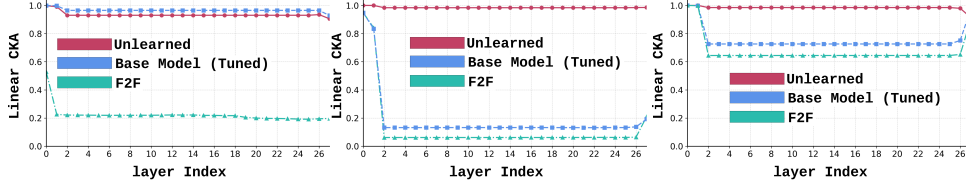


Figure 3: Representational drift measured by linear CKA. Across three domains, tuning substantially lowers similarity to the unlearned model; F2F exhibits the most pronounced departure. From left to right : Medical Domain, Mathematics Domain and Coding Domain.

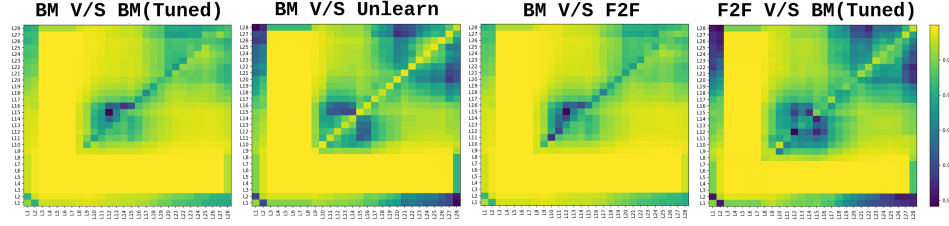


Figure 4: Heatmaps of representational similarity measured by SVCCA. We compare layer-wise representations across (i) Base model (BM) vs. fine-tuned base model, (ii) Base model (BM) vs. unlearned model, (iii) Base model (BM) vs. F2F, and (iv) Fine-tuned base model vs. F2F. SVCCA emphasizes alignment of shared high-variance factors.

tently improves downstream performance across all three domains, including coding, medical, and mathematical benchmarks.

For Qwen3-0.6B, $Unl_{GA+GD} + Tuning$ achieves the best results on all six benchmarks, improving over direct tuning from 28.80 to 33.50 on MBPP, 31.71 to 45.55 on HumanEval, 62.60 to 80.50 on PubMedQA, and 50.06 to 60.11 on Hendrycks-MATH. Similar trends are observed for larger models, e.g., on LLaMA3.1-8B, the same configuration improves HumanEval from 56.71 to 64.57 and PubMedQA from 85.31 to 90.22; on LLaMA2-13B, it improves MBPP from 37.01 to 54.51 and PubMedQA from 86.30 to 94.75. These improvements are most evident after the tuning stage. The unlearned checkpoint alone is not always the strongest model; but once reshaping the pretrained checkpoint with the unlearning, F2F provides a stronger initialization than the original pretrained model.

We observe a similar effect of retention-aware unlearning discussed in Section 4.2, as shown by the comparison between $Unl_{GA+GD} + Tuning$ and $Unl_{GA} + SFT$. For example, on Qwen3-0.6B, adding the GD term improves the final tuned result from 74.39 to 80.50 on PubMedQA and from 53.33 to 60.11 on Hendrycks-MATH. It suggests that GA helps remove harmful or conflicting priors, while GD preserves useful general capabilities, making F2F a more effective and balanced initial-

ization strategy.

4.4 Effect of Unlearning Step Sizes

Figure 2 evaluates the changes in the accuracy and runtime over different unlearning step counts for medical domain benchmarks. The results show that as the number of unlearning steps increases, accuracy on the medical domain benchmarks remains essentially unchanged, while runtime grows. Accordingly, we choose 1,000 unlearning steps as a practical default as it matches the best accuracy with reduced runtime demands.

Appendix D reports additional ablation studies, including forget-set quality, NG-DS hyperparameter sensitivity, unlearning and fine-tuning variants, dataset size, unlearning weights, multi-seed robustness, and extended coding results.

4.5 Computational cost.

Proxy generation and activation collection constitute the dominant preprocessing costs of F2F, whereas candidate selection itself adds negligible overhead. For a 1,000-example forget set, the complete two-stage procedure requires 1.02 wall-clock hours on four A100-80GB GPUs, compared with 0.14 hours for vanilla QLoRA SFT under the reported setting. The peak memory cost increases from 55.9 to 59.1 GiB/GPU. Importantly, although the generalist proxy is model-specific, it is not target-domain-specific and can therefore be reused

Configuration	Forget	C_{proxy}	C_{act}	C_{sel}	C_{unl}	C_{SFT}	C_{total}	Mem.
F2F (4-bit QLoRA)	1,000	0.41 h	0.41 h	0.01 h	0.12 h	0.07 h	1.02 h	59.1 GiB
F2F (4-bit QLoRA)	500	0.41 h	0.41 h	0.01 h	0.06 h	0.07 h	0.96 h	59.1 GiB
F2F (4-bit QLoRA)	100	0.41 h	0.41 h	0.01 h	0.01 h	0.07 h	0.91 h	59.1 GiB
Vanilla SFT	—	—	—	—	—	0.14 h	0.14 h	55.9 GiB

Table 3: Stage-wise wall-clock cost of F2F on Qwen-72B using four A100-80GB GPUs with a proxy pool of 10k samples. All F2F experiments use 4-bit QLoRA. The same model-elicited proxy can be reused across specialization domains for the same checkpoint, so proxy construction need not be repeated for each target domain.

when specializing the same checkpoint to multiple target domains. This amortizes the dominant proxy-generation cost in multi-domain settings.

4.6 Representation Geometry Analysis

The accuracy results show that F2F improves downstream performance, but they do not by themselves explain how the two-stage pipeline changes the model. We therefore examine whether F2F reaches a different internal state from ordinary tuning using Centered Kernel Alignment (CKA) (Kornblith et al., 2019) and Singular Vector Canonical Correlation Analysis (SVCCA) (Raghu et al., 2017). CKA compares representational geometry between corresponding layers, while SVCCA compares the alignment of shared high-variance subspaces. Formal definitions and implementation details are provided in Appendix C.

Figure 3 compares layer-wise representations across the three domains. In all cases, F2F induces stronger representational drift than ordinary tuning. We observe more significant influences in the medical and mathematical domains, where F2F sharply lowers similarity across most layers, and remains visible in the coding domain, where the reduction is more moderate but consistent. This suggests that F2F does not simply reproduce the trajectory of standard fine-tuning after an additional optimization step. Instead, the unlearning changes the initialization geometry from which subsequent domain tuning proceeds.

Figure 4 complements the CKA analysis by comparing high-variance representational subspaces across layers. The heatmaps show that the models remain broadly aligned, as expected because they share the same pretrained checkpoint and architecture. However, the relative reductions in similarity are concentrated in the middle and upper layers, especially when comparing F2F with the tuned base model. This pattern indicates that F2F pre-

serves much of the shared representational backbone while reorganizing task-relevant subspaces that are more affected by adaptation.

The comparison with the unlearned model further suggests that unlearning alone changes the representation, but the final F2F model reflects the combined effect of preparatory forgetting and subsequent target-domain tuning. The CKA and SVCCA analysis results suggest that targeted unlearning reshapes internal representations in a way that makes subsequent domain specialization more effective, rather than merely adding another training stage that prolonged standard finetuning hardly achieves.

Compute-matched Comparison To also empirically verify whether the gains of F2F can be explained simply by additional training, we compare against baselines trained for the same number of epochs as the full F2F pipeline.

In these experiments, we test the performance for Qwen3-0.6B and LLaMA3.1-8B-Instruct in the coding domain. For Qwen3-0.6B, the matched budget is 13 epochs, corresponding to unlearning followed by fine-tuning; for LLaMA3.1-8B-Instruct, it is 5 epochs. As shown in Figure 5, F2F remains the strongest method under this matched epoch budget. For example, F2F with Unl_{GA+GD} followed by SFT achieves the largest average improvement, with gains of +10.90 on MBPP and +26.05 on HumanEval, yielding an average gain of +18.48. These results further narrow the explanation of F2F’s effectiveness, i.e., its gains are not merely due to additional fine-tuning epochs, but comes from the unlearning stage shaping a more favorable initialization for subsequent fine-tuning.

We provide more empirical results for the representation changes by F2F in Appendix E, including empirical analyses of retention, Fisher sensitivity, LoRA rank efficiency, and calibration along with representative NG-DS forget-set samples.

4.7 Mechanistic Analysis of F2F

Targeted Neuron Intervention. To test whether NG-DS-selected neurons influence target-domain predictions, we suppress the generalist-dominant, domain-dominant, or a size-matched random neuron group on 400 PubMedQA examples. As shown in Table 4, suppressing generalist-dominant neurons significantly increases the gold-label margin by +0.154, compared with only +0.023 for random suppression. Suppressing domain-dominant

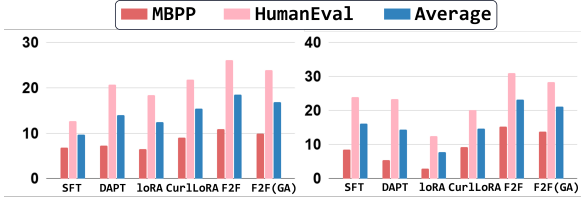


Figure 5: Compute-matched comparison between standard fine-tuning baselines and F2F on coding benchmarks. Improvements are reported relative to the base model. Left: Qwen 3 0.6B and Right : LLaMA 3.1 8B-Instruct

Suppressed neuron group	n	Δ Gold-label margin	BH q -value
NG-DS generalist-dominant	400	+0.154 [+0.100, +0.205]	7.62×10^{-9}
NG-DS domain-dominant	400	-0.087 [-0.147, -0.025]	8.48×10^{-2}
Matched random	400	+0.023 [-0.018, +0.060]	5.50×10^{-1}

Table 4: Targeted neuron suppression on PubMedQA using Qwen-3-0.6B. Intervals denote 95% confidence intervals; q -values use Benjamini–Hochberg correction.

neurons instead decreases the margin. These results provide local mechanistic evidence that NG-DS identifies functionally relevant, non-random neuron groups.

Activation Dynamics. Table 5 tracks NG-DS neuron activations across F2F stages. Unlearning reduces generalist-dominant activation on the forget set ($0.232 \rightarrow 0.205$), while subsequent SFT increases domain-dominant activation on target-domain prompts ($0.622 \rightarrow 0.651$). This supports the intended F2F behavior: attenuating selected checkpoint behavior before strengthening target-domain representations.

Evaluation set	Neuron group	Base	Unlearned	F2F
NG-DS forget set	Generalist-dominant	0.2320	0.2050	0.2110
NG-DS forget set	Domain-dominant	0.1340	0.1310	0.1360
Target-domain prompts	Generalist-dominant	0.2040	0.1960	0.1930
Target-domain prompts	Domain-dominant	0.6260	0.6220	0.6510

Table 5: Mean activation of NG-DS neuron groups across F2F stages on Qwen-3-0.6B.

5 Limitation

This work shows that targeted preparatory unlearning can improve domain specialization, but several limitations remain. A further direction is to make NG-DS applicable beyond open-weight settings. In this work, we focus on model-aware forget-set construction, where NG-DS is run separately for each target backbone using internal activation statistics. Extending this idea to activation-free or black-box selection is outside the scope of the present study,

but would be an important future direction for reducing preprocessing cost and enabling use with closed-source models.

6 Conclusion

We propose FORGET-TO-FOCUS (F2F), a two-stage framework for domain specialization that treats unlearning as a preparatory operation rather than as a post hoc deletion mechanism. Instead of adapting directly from the original pretrained checkpoint, F2F first weakens selected generalist behavior through controlled unlearning and then applies standard supervised fine-tuning from the modified initialization. To make this process targeted, NG-DS constructs a model-elicited generalist proxy and uses feed-forward activation statistics to select forget examples that activate generalist-dominant neurons while avoiding domain-dominant neurons. In our evaluation across coding, medical question answering, and mathematical reasoning, F2F improves over standard SFT and adaptation baselines across model families and scales. For example, on HumanEval, F2F improves over SFT by +13.84 for Qwen3-0.6B, +7.86 for LLaMA3.1-8B, and +10.14 for LLaMA2-13B. Our representational analyses with CKA and SVCCA further show that F2F reaches an internal geometry distinct from ordinary tuning. This suggests that its gains do not come from simply extending the tuning trajectory, but from reshaping the pretrained checkpoint into a stronger initialization for domain-specific adaptation.

7 Acknowledgments

This research was supported by the Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency (KOCCA), funded by the Ministry of Culture, Sports and Tourism (MCST) in 2026 (Project Name: Development of an Integrated Data Management Platform for Personal Sports Records (PSR), Project Number: RS-2026-25547709), the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2025-24803164), and the Technology Innovation Program (RS-2024-00445759, Development of Navigation Technology Utilizing Visual Information Based on Vision-Language Models for Understanding Dynamic Environments in Non-Learned Spaces) funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea)

References

- Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and 1 others. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.
- Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, and 1 others. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.
- Sungmin Cha, Sungjun Cho, Dasol Hwang, and Moon-tae Lee. 2024. Towards robust and parameter-efficient knowledge unlearning for llms. *arXiv preprint arXiv:2408.06621*.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, and 39 others. 2021. [Evaluating large language models trained on code](#). *Preprint*, arXiv:2107.03374.
- Zeming Chen, Alejandro Hernández-Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, Alexandre Sallinen, Alireza Sakhaeirad, Vinitra Swamy, Igor Krawczuk, Deniz Bayazit, Axel Marmet, Syrielle Montariol, Mary-Anne Hartley, Martin Jaggi, and Antoine Bosselut. 2023a. [Meditron-70b: Scaling medical pretraining for large language models](#). *Preprint*, arXiv:2311.16079.
- Zeming Chen, Alejandro Hernández-Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, Alexandre Sallinen, Alireza Sakhaeirad, Vinitra Swamy, Igor Krawczuk, Deniz Bayazit, Axel Marmet, Syrielle Montariol, Mary-Anne Hartley, Martin Jaggi, and Antoine Bosselut. 2023b. [Meditron-70b: Scaling medical pretraining for large language models](#).
- Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.
- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.
- Muhammad Fawi. 2024. [CURLoRA: Leveraging CUR Matrix Decomposition for Stable LLM Continual Fine-Tuning and Catastrophic Forgetting Mitigation](#).
- 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, and 5 others. 2024. [The language model evaluation harness](#).
- Jiahui Geng, Qing Li, Herbert Woisetschlaeger, Zongxiong Chen, Fengyu Cai, Yuxia Wang, Preslav Nakov, Hans-Arno Jacobsen, and Fakhri Karray. 2025. A comprehensive survey of machine unlearning techniques for large language models. *arXiv preprint arXiv:2503.01854*.
- Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 5484–5495.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don’t stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.
- Chris Jay Hoofnagle, Bart van der Sloot, and Fredrik Zuiderveen Borgesius. 2019. [The european union general data protection regulation: what it is and what it means*](#). *Information & Communications Technology Law*, 28(1):65–98.

- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.
- Siming Huang, Tianhao Cheng, Jason Klein Liu, Jiaran Hao, Liuyihan Song, Yang Xu, J Yang, Jiaheng Liu, Chenchen Zhang, Linzheng Chai, and 1 others. 2024. Opencoder: The open cookbook for top-tier code large language models. *arXiv preprint arXiv:2411.04905*.
- Chunyang Jiang, Chi-min Chan, Yiyang Cai, Yulong Liu, Wei Xue, and Yike Guo. 2025. Graceful forgetting in generative language models. *arXiv preprint arXiv:2505.19715*.
- Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. Pubmedqa: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577.
- Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMIR.
- Na Li, Chunyi Zhou, Yansong Gao, Hui Chen, Zhi Zhang, Boyu Kuang, and Anmin Fu. 2025. [Machine unlearning: Taxonomy, metrics, applications, challenges, and prospects](#). *IEEE Transactions on Neural Networks and Learning Systems*, 36(8):13709–13729.
- Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. [Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering](#). In *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pages 248–260. PMLR.
- Venkatesh Balavadhani Parthasarathy, Ahtsham Zafar, Aafaq Khan, and Arsalan Shahid. 2024. The ultimate guide to fine-tuning llms from basics to breakthroughs: An exhaustive review of technologies, research, best practices, applied research challenges and opportunities. *arXiv preprint arXiv:2408.13296*.
- Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. 2023. Yarn: Efficient context window extension of large language models. *arXiv preprint arXiv:2309.00071*.
- Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. 2017. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. *Advances in neural information processing systems*, 30.
- Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.
- Kaiser Sun and Mark Dredze. 2025. [Amuro & char: Analyzing the relationship between pre-training and fine-tuning of large language models](#).
- Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, and 1 others. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.
- Shubham Toshniwal, Ivan Moshkov, Sean Narenthiran, Daria Gitman, Fei Jia, and Igor Gitman. 2024. Openmathinstruct-1: A 1.8 million math instruction tuning dataset. *Advances in Neural Information Processing Systems*, 37:34737–34774.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.
- Henrique Schechter Vera, Sahil Dua, Biao Zhang, Daniel Salz, Ryan Mullins, Sindhu Raghuram Panayam, Sara Smoot, Iftexhar Naim, Joe Zou, Feiyang Chen, and 1 others. 2025. Embeddinggemma: Powerful and lightweight text representations. *arXiv preprint arXiv:2509.20354*.
- Xiaozhi Wang, Kaiyue Wen, Zhengyan Zhang, Lei Hou, Zhiyuan Liu, and Juanzi Li. 2022. Finding skill neurons in pre-trained transformer-based language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11132–11152.
- Xiaoyu Xu, Xiang Yue, Yang Liu, Qingqing Ye, Haibo Hu, and Minxin Du. 2025. Unlearning isn’t deletion: Investigating reversibility of machine unlearning in llms. *arXiv preprint arXiv:2505.16831*.
- Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. 2024. [Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing](#). *ArXiv*, abs/2406.08464.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Jin Yao, Eli Chien, Minxin Du, Xinyao Niu, Tianhao Wang, Zezhou Cheng, and Xiang Yue. 2024. Machine unlearning of pre-trained large language models. *arXiv preprint arXiv:2402.15159*.

- Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.
- Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. 2024. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2404.05868*.
- Wen Zhang, Lingfei Deng, Lei Zhang, and Dongrui Wu. 2022. A survey on negative transfer. *IEEE/CAA Journal of Automatica Sinica*, 10(2):305–329.

A Theoretical Properties of NG-DS

We now state theoretical properties of NG-DS. The analysis explains why the selected examples are suitable for the first stage of two-stage F2F. The key idea is that NG-DS selects examples with high pretraining-dominant activation relative to domain-dominant activation, thereby reducing domain leakage in the forget set.

A.1 Proxy Separation of Dominant Neurons

The dominance-ratio construction gives a direct separation guarantee for the selected neuron groups.

Proposition 1 (Proxy separation). *For any layer ℓ with $|N_p^\ell| > 0$,*

$$\mathbb{E}_{x \sim \mathcal{P}} [A_p^\ell(x)] > \alpha \mathbb{E}_{x \sim \mathcal{D}} [A_p^\ell(x)]. \quad (18)$$

Similarly, for any layer ℓ with $|N_d^\ell| > 0$,

$$\mathbb{E}_{x \sim \mathcal{D}} [A_d^\ell(x)] > \alpha \mathbb{E}_{x \sim \mathcal{P}} [A_d^\ell(x)]. \quad (19)$$

Proof. For $j \in N_p^\ell$, the definition of N_p^ℓ gives

$$f_p(\ell, j) > \alpha f_d(\ell, j). \quad (20)$$

Averaging over $j \in N_p^\ell$,

$$\begin{aligned} \mathbb{E}_{x \sim \mathcal{P}} [A_p^\ell(x)] &= \frac{1}{|N_p^\ell|} \sum_{j \in N_p^\ell} f_p(\ell, j) \\ &> \frac{\alpha}{|N_p^\ell|} \sum_{j \in N_p^\ell} f_d(\ell, j) \\ &= \alpha \mathbb{E}_{x \sim \mathcal{D}} [A_p^\ell(x)]. \end{aligned} \quad (21)$$

This proves Equation 18. The proof of Equation 19 is identical after swapping $\mathcal{P}$ and $\mathcal{D}$. $\square$

Interpretation. The neuron sets used by NG-DS are created so that pretraining-dominant neurons have provably higher activation frequency on $\mathcal{P}$ than on $\mathcal{D}$, while domain-dominant neurons have the opposite property. Thus, the ratio score in $q(x) = \frac{\bar{A}_p(x)}{\bar{A}_d(x) + \epsilon}$ is computed using neurons that are proxy-discriminative by construction.

A.2 Optimality of Top- k Selection

Because NG-DS selects the k candidates with the largest $q(x)$, it exactly maximizes the modular score over all size- k subsets.

Proposition 2 (Top- k optimality). *Let $\mathcal{F}$ be selected by $\mathcal{F} = \text{TopK}_{x \in \mathcal{C}}^k(q(x))$, $|\mathcal{F}| = k$. Then for any $S \subseteq \mathcal{C}$ with $|S| = k$,*

$$\sum_{x \in \mathcal{F}} q(x) \geq \sum_{x \in S} q(x). \quad (22)$$

Proof. Let the candidates be sorted so that

$$q(x_1) \geq q(x_2) \geq \dots \geq q(x_{|C|}). \quad (23)$$

Then $\mathcal{F} = \{x_1, \dots, x_k\}$. For any other size- k set S , each selected element of S has score no larger than one of the top- k scores. Therefore,

$$\sum_{x \in S} q(x) \leq \sum_{i=1}^k q(x_i) = \sum_{x \in \mathcal{F}} q(x). \quad (24)$$

$\square$

A.3 Ratio-Based Leakage Control

Define the selected set's average pretraining-dominant and domain-dominant activation:

$$A_p(S) = \frac{1}{|S|} \sum_{x \in S} \bar{A}_p(x), \quad (25)$$

$$A_d(S) = \frac{1}{|S|} \sum_{x \in S} \bar{A}_d(x). \quad (26)$$

Let

$$q_{\min}(S) = \min_{x \in S} q(x). \quad (27)$$

Proposition 3 (Activation-ratio guarantee). *For any selected set S with $q_{\min}(S) > 0$,*

$$A_d(S) \leq \frac{A_p(S)}{q_{\min}(S)} - \epsilon. \quad (28)$$

Equivalently,

$$A_p(S) \geq q_{\min}(S) (A_d(S) + \epsilon). \quad (29)$$

Proof. For every $x \in S$, the definition of $q(x)$ gives

$$q(x) = \frac{\bar{A}_p(x)}{\bar{A}_d(x) + \epsilon}. \quad (30)$$

Since $q(x) \geq q_{\min}(S)$,

$$\bar{A}_p(x) \geq q_{\min}(S) (\bar{A}_d(x) + \epsilon). \quad (31)$$

Averaging over $x \in S$ gives

$$A_p(S) \geq q_{\min}(S) (A_d(S) + \epsilon). \quad (32)$$

Rearranging yields Equation 28. $\square$

Interpretation. A high minimum NG-DS score forces the selected set to have low domain-dominant activation relative to its pretraining-dominant activation. Thus, NG-DS does not merely select examples with high raw activation; it selects examples with a high pretraining-to-domain activation advantage.

A.4 Connection to F2F Interference Control

We connect NG-DS to the two-stage F2F analysis. Let $\mathcal{U}$ denote the pretraining-interference subspace and $\mathcal{V}$ denote the domain-relevant subspace. Let $\mathcal{I}_{\mathcal{F},S}$ be the interference potential induced by a selected forget set S .

We use the following assumptions.

Assumption 1 (Pretraining activation controls forget alignment). *There exist constants $\varepsilon_0, L_p \geq 0$ such that the forget-gradient alignment error satisfies*

$$\varepsilon_F(S) \leq \varepsilon_0 + L_p (1 - A_p(S)). \quad (33)$$

Assumption 2 (Domain activation controls domain leakage). *There exists a constant $L_d \geq 0$ such that*

$$\|P_{\mathcal{V}} \nabla \mathcal{I}_{\mathcal{F},S}(\theta)\| \leq L_d A_d(S). \quad (34)$$

Assumption 1 states that examples with stronger pretraining-dominant activation yield forget gradients more aligned with the interference subspace. Assumption 2 states that examples with stronger domain-dominant activation create greater risk of domain-relevant drift.

Theorem 1 (NG-DS-dependent F2F bound). *Let $S = \mathcal{F}$ be selected by NG-DS. Under Assumptions 1 and 2, the interference contraction term in two-stage F2F depends on*

$$\varepsilon_F(S) \leq \varepsilon_0 + L_p (1 - A_p(S)), \quad (35)$$

while the domain-relevant leakage satisfies

$$\|P_{\mathcal{V}} \nabla \mathcal{I}_{\mathcal{F},S}(\theta)\| \leq L_d \left(\frac{A_p(S)}{q_{\min}(S)} - \epsilon \right). \quad (36)$$

Proof. Equation 35 follows directly from Assumption 1. For the leakage term, Assumption 2 gives

$$\|P_{\mathcal{V}} \nabla \mathcal{I}_{\mathcal{F},S}(\theta)\| \leq L_d A_d(S). \quad (37)$$

By Proposition 3,

$$A_d(S) \leq \frac{A_p(S)}{q_{\min}(S)} - \epsilon. \quad (38)$$

Substituting this bound into the leakage inequality proves Equation 36. $\square$

Interpretation. The theorem shows how NG-DS affects two-stage F2F. Large $A_p(S)$ improves forget-gradient alignment with the interference subspace. Large $q_{\min}(S)$ prevents the selected examples from having high domain-dominant activation relative to their pretraining-dominant activation. Therefore, NG-DS supports the two requirements of useful preparatory forgetting, strong interference targeting and limited domain leakage.

A.5 Combined Selector-Dependent Initialization Bound

We now combine the NG-DS quantities with the F2F contraction result. Let the unlearning update be

$$\theta_{t+1} = \theta_t + \eta \lambda_f \nabla \mathcal{L}_{\mathcal{F}}(\theta_t) - \eta \lambda_r \nabla \mathcal{L}_{\mathcal{R}}(\theta_t) - \eta \lambda_{\text{KL}} \nabla \mathcal{L}_{\text{KL}}(\theta_t). \quad (39)$$

Assume the interference contraction theorem from Appendix B holds:

$$\begin{aligned} \|P_{\mathcal{U}} \tilde{\theta}_0\| &\leq \rho^{T_u} \|P_{\mathcal{U}} \theta_0\| \\ &\quad + \frac{\lambda_f \varepsilon_F(S) + \lambda_r G_R + \lambda_{\text{KL}} G_K}{\lambda_f \mu_F}, \end{aligned} \quad (40)$$

where $\rho = 1 - \eta \lambda_f \mu_F$.

F2F bound with NG-DS selection

Corollary 1. *For the NG-DS selected forget set $S = \mathcal{F}$,*

$$\begin{aligned} \|P_{\mathcal{U}} \tilde{\theta}_0\| &\leq \rho^{T_u} \|P_{\mathcal{U}} \theta_0\| \\ &\quad + \frac{\lambda_f [\varepsilon_0 + L_p (1 - A_p(S))]}{\lambda_f \mu_F} \\ &\quad + \frac{\lambda_r G_R + \lambda_{\text{KL}} G_K}{\lambda_f \mu_F}. \end{aligned} \quad (41)$$

Moreover, if domain drift is controlled by

$$\begin{aligned} \|P_{\mathcal{V}}(\tilde{\theta}_0 - \theta_0)\| &\leq \eta T_u [\lambda_f \|P_{\mathcal{V}} \nabla \mathcal{I}_{\mathcal{F},S}(\theta)\| \\ &\quad + \lambda_r G_{R,V} + \lambda_{\text{KL}} G_{K,V}]. \end{aligned} \quad (42)$$

then

$$\begin{aligned} \|P_{\mathcal{V}}(\tilde{\theta}_0 - \theta_0)\| &\leq \eta T_u \left[\lambda_f L_d \left(\frac{A_p(S)}{q_{\min}(S)} - \epsilon \right) \right. \\ &\quad \left. + \lambda_r G_{R,V} + \lambda_{\text{KL}} G_{K,V} \right]. \end{aligned} \quad (43)$$

$$\lambda_{\text{KL}} G_{K,V} \Big]. \quad (44)$$

Proof. Equation 41 follows by substituting Equation 35 into the base F2F contraction bound in

Equation 40. Equation 43 follows by substituting Equation 36 into the assumed domain-drift bound. $\square$

NG-DS gives a concrete activation-based criterion for forget-set construction. The selected set is useful for two-stage F2F when it has high pretraining-dominant activation $A_p(S)$ and high pretraining-to-domain advantage $q_{\min}(S)$. The former improves interference targeting, while the latter limits domain-dominant activation and reduces the risk of destructive forgetting.

B Supplementary Theory F2F

This section gives the formal analysis behind the main theoretical intuition. The analysis is local and surrogate-based. It is intended to explain when the two-stage procedure improves downstream specialization, not to provide a global convergence proof for nonconvex LLM training.

B.1 Local Interference Model

Assume that, in a local neighborhood of adaptation, the parameter space decomposes orthogonally as

$$\mathbb{R}^p = \mathcal{V} \oplus \mathcal{U}, \quad (45)$$

where $\mathcal{V}$ contains domain-relevant directions and $\mathcal{U}$ contains pretraining-like interference directions. Let $P_{\mathcal{V}}$ and $P_{\mathcal{U}}$ be the corresponding projections.

The unlearning update is

$$\theta_{t+1} = \theta_t + \eta\lambda_f \nabla \mathcal{L}_{\mathcal{F}}(\theta_t) - \eta\lambda_r \nabla \mathcal{L}_{\mathcal{R}}(\theta_t) - \eta\lambda_{\text{KL}} \nabla \mathcal{L}_{\text{KL}}(\theta_t). \quad (46)$$

Because the forget term performs gradient ascent on $\mathcal{L}_{\mathcal{F}}$, we analyze it through an interference potential $\mathcal{I}_{\mathcal{F}}$. The potential $\mathcal{I}_{\mathcal{F}}$ measures the amount of pretraining-like interference induced by the selected forget set.

Assumption 3 (Forget-gradient alignment). *There exists an interference potential $\mathcal{I}_{\mathcal{F}}$ such that*

$$\|P_{\mathcal{U}}(\nabla \mathcal{I}_{\mathcal{F}}(\theta) + \nabla \mathcal{L}_{\mathcal{F}}(\theta))\| \leq \varepsilon_F. \quad (47)$$

Thus, $+\nabla \mathcal{L}_{\mathcal{F}}$ is an approximate descent direction for $\mathcal{I}_{\mathcal{F}}$ in $\mathcal{U}$.

Assumption 4 (Interference curvature). *For fixed $v = P_{\mathcal{V}}\theta$, the function $\mathcal{I}_{\mathcal{F}}(v + u)$ is μ_F -strongly convex and β_F -smooth in $u = P_{\mathcal{U}}\theta$, with minimizer $u = 0$:*

$$\mu_F I \preceq \nabla_{uu}^2 \mathcal{I}_{\mathcal{F}}(v + u) \preceq \beta_F I. \quad (48)$$

Assumption 5 (Bounded retain and KL perturbations). *There exist $G_R, G_K \geq 0$ such that*

$$\|P_{\mathcal{U}}\nabla \mathcal{L}_{\mathcal{R}}(\theta)\| \leq G_R, \quad \|P_{\mathcal{U}}\nabla \mathcal{L}_{\text{KL}}(\theta)\| \leq G_K. \quad (49)$$

B.2 Interference Contraction

Theorem 2 (Interference contraction). *Suppose Assumptions 3, 4, and 5 hold. If $0 < \eta\lambda_f \leq 1/\beta_F$, then*

$$\begin{aligned} \|P_{\mathcal{U}}\theta_{t+1}\| &\leq (1 - \eta\lambda_f\mu_F)\|P_{\mathcal{U}}\theta_t\| \\ &\quad + \eta(\lambda_f\varepsilon_F + \lambda_r G_R + \lambda_{\text{KL}} G_K). \end{aligned} \quad (50)$$

After T_u unlearning steps,

$$\begin{aligned} \|P_{\mathcal{U}}\theta_{T_u}\| &\leq \rho^{T_u}\|P_{\mathcal{U}}\theta_0\| \\ &\quad + \frac{\lambda_f\varepsilon_F + \lambda_r G_R + \lambda_{\text{KL}} G_K}{\lambda_f\mu_F}, \end{aligned} \quad (51)$$

where $\rho = 1 - \eta\lambda_f\mu_F$.

Proof. Let $u_t = P_{\mathcal{U}}\theta_t$. Projecting Equation 46 onto $\mathcal{U}$ gives

$$\begin{aligned} u_{t+1} &= u_t + \eta\lambda_f P_{\mathcal{U}}\nabla \mathcal{L}_{\mathcal{F}}(\theta_t) - \eta\lambda_r P_{\mathcal{U}}\nabla \mathcal{L}_{\mathcal{R}}(\theta_t) \\ &\quad - \eta\lambda_{\text{KL}} P_{\mathcal{U}}\nabla \mathcal{L}_{\text{KL}}(\theta_t). \end{aligned} \quad (52)$$

Add and subtract $\eta\lambda_f P_{\mathcal{U}}\nabla \mathcal{I}_{\mathcal{F}}(\theta_t)$:

$$\begin{aligned} u_{t+1} &= u_t - \eta\lambda_f P_{\mathcal{U}}\nabla \mathcal{I}_{\mathcal{F}}(\theta_t) \\ &\quad + \eta\lambda_f P_{\mathcal{U}}(\nabla \mathcal{L}_{\mathcal{F}}(\theta_t) + \nabla \mathcal{I}_{\mathcal{F}}(\theta_t)) \\ &\quad - \eta\lambda_r P_{\mathcal{U}}\nabla \mathcal{L}_{\mathcal{R}}(\theta_t) - \eta\lambda_{\text{KL}} P_{\mathcal{U}}\nabla \mathcal{L}_{\text{KL}}(\theta_t). \end{aligned} \quad (53)$$

By Assumption 4, gradient descent on $\mathcal{I}_{\mathcal{F}}$ in $\mathcal{U}$ is contractive for step size $\eta\lambda_f \leq 1/\beta_F$:

$$\|u_t - \eta\lambda_f P_{\mathcal{U}}\nabla \mathcal{I}_{\mathcal{F}}(\theta_t)\| \leq (1 - \eta\lambda_f\mu_F)\|u_t\|. \quad (54)$$

Using the triangle inequality and Assumptions 3 and 5,

$$\begin{aligned} \|u_{t+1}\| &\leq (1 - \eta\lambda_f\mu_F)\|u_t\| \\ &\quad + \eta(\lambda_f\varepsilon_F + \lambda_r G_R + \lambda_{\text{KL}} G_K). \end{aligned} \quad (55)$$

This proves the one-step bound. Iterating the recursion and using the finite geometric sum gives Equation 51. $\square$

B.3 Effect on Downstream Initialization

Assumption 6 (Target-domain smoothness). *The target-domain loss $\mathcal{L}_{\mathcal{D}}$ is locally $\beta_{\mathcal{D}}$ -smooth around the post-unlearning initialization and has a minimizer $\theta_{\mathcal{D}}^*$ satisfying*

$$P_{\mathcal{U}}\theta_{\mathcal{D}}^* = 0. \quad (56)$$

Corollary 2 (Post-unlearning risk). *Let $\tilde{\theta}_0 = \theta_{T_u}$. Under Assumption 6,*

$$\mathcal{L}_{\mathcal{D}}(\tilde{\theta}_0) - \mathcal{L}_{\mathcal{D}}(\theta_{\mathcal{D}}^*) \leq \frac{\beta_D}{2} \left(\|P_{\mathcal{V}}\tilde{\theta}_0 - P_{\mathcal{V}}\theta_{\mathcal{D}}^*\|^2 + \|P_{\mathcal{U}}\tilde{\theta}_0\|^2 \right). \quad (57)$$

Proof. By β_D -smoothness,

$$\mathcal{L}_{\mathcal{D}}(\tilde{\theta}_0) - \mathcal{L}_{\mathcal{D}}(\theta_{\mathcal{D}}^*) \leq \frac{\beta_D}{2} \|\tilde{\theta}_0 - \theta_{\mathcal{D}}^*\|^2. \quad (58)$$

Using the orthogonal decomposition and $P_{\mathcal{U}}\theta_{\mathcal{D}}^* = 0$,

$$\|\tilde{\theta}_0 - \theta_{\mathcal{D}}^*\|^2 = \|P_{\mathcal{V}}\tilde{\theta}_0 - P_{\mathcal{V}}\theta_{\mathcal{D}}^*\|^2 + \|P_{\mathcal{U}}\tilde{\theta}_0\|^2. \quad (59)$$

Substitution gives the result. $\square$

B.4 Domain Leakage and Safe Forgetting

For a forget set S , define leakage:

$$\Delta(S) = \frac{1}{|S|} \sum_{x \in S} [A_d(x) + A_s(x) + \max(0, \cos(h(x), c_d) - \cos(h(x), c_p))]. \quad (60)$$

This score is small when selected examples weakly activate domain-like and shared neurons and remain closer to the pretraining proxy than the target-domain proxy.

Assumption 7 (Leakage controls domain-gradient contamination). *For the selected forget set S , the domain-relevant projection of the interference gradient is bounded by*

$$\|P_{\mathcal{V}}\nabla\mathcal{I}_{\mathcal{F},S}(\theta)\| \leq L_{\Delta}\Delta(S). \quad (61)$$

Assumption 8 (Bounded domain-subspace perturbations). *There exist $G_{R,V}, G_{K,V} \geq 0$ such that*

$$\|P_{\mathcal{V}}\nabla\mathcal{L}_{\mathcal{R}}(\theta)\| \leq G_{R,V}, \|P_{\mathcal{V}}\nabla\mathcal{L}_{\mathcal{KL}}(\theta)\| \leq G_{K,V}. \quad (62)$$

Theorem 3 (Domain-relevant drift). *Suppose Assumptions 7 and 8 hold. Then the domain-relevant displacement after T_u unlearning steps is bounded by*

$$\|P_{\mathcal{V}}(\theta_{T_u} - \theta_0)\| \leq \eta T_u \left[\lambda_f (L_{\Delta}\Delta(\mathcal{F}) + \varepsilon_V) + \lambda_r G_{R,V} + \lambda_{\mathcal{KL}} G_{K,V} \right], \quad (63)$$

where ε_V measures alignment error in $\mathcal{V}$.

Proof. Project the unlearning update onto $\mathcal{V}$:

$$P_{\mathcal{V}}(\theta_{t+1} - \theta_t) = \eta \lambda_f P_{\mathcal{V}}\nabla\mathcal{L}_{\mathcal{F}}(\theta_t) - \eta \lambda_r P_{\mathcal{V}}\nabla\mathcal{L}_{\mathcal{R}}(\theta_t) - \eta \lambda_{\mathcal{KL}} P_{\mathcal{V}}\nabla\mathcal{L}_{\mathcal{KL}}(\theta_t). \quad (64)$$

Add and subtract $\eta \lambda_f P_{\mathcal{V}}\nabla\mathcal{I}_{\mathcal{F}}(\theta_t)$, then apply the triangle inequality:

$$\|P_{\mathcal{V}}(\theta_{t+1} - \theta_t)\| \leq \eta \left[\lambda_f \|P_{\mathcal{V}}\nabla\mathcal{I}_{\mathcal{F}}(\theta_t)\| + \lambda_f \|P_{\mathcal{V}}(\nabla\mathcal{L}_{\mathcal{F}}(\theta_t) - \nabla\mathcal{I}_{\mathcal{F}}(\theta_t))\| + \lambda_r \|P_{\mathcal{V}}\nabla\mathcal{L}_{\mathcal{R}}(\theta_t)\| + \lambda_{\mathcal{KL}} \|P_{\mathcal{V}}\nabla\mathcal{L}_{\mathcal{KL}}(\theta_t)\| \right]. \quad (65)$$

Using Assumptions 7 and 8, and denoting the alignment error in $\mathcal{V}$ by ε_V , gives the per-step bound. Summing over T_u steps gives the theorem. $\square$

B.5 Why NG-DS Improves F2F

Define the empirical pretraining-alignment score

$$\hat{A}(x) = \sum_{\ell \in \mathcal{L}} \omega_{\ell} A_p^{\ell}(x), \quad (66)$$

and empirical leakage score

$$\hat{\Delta}(x) = \sum_{\ell \in \mathcal{L}} \omega_{\ell} \left[A_d^{\ell}(x) + A_s^{\ell}(x) + \max(0, \cos(h_{\ell}(x), c_d^{\ell}) - \cos(h_{\ell}(x), c_p^{\ell})) \right]. \quad (67)$$

Ignoring stability and diversity for clarity, the NG-DS score has the form

$$q(x) = \hat{A}(x) - \lambda_{\Delta} \hat{\Delta}(x). \quad (68)$$

Thus, NG-DS directly selects examples that are likely to reduce the alignment error ε_F while keeping leakage $\Delta(\mathcal{F})$ small.

Assumption 9 (Alignment controls forget-gradient error). *For a selected forget set S ,*

$$\varepsilon_F(S) \leq \varepsilon_0 + L_A (A_{\max} - A^*(S)), \quad (69)$$

where $A^*(S)$ is the true pretraining-alignment score of S .

Theorem 4 (Selector-dependent F2F bound). *Under the preceding assumptions, the post-unlearning interference component satisfies*

$$\begin{aligned} \|P_U \tilde{\theta}_0(S)\| &\leq \rho^{T_u} \|P_U \theta_0\| \\ &+ \frac{\lambda_f [\varepsilon_0 + L_A(A_{\max} - A^*(S))]}{\lambda_f \mu_F} \\ &+ \frac{\lambda_r G_R + \lambda_{KL} G_K}{\lambda_f \mu_F}. \end{aligned} \quad (70)$$

The domain-relevant drift satisfies

$$\begin{aligned} \|P_V(\tilde{\theta}_0(S) - \theta_0)\| &\leq \eta T_u \left[\lambda_f (L_\Delta \Delta^*(S) + \varepsilon_V) \right. \\ &\quad \left. + \lambda_r G_{R,V} + \lambda_{KL} G_{K,V} \right]. \end{aligned} \quad (71)$$

Proof. The first inequality follows by substituting Assumption 9 into Theorem 2. The second inequality follows directly from Theorem 3. $\square$

Interpretation. The selector affects F2F through two quantities. Higher pretraining alignment $A^*(S)$ improves interference contraction. Lower domain leakage $\Delta^*(S)$ reduces destructive drift in domain-relevant directions. NG-DS is designed to optimize both conditions.

B.6 Greedy Approximation for Diversity-Aware Selection

The diversity-aware objective is

$$F(S) = \sum_{x \in S} q_+(x) + \lambda_{\text{div}} \sum_{z \in \mathcal{C}} \max_{x \in S} \kappa(z, x), |S| \leq k, \quad (72)$$

where $q_+(x) \geq 0$ and $\kappa(z, x) \geq 0$.

Proposition 4 (Submodularity). *The function $F(S)$ is monotone submodular.*

Proof. The term $\sum_{x \in S} q_+(x)$ is nonnegative modular, hence monotone submodular. For fixed z , the function $S \mapsto \max_{x \in S} \kappa(z, x)$ is monotone submodular because adding an element cannot decrease the maximum, and the marginal gain decreases as the set grows. A nonnegative weighted sum of monotone submodular functions is monotone submodular. $\square$

Corollary 3 (Greedy approximation). *Let S_{greedy} be the set returned by greedy maximization of $F(S)$ under $|S| \leq k$, and let S^* be the optimal set. Then*

$$F(S_{\text{greedy}}) \geq \left(1 - \frac{1}{e}\right) F(S^*). \quad (73)$$

Proof. This follows from the classical approximation guarantee for greedy maximization of a non-negative monotone submodular function under a cardinality constraint. $\square$

C CKA and SVCCA

In this section, we describe the detailed definitions of Centered Kernel Alignment (CKA) (Kornblith et al., 2019) and Singular Vector Canonical Correlation Analysis (SVCCA) (Raghu et al., 2017) used in the analysis of Section 4.6.

CKA. Let $X \in \mathbb{R}^{N \times d_x}$ and $Y \in \mathbb{R}^{N \times d_y}$ be mean-pooled, sample-centered layer representations of the same inputs, i.e., $X_c = X - \bar{X}$ and $Y_c = Y - \bar{Y}$. The linear CKA is

$$\text{CKA}(X, Y) = \frac{\|X_c^\top Y_c\|_F^2}{\|X_c^\top X_c\|_F \|Y_c^\top Y_c\|_F},$$

which captures similarity of representational geometry and is invariant to orthogonal transforms and isotropic rescaling.

SVCCA. SVCCA compresses each representation space to retain a fraction $\alpha = 0.99$ of variance via SVD, then computes the mean canonical correlation between the compressed features. If $X' \in \mathbb{R}^{N \times k_x}$, $Y' \in \mathbb{R}^{N \times k_y}$, and $\rho_1, \dots, \rho_k$ are the canonical correlations with $k = \min(k_x, k_y)$, then

$$\text{SVCCA}(X, Y) = \frac{1}{k} \sum_{i=1}^k \rho_i.$$

SVCCA complements CKA by focusing on shared high-variance factors.

D Ablation

Appendix D provides additional ablation studies that further analyze the behavior of F2F. Specifically, we examine the effect of forget-set quality (D.1), the sensitivity of NG-DS to its hyperparameters (D.2), different unlearning and fine-tuning variants (D.3), the impact of retain-set size and σ (D.4), the effect of dataset size and unlearning weights (D.5), multi-seed robustness (D.6), and extended coding results across Qwen2.5 model sizes (D.7).

D.1 Effect of Forget-Set Quality

To examine how forget-set quality affects downstream performance, we compare four forget-set construction strategies: BC-Select, BookCorpus, BC-Cosine, and NG-DS.

BC-Select is a curated subset in which texts overlapping with the target domain, such as biomedical content for PubMedQA, are manually removed, leaving mostly general narrative and fiction data with minimal domain-relevant knowledge. *BookCorpus* uses the original BookCorpus samples without additional filtering. *BC-Cosine* automatically selects samples that are distant from the target domain in representation space. Specifically, each sample x is encoded with a Transformer encoder (Vera et al., 2025) as $h_x = f_\theta(x)$, the target-domain centroid is computed as $c_T = \frac{1}{|D_T|} \sum_{x' \in D_T} f_\theta(x')$, and candidates are ranked by cosine distance $d_{\cos}(x) = 1 - \frac{h_x^\top c_T}{\|h_x\| \|c_T\|}$, selecting samples with large $d_{\cos}(x)$. We compare these baselines with the proposed NG-DS forget set, which selects examples using model-internal neuron activation patterns.

Mechanistic Diagnostics of Forget-Set Quality

Forget-domain overlap. Let $h(x)$ denote the representation of an input x , and let the target-domain centroid be $c_D = \frac{1}{|D|} \sum_{x \in D} h(x)$. We define the forget-domain overlap as

$$\text{FDO}(\mathcal{F}, \mathcal{D}) = \mathbb{E}_{x \in \mathcal{F}} [\cos(h(x), c_D)]. \quad (74)$$

A large $\text{FDO}(\mathcal{F}, \mathcal{D})$ indicates that the forget set is close to the target-domain representation region. Such a forget set is risky: unlearning it may suppress domain-relevant features that should instead be preserved for fine-tuning. Conversely, a lower overlap suggests that the forget set is less likely to directly erase target-domain knowledge.

Unlearning alignment. Low overlap alone is not sufficient. A forget set should also induce an unlearning direction that is compatible with target-domain optimization. We therefore define the unlearning alignment score as

$$\text{UA}(\mathcal{F}, \mathcal{D}) = \cos(\nabla_\theta \ell_{\mathcal{F}}(\theta), -\nabla_\theta \mathcal{L}_{\mathcal{D}}(\theta)). \quad (75)$$

Here, $\nabla_\theta \ell_{\mathcal{F}}(\theta)$ is the gradient direction used to increase the forget-set loss during unlearning, while $-\nabla_\theta \mathcal{L}_{\mathcal{D}}(\theta)$ is the descent direction for the target-domain objective. A higher value of $\text{UA}(\mathcal{F}, \mathcal{D})$ means that forgetting $\mathcal{F}$ moves the model in a direction more consistent with reducing the target-domain loss.

Downstream effect of forget-set quality. Table 6 shows that forget-set quality strongly affects downstream specialization. Across Qwen3-0.6B, LLaMA3.1-8B-Instruct, and LLaMA2-13B,

NG-DS consistently gives the strongest or near-strongest performance across coding, medical, and mathematical benchmarks. For Qwen3-0.6B, NG-DS with $\text{Unl}_{GA+GD} + \text{Tuning}$ achieves the best results on all six benchmarks, improving over standard tuning from 31.71 to 45.55 on HumanEval, from 62.60 to 80.50 on PubMedQA, and from 50.06 to 60.11 on Hendrycks-MATH. Similar trends hold for larger models, where NG-DS also produces the best or second-best results across most tasks.

In contrast, the original BookCorpus forget set gives weaker and less stable gains. Because it is not explicitly filtered for target-domain leakage, unlearning on BookCorpus can remove both irrelevant pretraining priors and features that are useful for the target domain. BC-Select improves over BookCorpus by manually removing domain-overlapping samples, while BC-Cosine shows that representation-level distance from the target domain is helpful but not sufficient.

Connection between diagnostics and performance.

Table 7 connects the proposed diagnostics with downstream gains. Since all rows use the same downstream recipe, $\text{Unl}_{GA+GD} + \text{Tuning}$, differences in Δ_{F2F} isolate the effect of the forget set. NG-DS achieves the most favorable profile: it has the lowest forget-domain overlap ($\text{FDO} = 0.5808$) and is the only method with non-negative unlearning alignment ($\text{UA} = 0.0008$). Correspondingly, it yields the largest downstream improvement, with $\Delta_{\text{F2F}} = +17.90$ on PubMedQA, $+5.10$ on MedMCQA, and an average gain of $+9.42$ across six benchmarks.

BookCorpus shows the opposite behavior. It has the highest overlap ($\text{FDO} = 0.7115$) and the most negative alignment ($\text{UA} = -0.0982$), leading to negative downstream transfer, especially in the medical domain. BC-Cosine provides an intermediate case: it reduces FDO close to NG-DS, but its UA remains negative and its average gain is much smaller. This suggests that simply selecting examples far from the target-domain embedding centroid does not guarantee a useful unlearning direction.

F2F does not benefit from arbitrary forgetting. Its gains depend on selecting a forget set that both avoids target-domain leakage and induces an optimization-compatible unlearning update. NG-DS satisfies these conditions more consistently than generic BookCorpus forgetting or cosine-

Table 6: Effect of forget-set quality on F2F across models and domains. We compare different forget sets on Qwen3-0.6B, LLaMA3.1-8B-Instruct, and LLaMA2-13B across coding, medical, and mathematics benchmarks. Higher is better. **Best**; **Second best**.

Forget Set	Method	MBPP	HumanEval	PubMedQA	MedMCQA	Hendrycks-MATH	GSM8K
Qwen3-0.6B							
BC-Select	Unl_{GA+GD}	30.00	21.34	61.60	31.44	39.07	0.26
	Unl_{GA+GD} + Tuning	31.60	42.07	69.60	45.31	54.11	15.30
	Unl_{GA}	24.00	20.73	60.40	31.29	25.09	0.24
	Unl_{GA} + SFT	31.60	40.02	58.80	40.26	51.20	14.00
BookCorpus	Unl_{GA+GD}	24.20	20.12	61.80	30.38	31.47	0.06
	Unl_{GA+GD} + Tuning	29.90	40.00	60.20	23.31	52.00	13.21
	Unl_{GA}	23.80	20.12	60.20	31.89	25.00	0.00
	Unl_{GA} + Tuning	28.00	33.10	61.20	35.43	50.00	13.51
BC-Cosine	Unl_{GA+GD}	24.01	18.00	61.20	29.32	29.05	0.10
	Unl_{GA+GD} + Tuning	31.55	41.90	67.33	45.00	54.85	13.20
	Unl_{GA}	22.22	18.23	58.45	29.00	25.00	0.00
	Unl_{GA} + Tuning	29.91	40.50	57.65	38.75	48.90	13.07
NG-DS	Unl_{GA+GD}	32.77	23.04	61.66	32.35	40.07	0.02
	Unl_{GA+GD} + Tuning	33.50	45.55	80.50	47.22	60.11	17.20
	Unl_{GA}	24.10	20.50	59.20	33.00	38.01	0.08
	Unl_{GA} + SFT	32.55	43.33	74.39	42.30	53.33	15.33
–	Baseline	22.60	19.50	57.80	32.25	41.09	0.02
	Baseline + Tuning	28.80	31.71	62.60	42.12	50.06	12.30
LLaMA3.1-8B-Instruct							
BC-Select	Unl_{GA+GD}	43.60	54.88	79.21	60.50	18.00	61.70
	Unl_{GA+GD} + Tuning	60.10	60.37	89.90	70.25	29.50	70.51
	Unl_{GA}	22.60	1.20	75.22	60.20	10.51	50.91
	Unl_{GA} + Tuning	58.66	57.70	87.00	67.15	25.70	67.20
BookCorpus	Unl_{GA+GD}	40.50	52.52	79.50	59.30	17.00	51.00
	Unl_{GA+GD} + Tuning	56.20	55.76	87.61	70.10	28.81	65.20
	Unl_{GA}	33.20	25.50	72.30	57.00	5.20	35.20
	Unl_{GA} + Tuning	52.30	40.90	86.90	61.20	23.01	66.15
BC-Cosine	Unl_{GA+GD}	42.55	53.76	79.00	58.22	17.91	61.00
	Unl_{GA+GD} + Tuning	59.55	59.86	85.31	71.02	28.33	68.57
	Unl_{GA}	20.35	0.90	73.05	58.99	9.32	50.00
	Unl_{GA} + Tuning	57.76	57.55	85.31	66.00	27.01	67.10
NG-DS	Unl_{GA+GD}	47.80	59.08	84.25	60.33	30.50	65.78
	Unl_{GA+GD} + Tuning	64.30	64.57	90.22	74.20	38.00	69.50
	Unl_{GA}	26.80	5.40	81.00	61.30	17.30	58.30
	Unl_{GA} + SFT	62.86	61.90	88.73	72.33	35.50	65.30
–	Baseline	49.00	33.54	75.20	59.43	19.90	62.85
	Baseline + Tuning	56.60	56.71	85.31	64.20	25.51	66.70
LLaMA2-13B							
BC-Select	Unl_{GA+GD}	27.22	30.60	74.90	38.68	29.00	5.10
	Unl_{GA+GD} + Tuning	50.31	46.15	90.11	60.10	51.50	21.50
	Unl_{GA}	0.00	25.50	70.00	36.51	24.35	2.00
	Unl_{GA} + Tuning	45.01	44.70	89.33	57.93	50.90	20.00
BookCorpus	Unl_{GA+GD}	27.20	30.45	73.00	37.50	27.00	5.10
	Unl_{GA+GD} + Tuning	47.50	45.91	89.55	61.30	50.30	20.00
	Unl_{GA}	0.00	10.00	65.99	29.55	23.55	1.05
	Unl_{GA} + Tuning	39.55	40.01	87.00	50.60	47.60	20.00
BC-Cosine	Unl_{GA+GD}	25.30	30.52	73.44	37.62	29.09	6.30
	Unl_{GA+GD} + Tuning	48.91	44.30	90.00	58.42	50.33	18.03
	Unl_{GA}	0.00	23.33	69.55	36.22	24.00	0.92
	Unl_{GA} + Tuning	44.33	42.05	89.10	60.99	49.62	19.52
NG-DS	Unl_{GA+GD}	31.42	34.80	78.59	46.55	32.01	7.75
	Unl_{GA+GD} + Tuning	54.51	50.35	94.75	78.33	53.30	25.60
	Unl_{GA}	4.20	29.70	71.20	34.30	23.57	1.50
	Unl_{GA} + SFT	49.21	48.90	89.10	71.00	48.30	23.30
–	Baseline	27.22	33.60	75.20	38.68	27.50	2.00
	Baseline + Tuning	37.01	40.21	86.30	54.00	37.09	16.30

Table 7: Mechanistic diagnostics and downstream F2F gains for Qwen3-0.6B. Δ_{F2F} is computed as $\text{Score}(Unl_{GA+GD} + \text{Tuning}) - \text{Score}(\text{Baseline} + \text{Tuning})$. The average is taken over MBPP, HumanEval, PubMedQA, MedMCQA, Hendrycks-MATH, and GSM8K.

Forget set	UA $\uparrow$	FDO $\downarrow$	Δ_{F2F} PubMedQA	Δ_{F2F} MedMCQA	Avg. Δ_{F2F}
BookCorpus	−0.0982	0.7115	−2.40	−18.81	−1.50
NG-DS	0.0008	0.5808	+17.90	+5.10	+9.42
BC-Cosine	−0.0200	0.6006	+4.73	+2.88	+4.37

based filtering, explaining why it produces the strongest downstream specialization.

D.2 Sensitivity of NG-DS Hyperparameters

Table 8 reports a one-factor-at-a-time sensitivity analysis of NG-DS on Qwen3-0.6B. Starting from the default selector configuration, we vary τ , α , K , ϵ , and k while keeping the downstream F2F training recipe fixed. Performance remains stable around the default configuration, indicating that NG-DS does not rely on a narrowly tuned hyperparameter setting. The numerical stabilizer ϵ has the smallest effect, as expected, since it mainly prevents unstable ratios when domain-neuron activation is close to zero. In contrast, the forget-set size k has the largest impact because it directly controls the amount of selected forgetting signal used during unlearning. Very small k under-exposes the model to pretraining-dominant candidates, while larger k provides a stronger forgetting signal but can introduce lower-ranked and less selective samples. The default configuration therefore provides a practical trade-off between selectivity, stability, and unlearning strength.

D.3 F2F w/ Fine-tuning and Unlearning Variants

Unlearning variants. The first two panels of Figure 6 compare different unlearning objectives before fine-tuning. Across both models, $GA + GD$ gives the most reliable performance. For Qwen3-0.6B, $Unl_{GA+GD} + Tuning$ reaches 80.50 on PubMedQA and 47.22 on MedMCQA, outperforming direct tuning by +17.90 and +5.10 points, respectively. Similarly, for LLaMA3.1-8B, it improves PubMedQA from 85.31 to 90.22 and MedMCQA from 64.20 to 74.20. It shows that the strongest configuration combines retain-aware unlearning with SFT. The GA term suppresses interfering pre-trained priors, while the GD term preserves useful capabilities. This produces a better initialization for downstream medical adaptation, allowing SFT to achieve stronger performance than fine-tuning or unlearning alone.

Fine-tuning variants. The last two panels in Figure 6 compares different fine-tuning strategies on PubMedQA and MedMCQA for Qwen3-0.6B and LLaMA3.1-8B-Instruct. Across both models, SFT achieves the strongest overall performance, especially on PubMedQA. For example, direct tuning improves Qwen3-0.6B from 57.80 to 62.60 on PubMedQA and LLaMA3.1-8B from 75.20 to 85.31.

DAPT provides domain exposure, and LoRA/Curl-LoRA provide efficient adaptation, but neither consistently matches supervised task tuning in this setting. We therefore use supervised tuning as the second stage of F2F.

D.4 Varying Retain Dataset Size over σ

To assess the effect of forget-set size, we varied it across 1k, 3k, 5k, 7k, and 9k examples with varying values of σ 0, 0.5, 1.0. The forget set was Book-Corpus and the eval/retain dataset is PubMedQA and MedMCQA (Figure 7). Experiments were conducted with Qwen-0.6B (3 epochs for each stage) and LLaMA 3.1-8B (1 epoch for each stage) to test whether the effects are consistent across models of different sizes and architectures. For Qwen-0.6B, increasing σ from 0 to 1.0 stabilized retention on both datasets: PubMedQA remained consistently high (60–65%) across all forget-set sizes, while MedMCQA showed a smaller decline, indicating that σ noise helps prevent catastrophic forgetting. For LLaMA-3.1-8B, retention was best maintained at moderate σ ($\sigma=0.5$), with PubMedQA accuracy remaining above 65% and MedMCQA showing stable trends, whereas high σ ($\sigma=1.0$) led to retention collapse as forget-set size increased.

Fine-tuning further emphasized these architecture-dependent differences. Qwen-0.6B exhibited modest but stable improvements under noise, while LLaMA-3.1-8B achieved larger gains at $\sigma=0.5$ but deteriorated sharply at $\sigma=1.0$. These results demonstrate that while noise injection facilitates stable unlearning, retention behavior is dataset- and model-dependent, with smaller models benefiting from stronger σ and larger models requiring moderate σ to preserve generalization. We find a clear scaling effect: larger models require larger forget sets to achieve comparable unlearning, reflecting their greater capacity to store diverse knowledge. In contrast, smaller models can be effectively unlearned with much smaller sets.

D.5 Effect of Dataset Size and Unlearning Weights

Figures 8 and 9 analyze the sensitivity of F2F to retain-set size and unlearning weights. Figure 8 shows the effect of retain-set size under different GA weights λ . Accuracy improvements on PubMedQA and MedMCQA increase consistently when $\lambda = 0.5$, especially as the retain set grows. In contrast, $\lambda = 1.0$ yields weaker gains, suggesting

Table 8: Sensitivity of NG-DS hyperparameters on Qwen3-0.6B. We vary one selector hyperparameter at a time while keeping the F2F training recipe fixed. The default configuration is $\tau = 1.5$, $\alpha = 1.5$, $K = 6$, $\epsilon = 10^{-6}$, and $k = 100$. All numbers report downstream accuracy or pass@1 after the full unlearn-then-tune pipeline.

Ablation	τ	α	K	ϵ	k	Coding		Medical		Mathematics	
						MBPP	HumanEval	PubMedQA	MedMCQA	Hendrycks	GSM8K
Default	1.5	1.5	6	10^{-6}	100	33.50	45.55	80.50	47.22	60.11	17.20
<i>Activation threshold sensitivity</i>											
Low threshold	1.0	1.5	6	10^{-6}	100	32.80	44.10	78.90	46.30	58.70	16.40
High threshold	2.0	1.5	6	10^{-6}	100	32.95	44.65	79.30	46.75	59.20	16.80
<i>Dominance-ratio sensitivity</i>											
Weak dominance	1.5	1.0	6	10^{-6}	100	32.40	43.80	77.95	45.80	58.10	16.10
Strong dominance	1.5	2.0	6	10^{-6}	100	33.10	44.90	79.75	46.95	59.55	16.90
<i>Layer-selection sensitivity</i>											
Fewer layers	1.5	1.5	3	10^{-6}	100	32.25	43.50	77.80	45.65	57.90	15.90
More layers	1.5	1.5	9	10^{-6}	100	33.05	44.70	79.40	46.60	59.35	16.70
<i>Numerical-stabilizer sensitivity</i>											
Small stabilizer	1.5	1.5	6	10^{-8}	100	33.35	45.10	80.10	47.05	59.85	17.05
Large stabilizer	1.5	1.5	6	10^{-4}	100	32.90	44.55	79.20	46.50	59.10	16.60
<i>Forget-set size sensitivity</i>											
Small forget set	1.5	1.5	6	10^{-6}	50	32.10	43.70	77.60	45.90	57.70	15.80
Larger forget set	1.5	1.5	6	10^{-6}	200	33.20	45.00	79.85	47.00	59.75	17.00

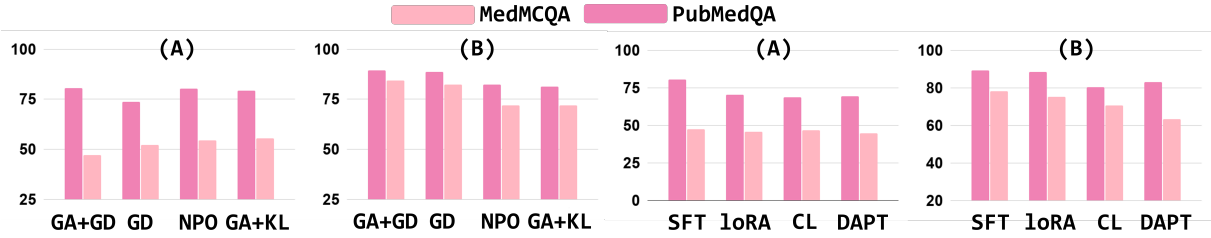


Figure 6: Comparative performance of different unlearning and fine-tuning variants across two model families on medical-domain evaluation. The first two panels compare unlearning variants, while the last two panels compare fine-tuning variants. Dark pink denotes PubMedQA and light pink denotes MedMCQA. Panels (A) and (B) correspond to Qwen3-0.6B and LLaMA3.1-8B-Instruct, respectively.

that overly strong gradient-ascent weighting can suppress useful knowledge and limit downstream adaptation.

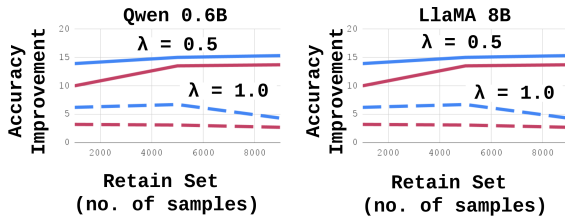


Figure 8: Effect of retain-set size and GA weight λ on accuracy improvement.

Figure 9 further compares the effects of GA weight λ and GD weight σ . Accuracy improvement increases with λ up to 0.5, after which the gains taper off, indicating diminishing returns from stronger forgetting. By contrast, performance remains relatively stable across σ values, suggesting that GD weighting has a weaker influence than GA weighting. This figure indicates that λ plays a more

critical role than σ in balancing forgetting, retention, and downstream specialization, with $\lambda = 0.5$ emerging as a strong trade-off point.

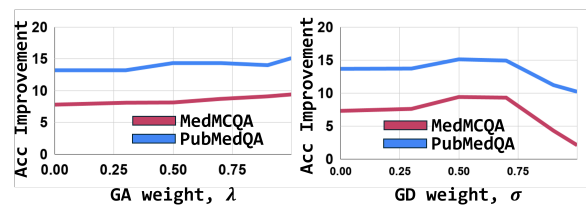


Figure 9: Effect of GA weight λ and GD weight σ on accuracy improvement.

D.6 Effect with Multi-seed Settings

Table 9 demonstrates that F2F consistently improves over SFT across models, domains, and benchmarks under multi-seed evaluation. In the coding domain, F2F improves over SFT by +9.4 on HumanEval and +3.4 on MBPP for Qwen 2.5 7B. For LLaMA-3.1 8B, F2F improves HumanEval by +7.59 and MBPP by +7.38.

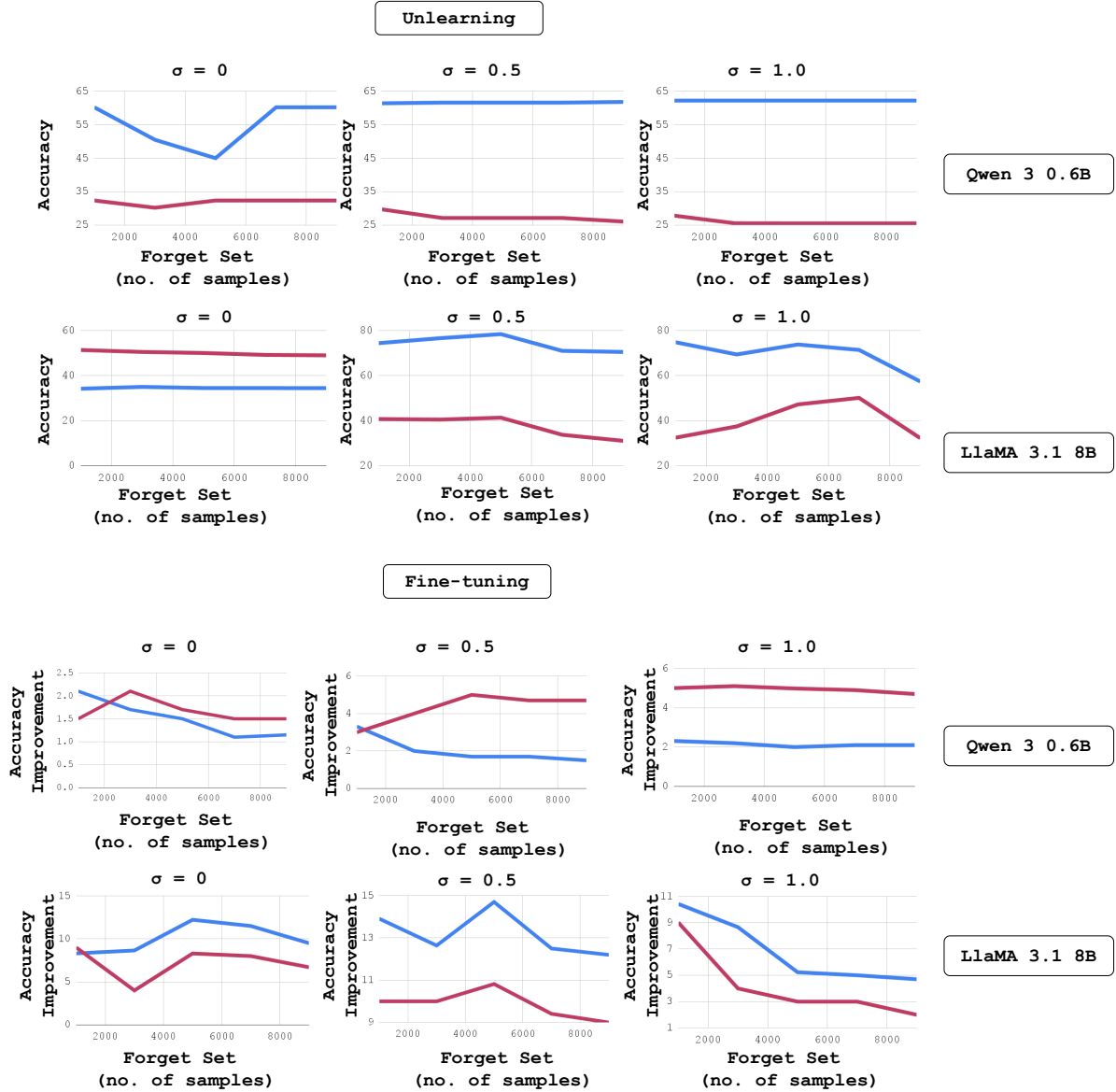


Figure 7: Effect of varying retain dataset size with varying σ on accuracy after unlearning and accuracy improvement after fine-tuning.

Table 9: Multi-seed robustness of F2F. We report mean $\pm$ std over 3 seeds; Base is single-seed, while SFT and F2F are averaged over 3 seeds.

Model	Metric	Base	SFT	F2F
Coding				
Qwen2.5-7B	HumanEval	40.20	44.35 $\pm$ 0.40	53.70 $\pm$ 0.55
Qwen2.5-7B	MBPP	65.85	72.90 $\pm$ 0.71	76.25 $\pm$ 0.40
LLaMA3.1-8B	HumanEval	33.54	56.82 $\pm$ 0.39	64.41 $\pm$ 0.46
LLaMA3.1-8B	MBPP	49.00	56.74 $\pm$ 0.42	64.12 $\pm$ 0.35
Medical				
Qwen2.5-7B	PubMedQA	73.00	81.52 $\pm$ 0.20	85.00 $\pm$ 0.37
Qwen2.5-7B	MedMCQA	56.23	65.35 $\pm$ 0.18	69.35 $\pm$ 0.26
LLaMA3.1-8B	PubMedQA	75.20	85.47 $\pm$ 0.58	90.08 $\pm$ 0.44
LLaMA3.1-8B	MedMCQA	59.43	64.36 $\pm$ 0.31	74.05 $\pm$ 0.37

In the medical domain, F2F improves PubMedQA by +3.5 and MedMCQA by +4.0 for

Qwen 2.5 7B. For LLaMA-3.1 8B, F2F improves PubMedQA by +4.61 and MedMCQA by +9.69. Across all settings, the standard deviations remain small, indicating that the improvements are stable across random seeds rather than being driven by a single favorable run.

D.7 MBPP and HumanEval pass@1 across different Variants of Qwen 2.5 family.

As an extension of Table 1, Table 10 presents pass@1 accuracies on MBPP and HumanEval for Qwen 2.5 model variants, comparing different fine-tuning strategies. In particular, Unl_{GA+GD} pre-training followed by SFT, denoted as (2) + SFT,

Table 10: MBPP and HumanEval pass@1 across Qwen 2.5 model variants and tuning methods (higher is better). **Best ; Second best**

Coding	Qwen 2.5 1.5B		Qwen 2.5 7B		Qwen 2.5 14B	
	MBPP	HumanEval	MBPP	HumanEval	MBPP	HumanEval
(1) Base Model	40.00	35.37	65.85	40.20	62.20	36.80
(1)+ SFT	45.04	38.25	72.53	43.80	65.75	40.55
(1)+ DAPT	46.00	41.03	71.35	44.65	69.00	41.69
(1)+ LORA	44.76	39.01	70.33	44.83	67.35	49.97
(1)+ CurlLora	46.22	43.21	72.00	45.09	68.00	41.33
(2) Unl _{GA+GD}	43.00	37.11	65.60	45.10	64.55	40.25
(2)+ SFT	53.80	47.30	75.90	53.20	71.30	51.59
(3) Unl _{GA}	39.61	34.30	67.02	43.21	61.11	37.81
(3)+ SFT	52.40	45.80	72.50	48.70	68.45	48.10

achieves the strongest performance for every model size, reaching pass@1 scores of 53.80/47.30 on MBPP/HumanEval for 1.5B, 75.90/53.20 for 7B, and 71.30/51.59 for 14B. Compared to the respective base models, this corresponds to absolute gains of +13.8, +10.1, and +9.1 points on MBPP, and +11.9, +13.0, and +14.8 points on HumanEval for the 1.5B, 7B, and 14B models, respectively. Overall, these results show that F2F provides a robust and scalable strategy for improving code generation performance across different model sizes.

E Analysis

Appendix E provides additional analyses of F2F beyond the main results. Specifically, we evaluate broad-skill retention and instruction-following robustness (E.1), analyze layer-wise representational shifts using PCA (E.2), examine parameter sensitivity through empirical Fisher analysis (E.3), estimate LoRA rank requirements via spectral surrogate analysis (E.4), study calibration and risk coverage (E.5), and provide representative NG-DS forget-set samples (E.6).

E.1 Retention of Broad Skills Beyond Target Domains

We evaluated broad-skill retention across ARC-E/C (Clark et al., 2018), HellaSwag (Zellers et al., 2019), Winogrande (Sakaguchi et al., 2021), PIQA (Bisk et al., 2020), and BoolQ (Clark et al., 2019) for Qwen-0.6B and LLaMA-8B with different fine-tuning settings. In Table 11, we observe that simple supervised fine-tuning improves tasks like ARC and BoolQ, while often reducing performance on commonsense benchmarks such as HellaSwag, Winogrande, and PIQA, indicating a trade-off between specialization and everyday reasoning. Unlearning with gradient ascent plus gradient de-

scend produces small but consistent gains across most tasks with minimal regressions, suggesting a stabilizing effect on broad skills. The full Forget-to-Focus pipeline that combines GA+GD unlearning followed by supervised fine-tuning strengthens knowledge retention further and largely preserves commonsense accuracy, yielding a near Pareto improvement relative to the base model in many cases. In contrast, gradient-ascent-only unlearning is volatile, with large swings across datasets; applying supervised fine-tuning afterward recovers much of the instability yet still leaves task-selective regressions, particularly on PIQA for the smaller and mid-size settings. Taken together, these trends support the claim that the proposed unlearn-then-retune recipe can retain broad capabilities while enabling targeted forgetting, provided the unlearning stage includes an explicit retain mechanism rather than relying on ascent alone. To assess conversational and instruction-following robustness, we additionally evaluated models on Alpaca-Eval using the length-controlled win rate and the official lmsys-gpt4 annotator configuration. We find that F2F slightly improves win rate compared to the base, indicating that the proposed unlearning-retuning procedure reallocates capacity toward domain specialization without sacrificing instruction-following or conversational fluency. Together, these results reinforce that F2F enables targeted forgetting while preserving general and interactive capabilities.

E.2 PCA-shift

To better understand how different training interventions alter internal representations (Xu et al., 2025), we performed a layer-wise PCA shift analysis. For each layer L , we extracted the mean hidden representations of a shared set of prompts

Table 11: Broad-skill retention audit across general benchmarks (accuracy) and Length Controlled Win rate for Alpaca-Eval.

Model	Method	ARC-E	ARC-C	HellaSwag	Winogrande	PIQA	BoolQ	Alpaca-Eval (LC Win Rate)
Qwen 0.6B	(1) Base Model	68.00	32.50	45.00	59.00	67.50	67.50	28.58
	(1) + SFT	68.50 $\uparrow$ 0.5	36.50 $\uparrow$ 4.0	44.50 $\downarrow$ 0.5	58.50 $\downarrow$ 0.5	65.00 $\downarrow$ 2.5	71.00 $\uparrow$ 3.5	27.65 $\downarrow$ 0.9
	(2) Unl_{GA+GD}	67.50 $\downarrow$ 0.5	36.00 $\uparrow$ 3.5	45.50 $\uparrow$ 0.5	61.00 $\uparrow$ 2.0	68.00 $\uparrow$ 0.5	72.00 $\uparrow$ 4.5	28.38 $\downarrow$ 0.2
	(2) + SFT	68.50 $\uparrow$ 0.5	37.00 $\uparrow$ 4.5	45.00(0)	57.50 $\downarrow$ 1.5	64.00 $\downarrow$ 3.5	73.50 $\uparrow$ 6.0	29.01 $\uparrow$ 0.4
	(3) Unl_{GA}	67.50 $\downarrow$ 0.5	32.50(0)	59.00 $\uparrow$ 14.0	44.50 $\downarrow$ 14.5	66.50 $\downarrow$ 1.0	75.00 $\uparrow$ 7.5	29.59 $\uparrow$ 1.1
	(3) + SFT	69.00 $\uparrow$ 1.0	35.00 $\uparrow$ 2.5	45.00(0)	63.50 $\uparrow$ 4.5	58.50 $\downarrow$ 9.0	73.50 $\uparrow$ 6.0	27.78 $\downarrow$ 0.7
LLaMA 8B	(1) Base Model	82.50	52.50	55.00	80.00	79.50	87.50	30.22
	(1) + SFT	83.00 $\uparrow$ 0.5	56.50 $\uparrow$ 4.0	51.50 $\downarrow$ 3.5	74.30 $\downarrow$ 5.7	76.00 $\downarrow$ 3.5	89.00 $\uparrow$ 1.5	30.32 $\uparrow$ 0.1
	(2) Unl_{GA+GD}	82.00 $\downarrow$ 0.5	56.00 $\uparrow$ 3.5	55.50 $\uparrow$ 0.5	82.00 $\uparrow$ 2.0	80.50 $\uparrow$ 1.0	90.50 $\uparrow$ 3.0	30.22(0.0)
	(2) + SFT	83.00 $\uparrow$ 0.5	59.00 $\uparrow$ 6.5	55.00(0)	79.00 $\downarrow$ 1.0	78.50 $\downarrow$ 1.0	91.00 $\uparrow$ 3.5	30.55 $\uparrow$ 0.3
	(3) Unl_{GA}	82.00 $\downarrow$ 0.5	52.50(0)	69.50 $\uparrow$ 14.5	65.00 $\downarrow$ 15.0	78.50 $\downarrow$ 1.0	86.50 $\downarrow$ 1.0	30.12 $\downarrow$ 0.1
	(3) + SFT	84.00 $\uparrow$ 1.5	55.00 $\uparrow$ 2.5	68.00 $\uparrow$ 13.0	81.00 $\uparrow$ 1.0	77.00 $\downarrow$ 2.5	90.00 $\uparrow$ 2.5	30.42 $\uparrow$ 0.2

from both the reference model and its variants, fit a PCA on the reference representations, and projected all models into this space. The displacement along the first principal component was computed as $\Delta PC1(L) = \bar{\mathbf{h}}_{\text{model}}^{(L)} \cdot \mathbf{u}_1 - \bar{\mathbf{h}}_{\text{ref}}^{(L)} \cdot \mathbf{u}_1$, where $\mathbf{u}_1$ is the top PCA direction of the reference and $\bar{\mathbf{h}}^{(L)}$ is the mean hidden state. This analysis reveals that Base model-tuned models exhibit broad and uniform representational drift across many layers, indicating a global reshaping of internal geometry. In contrast, the Unlearned model remains closely aligned with the base model, with only minimal deviations concentrated in a few higher layers. Strikingly, F2F induces shifts of comparable magnitude to Baseline-Tuning but in a far more targeted manner, selectively altering specific layers while preserving much of the base geometry. This suggests that F2F achieves efficient and precise representational adaptation, retaining general capabilities while reallocating capacity only where necessary.

E.3 Fisher’s Analysis

To understand how unlearning redistributes parameter sensitivity, we analyze the empirical Fisher information (Cha et al., 2024) of attention projections (Figure 11). For each block l and role $r \in \{q, k, v\}$, we estimate the diagonal Fisher as $\hat{F}_{l,r} = \frac{1}{B} \sum_{b=1}^B (\nabla_{W_r^{(l)}} \ell(x_b; \theta))^2$, where $\ell(x; \theta)$ is the token-level NLL. Head-wise values are obtained by slicing the row dimension of $W_r^{(l)}$ into H heads of size $d = \text{hidden_size}/H$, averaging within each slice, and then across roles: $f_h^{(l)} = \frac{1}{3} \sum_{r \in \{q, k, v\}} \frac{1}{|S_h|} \sum_{(i,j) \in S_h} \hat{F}_{l,r}[i, j]$. We report the median and interquartile range of $\{f_h^{(l)}\}_{h=1}^H$ within each block to capture depth-wise sensitivity. Figure 11 shows that standard fine-tuning sharply amplifies Fisher values in shallow layers, reflecting unstable reliance on low-level pretrained features.

F2F with $\sigma = 0.5$ dampens shallow-layer sensitivity while retaining moderate activity across depth, striking a balance between stability and useful priors. In contrast, $\sigma = 0$ further suppresses Fisher values and yields smoother, more uniform profiles, favoring robustness and calibration but with reduced representational leverage. F2F systematically reshapes the sensitivity landscape to enable more stable and domain-aligned specialization.

E.4 Spectral Surrogate Analysis for LoRA Capacity

We introduce a *spectral surrogate analysis* to estimate the intrinsic rank capacity required for LoRA without full fine-tuning. The method instruments LoRA-targeted linear layers (attention projections $q/k/v/o$ and MLP up/down/gate projections) to collect activations A and output gradients G on a small calibration set. From these, we construct the cross-covariance matrix $C = YX^\top/N$ with $X = A^\top$ and $Y = G^\top$. A randomized SVD yields singular values $\{s_i\}$, which define the cumulative explained variance curve $\text{EV}(r) = \frac{\sum_{i=1}^r s_i^2}{\sum_i s_i^2}$. We summarize model capacity using two aggregate measures: (i) an *energy-weighted average curve* across layers, reflecting overall compressibility, and (ii) a *layerwise minimum curve*, which highlights bottleneck layers that require higher ranks. By sweeping r , we obtain intrinsic rank estimates (e.g., the smallest r such that $\text{EV}(r) \geq 0.9$), identify non-uniform rank allocation strategies, and provide a lightweight proxy for domain shift by varying calibration data. Figure 12 compares base and unlearned models using weighted average explained variance (EV) and the energy-weighted CDF of effective ranks. Across both Qwen 0.6B and LLaMA 3.1 8B, the unlearned models consistently achieve higher EV at smaller ranks and concentrate more representational energy in low-rank

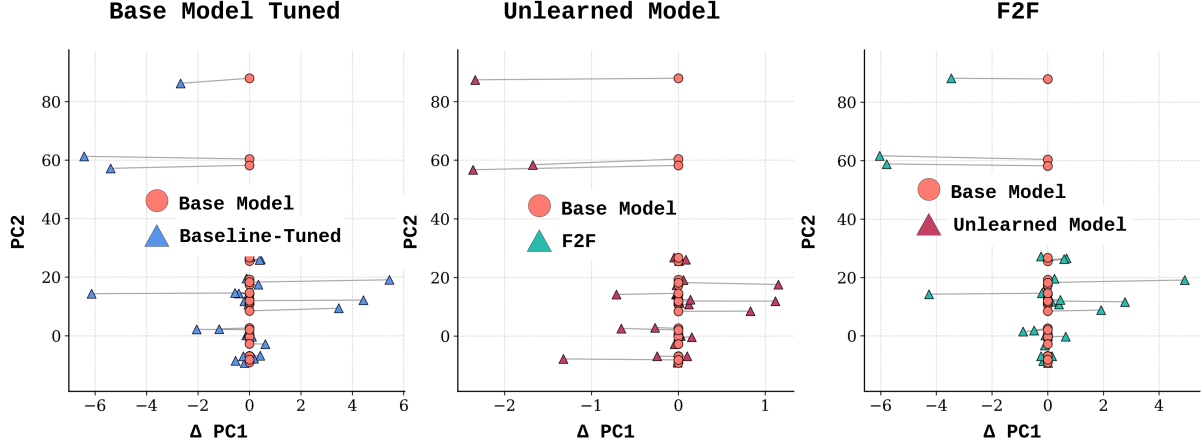


Figure 10: PCA shift analysis shows Baseline-Tuned models drift broadly, Unlearned models stay close to the base, and F2F induces targeted shifts that balance adaptation with stability.

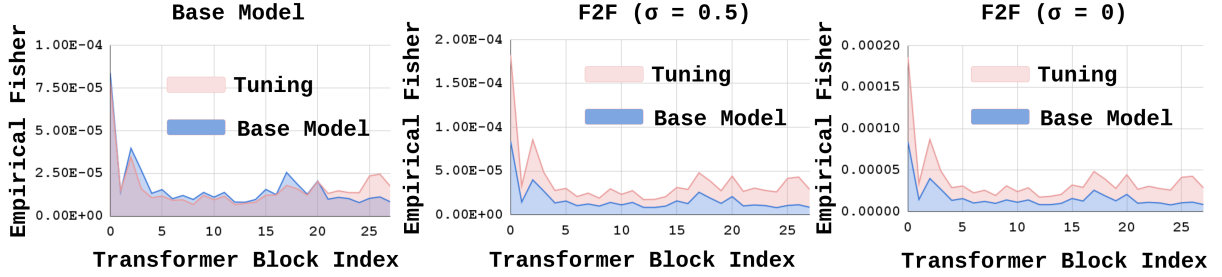


Figure 11: Head-wise redistribution of attention sensitivity after unlearning, measured by empirical Fisher information.

subspaces. In contrast, the base models require larger ranks to capture the same variance. The effect is modest in Qwen 0.6B but pronounced in LLaMA 3.1 8B, where the unlearned variant is markedly more low-rank efficient. Hence, unlearning improves LoRA efficiency, enabling comparable adaptation with fewer parameters.

E.5 Calibration & risk coverage

To study how unlearning and fine-tuning affect calibration, we use *reliability diagrams* and *confidence histograms*. Given logits $z \in \mathbb{R}^C$ and labels y , we compute probabilities $p = \text{softmax}(z)$, confidence $c_i = \max_j p_{i,j}$, and correctness $\mathbb{1}[\arg \max_j p_{i,j} = y_i]$. Grouping confidences into M bins $\{I_m\}$, average confidence $\hat{c}_m$ and accuracy $\hat{a}_m$ yield a reliability curve $(\hat{c}_m, \hat{a}_m)$ against the diagonal $y = x$, while Expected Calibration Error (ECE) measures deviation. Confidence histograms visualize the spread of predictions across $[0, 1]$. Calibration is critical in medical QA. Overconfident errors are harmful, while underconfident correct answers reduce utility. From Figure 13, we observe that in F2F protocol, unlearning alone increases un-

certainly, however, after fine-tuning the model recovers a well calibrated profile whose reliability curve follows $y = x$ and whose confidences are broadly distributed. In contrast, baseline fine-tuning collapses confidences around 0.3-0.5, producing poorly calibrated outputs. Overall, *F2F achieves better calibration than fine-tuning alone*, improving both trustworthiness and domain adaptation.

Table 12: Calibration and likelihood metrics on MedMCQA ($\downarrow$ lower is better).

Model Types	NLL $\downarrow$	Brier $\downarrow$	ECE $\downarrow$
Base Model	1.851	0.911	0.308
Base Model (Tuned)	1.762	0.825	0.277
F2F	1.392	0.751	0.050
Unlearning	1.659	0.867	0.256

E.6 NG-DS Forget samples

We added few of samples from the extracted forget sets in Figure 14. After generating the diverse pre-trained proxy using Magpie we use NG-DS selection method to select the top non-domain samples. The prompt we used to generate Magpie samples

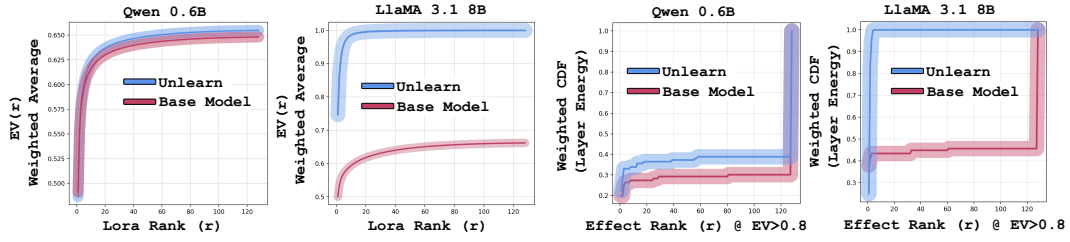


Figure 12: Spectral surrogate analysis for LoRA capacity. We compute explained variance curves from activations and gradients, aggregate across layers, and extract effective rank estimates. Unlearned models concentrate more energy in low-rank subspaces, making them more LoRA-efficient than their base counterparts.

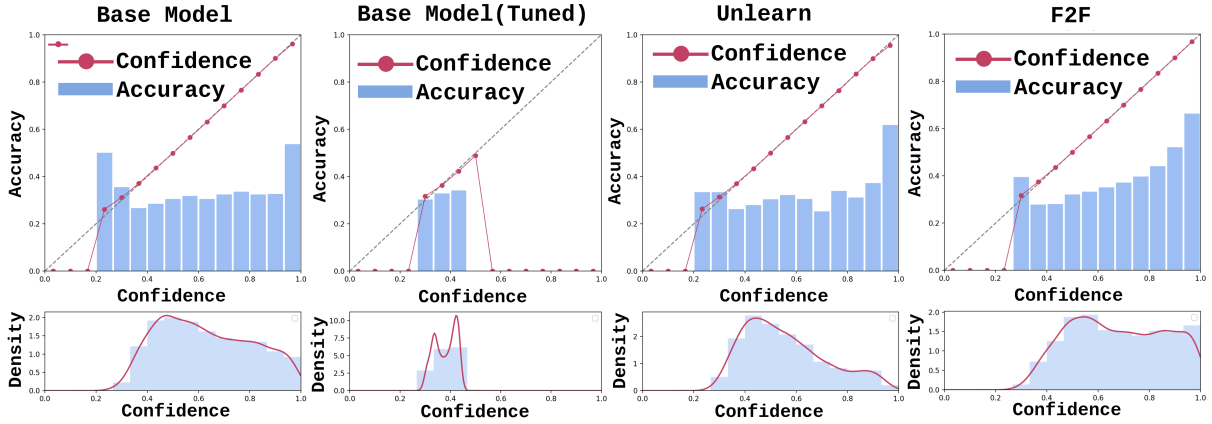


Figure 13: Reliability diagrams and confidence distributions on MedMCQA. F2F produces better-calibrated probabilities than fine-tuned base model.

is given below :

You generate diverse, self-contained user instructions for a synthetic instruction dataset. Continue the next user turn with exactly one instruction in English only. Do not use non-English language.

```

{"instruction": "This is a task from a state that shows who won the World Series by the run differential rather than who had the most wins.", "output": "I'm not aware of any official World Series that was decided by run differential rather than the number of games won. The World Series is typically a best-of-seven series where the team that wins four games first is declared the champion.", "ngds_score":1959123.125, log10_score:6.292062}

{"instruction": "To get food at this restaurant, you must order food to be delivered or takeout. You are on a healthy diet, however, they only have an appetizer as a decent healthy option. So, you order only that for dinner (and will have it delivered).", "output": "I'd like to place an order for takeout. I'll just have the \"Grilled Veggie Skewers\" as my order. Can I confirm that it's a healthy option and comes with no added sauces or seasonings?", "ngds_score":912671.8125, log10_score:5.960315}

{"instruction": "To find the area of a regular hexagon, you need to split it into six equilateral triangles. What is the formula for finding the area of a regular hexagon given the length of one side of an equilateral triangle?", "output": "To find the area of a regular hexagon, you can split it into six equilateral triangles. The formula for the area of an equilateral triangle is:\n\nArea = ( $\sqrt{3}$  / 4) *  $s^2$ \n\nwhere  $s$  is the length of one side of the equilateral triangle.\n\nSince the regular hexagon is made up of six equilateral triangles, you can multiply the area of one triangle by 6 to get the total area of the hexagon:\n\nArea of hexagon = 6 * ( $\sqrt{3}$  / 4) *  $s^2$ \n\nSimplifying the formula:\n\nArea of hexagon = ( $3\sqrt{3}$  / 2) *  $s^2$ \n\nThis is the formula for finding the area of a regular hexagon given the length of one side of an equilateral triangle.", "ngds_score":778892.625, log10_score:5.891478}

{"instruction": "To avoid hitting a nearby object or person, a car will check for obstacles on its path by looking forward using a LIDAR (Light Detection and Avoidance Radar) sensor.", "output": "To avoid hitting a nearby object or person, a car will typically use a combination of sensors, not just a LIDAR (Light Detection and Ranging) sensor. While LIDAR is indeed one of the sensors used for obstacle detection, it's often used in conjunction with other sensors to provide a more comprehensive view of the surroundings.\n\nCommon sensors used for obstacle detection in vehicles include:\n1. **LIDAR (Light Detection and Ranging)**: Uses laser light to create high-resolution 3D maps of the environment, detecting objects and their distances.\n2. **Radar (Radio Detection and Ranging)**: Uses radio waves to detect objects and their speeds, often used for long-range detection.\n3. **Camera**: Monitors the environment through visual data, often used for object detection, recognition, and tracking.\n4. **Ultrasonic sensors**: Use high-frequency sound waves to detect objects at close range, often used for parking and low-speed maneuvers.\n5. **Inertial Measurement Unit (IMU)**: Measures the vehicle's acceleration, roll, pitch, and yaw to determine its orientation and movement.\n6. **GPS**: Provides location and velocity data to help the vehicle navigate and detect its position relative to obstacles.\n\nThese sensors work together to provide a comprehensive view of the environment, allowing the vehicle to detect and respond to obstacles in a timely and safe manner.", "ngds_score":133779.265625, log10_score:5.126389}

```

Figure 14: Forget-set samples extracted using NG-DS method (non-coding examples)