FINE-TUNING BEYOND WEIGHT-SPACE GEOMETRY: ONE FRONTIER, TWO ORIGINS

Anonymous authors
Paper under double-blind review

ABSTRACT

Post-training methods are commonly evaluated through the tradeoff between new-task performance and retention, yet the shape of this frontier does not reveal its cause. A similar loss of adaptation can be either necessary, because the old and new tasks demand incompatible predictions, or avoidable, because an old-data-free objective constrains the wrong computations. We expose this ambiguity using controlled causal sequence tasks with known target distributions and replay solutions. We study WindowLoRA, a family that moves from ordinary LoRA to dense teacher distillation by progressively increasing the causal and positional coverage of hidden-state matching. Across both conflicting and compatible tasks, increasing constraint density produces the same qualitative continuum: old-task error decreases while new-task error increases. The meaning of this continuum, however, changes relative to replay. Under shared-input target conflict, dense output distillation reaches the replay compromise, showing that the remaining tradeoff is intrinsic. Under compatible disjoint support, replay solves both tasks while new-input matching remains on a nonzero frontier, showing that the tradeoff is introduced by incomplete preservation supervision. We then test this interpretation in pretrained decoder-only models by comparing the representation drift and downstream retention–accuracy frontiers induced by different adaptation methods. Although stronger constraints reduce drift, none eliminates the frontier, and WindowLoRA provides controlled intermediate operating points. Finally we position the family to K-prior which explains how a new-input matching can replace the unavailable old objective only when its probes cover the old-sensitive directions of the adaptation carrier.

1 INTRODUCTION

Large language models (LLMs) acquire broad capabilities through pretraining on large text collections, but must subsequently be adapted through post-training to follow instructions, specialize to new domains, or express human preferences (Qwen Team, 2025; Ouyang et al., 2022). This post-training stage introduces a central tension in which learning new behavior can degrade capabilities encoded by the pretraining stage (Luo et al., 2023; Li et al., 2024). Such degradation is often summarized as a retention-adaptation tradeoff, but the tradeoff is not self-explanatory. It may be unavoidable because the old and new tasks require incompatible predictions for the same input, or it may be induced by a preservation objective that only imperfectly substitutes for unavailable old data. Distinguishing these cases is essential, where in the former, even joint training with complete replay must compromise, and in the latter case, a common solution of both tasks may exist, but the preservation method itself prevents the model from reaching it.

Prior works approach this tension through different preservation mechanisms: parameter-efficient methods restrict the space of weight updates (Hu et al., 2022; Meng et al., 2024; Wang et al., 2025), while distillation and representation-based methods encourage the adapted model to remain close to a frozen source model (Gao et al., 2026; Dalal et al., 2026). These methods are typically compared through paired measurements of new-task learning and old-task retention. Such comparisons show that different methods occupy different points on the tradeoff, but they do not explain the source of it. In natural tasks, the conditional targets and the best jointly attainable solution are generally unknown; an adaptation cost may therefore reflect an intrinsic disagreement between the tasks, or a

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

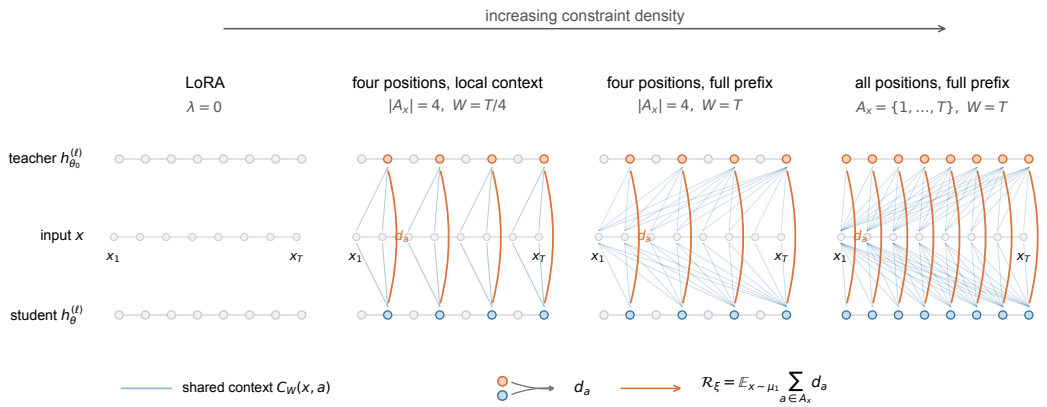


Figure 1: The WINDOWLORA family as increasing matching coverage. Orange edges impose teacher–student matching at anchors $a \in A_x$, while blue edges show the causal context $C_W(x, a)$ used to compute each matched state. Increasing $|A_x|$ adds positional constraints, and increasing W exposes each constraint to more of the prefix.

limitation of the preservation mechanism used to approximate the missing old-task objective. This ambiguity makes it difficult to observe whether the observed trade-off is inevitable or it could be reduced by adjusting the preservation mechanism.

To make this ambiguity observable, we construct finite sequence-prediction problems in which the input domain and conditional targets are known exactly for both the pre-training and post-training tasks. We compare a continuum of adaptation objectives with ordinary LoRA is assumed to achieve "ideal" adaptation performance and LoRA with dense teacher distillation is assumed to achieve "ideal" retention performance, with sparse local-window and full-context matching located as intermediate states. When the two tasks have disjoint input supports, complete replay achieves low risk on both, showing that no intrinsic tradeoff is required. Nevertheless, moving from LoRA toward denser distillation systematically improves retention at an increasing adaptation cost; this tradeoff is introduced by the preservation objective and its incomplete coverage of the old task. When the tasks instead assign conflicting targets to the same inputs, even complete replay must compromise, revealing an intrinsic component that any adaptation method cannot remove.

We interpret this continuum through knowledge-adaptation priors (Khan & Swaroop, 2021). A full K-prior reconstructs the missing old-data gradient using matched function- and weight-space divergences, whereas old-data-free teacher matching retains only a partial functional constraint evaluated on adaptation inputs. This view yields a family of partial K-priors that differ in the context available to each constraint and the sequence positions they cover. All-position full-context distillation is one dense member of this family, sparse full-context and local-window matching are progressively cheaper relaxations, and ordinary LoRA is recovered when the functional constraint is removed.

Having isolated these two sources of adaptation cost in our controlled toy tasks, we then examine how these tradeoffs manifest in real tasks. We evaluate Qwen2.5-1.5B across seven commonsense tasks and nine adaptation methods, and transfer the preservation family to Llama-3.1-8B. None of the evaluated method dominates: stronger adaptation generally produces greater functional drift, while stronger teacher constraints improve retention at an adaptation cost. Sparse local matching provides intermediate operating points with lower cost than dense distillation. Because task compatibility is unknown in these natural settings, we interpret these results as empirical frontiers without claiming whether they arise from either of the aforementioned sources.

Our contributions are:

- A controlled evaluation separating intrinsic target conflict from method-induced forgetting due to incomplete preservation supervision.
- Evidence that constraint density traces similar empirical frontiers in both settings, although replay reveals that only the conflict frontier is intrinsic.

- A WindowLoRA family connecting LoRA to dense distillation, with a K-prior analysis relating preservation to coverage of old-sensitive adapter directions.
- Validation on pretrained decoders through activation-drift and adaptation–retention comparisons across post-training methods.

2 Analysis of Retention and Adaptation

Existing post-training benchmarks reveal whether a method forgets, but generally not why. Their old and new conditional distributions are unknown, so an observed retention–adaptation tradeoff admits at least two explanations. The tasks may genuinely demand different predictions for the same input, in which case even joint retraining must compromise. Alternatively, a predictor that solves both tasks may exist, while the adaptation method fails to find or preserve it because the old supervision is unavailable. Distinguishing these cases is necessary before attributing forgetting to an adapter, a regularizer, or the underlying task relationship. We therefore begin with a controlled sequence setting in which the old and new input distributions, their conditional targets, and the replay solution are known.

2.1 Two controlled sources of forgetting

Let μ_0 and μ_1 denote the old and new input distributions, and let $q_0(\cdot \mid x)$ and $q_1(\cdot \mid x)$ denote their conditional targets. For an unrestricted predictor r, the population objective corresponding to joint training is

$$\mathcal{J}_\gamma(r) = \gamma\, \mathbb{E}_{x\sim\mu_0} D_{\mathrm{KL}}(q_0(\cdot \mid x) \,\|\, r(\cdot \mid x)) + \mathbb{E}_{x\sim\mu_1} D_{\mathrm{KL}}(q_1(\cdot \mid x) \,\|\, r(\cdot \mid x)), \tag{1}$$

where γ controls the relative old-task weight. At every input with nonzero total mass, its pointwise minimizer is

$$r^*_\gamma(\cdot \mid x) = \frac{\gamma\mu_0(x) q_0(\cdot \mid x) + \mu_1(x) q_1(\cdot \mid x)}{\gamma\mu_0(x) + \mu_1(x)}. \tag{2}$$

Equation 2 provides the reference needed to interpret adaptation. When the two tasks place mass on the same input but assign it different targets, the optimum is a mixture and neither task can generally attain zero KL. The resulting tradeoff is intrinsic: it remains even for an unrestricted predictor with complete replay. If the supports are disjoint and the target rules are compatible, both risks can instead be minimized simultaneously. Forgetting in this case is not required by the prediction problem.

We instantiate these alternatives using binary strings $x \in \{0,1\}^T$. At every causal prefix $x_{\le a}$, a fixed oracle supplies a four-class soft target $\rho(\cdot \mid x_{\le a}) \in \Delta^4$. The oracle remains fixed throughout the study. We construct two main relationships between the old and new tasks.

Conflicting targets. The old and new tasks have the same input distribution, $\mu_0 = \mu_1$, but receive opposing perturbations of the oracle target:

$$q_0(\cdot \mid x_{\le a}) = \rho(\cdot \mid x_{\le a}) - \delta(x_{\le a}), \qquad q_1(\cdot \mid x_{\le a}) = \rho(\cdot \mid x_{\le a}) + \delta(x_{\le a}). \tag{3}$$

The perturbation satisfies $\mathbf{1}^\top \delta(x_{\le a}) = 0$ and is bounded so that both targets remain valid distributions. Thus, every supervised input is shared, but the desired predictions differ.

We use two versions of this construction to vary the structure of the conflict. In the *global-conflict* condition, the perturbation is determined by the first token and therefore remains coupled to a global sequence feature. In the *local-conflict* condition, it depends only on the most recent tokens and the absolute prediction position.

Compatible disjoint supports. The old distribution contains strings beginning with 0, while the new distribution contains strings beginning with 1:

$$\mu_0 = \mathrm{Unif}\left(\{0\} \times \{0,1\}^{T-1}\right), \qquad \mu_1 = \mathrm{Unif}\left(\{1\} \times \{0,1\}^{T-1}\right). \tag{4}$$

Both tasks use the same oracle rule,

$$q_0(\cdot \mid x_{\le a}) = q_1(\cdot \mid x_{\le a}) = \rho(\cdot \mid x_{\le a}) \tag{5}$$

on their respective supports. A sufficiently expressive predictor can therefore solve both simultaneously. Table 3 gives concrete length-eight examples.

The comparison between conflict and disjoint support is the main diagnostic contrast.

2.2 TEACHER–STUDENT ADAPTATION PROTOCOL

For each controlled task pair, we first train a source model θ_0 on the old distribution:

$$\theta_0 \in \arg\min_{\theta} \mathcal{L}_0(\theta), \qquad \mathcal{L}_0(\theta) = \mathbb{E}_{x\sim\mu_0} \frac{1}{|\mathcal{P}(x)|} \sum_{a\in\mathcal{P}(x)} D_{\mathrm{KL}}(q_0(\cdot \mid x_{\le a}) \,\|\, p_\theta(\cdot \mid x_{\le a})), \tag{6}$$

where $\mathcal{P}(x) = \{1, \dots, T\}$. A frozen copy of θ_0 serves as the teacher. The student is initialized from the same model and adapted using samples from the new distribution only (i.e., $\mathcal{L}_1(\theta)$). Only the replay reference is allowed to sample from μ_0 during adaptation.

The study varies model size and sequence length to ensure that its conclusions are not tied to a single parameterization. We use sequence lengths $T \in \{8, 16, 32\}$ and three decoder sizes. The same task construction is repeated across five seeds.

Replay is optimized over an equal mixture of old and new examples:

$$\theta_{\mathrm{replay}} \in \arg\min_{\theta} \frac{1}{2}\mathcal{L}_0(\theta) + \frac{1}{2}\mathcal{L}_1(\theta). \tag{7}$$

In the conflict conditions, replay approximates the compromise in Equation 2. In the disjoint condition, it tests whether the chosen model and update carrier can represent both tasks. The conflict setting provides a deliberate calibration rather than an unexpected advantage for distillation, i.e., if $\mu_0 = \mu_1$ and the frozen teacher realizes the old target, then dense output distillation makes the distillation objective identical to replay, up to constants independent of θ. We use this case to verify the expected dense endpoint and then study how relaxing its contextual and positional coverage moves the student away from that reference.

3 A FAMILY OF ADAPTATION OBJECTIVES

To compare methods with different parameterizations, we separate the *adaptation carrier* from the *preservation constraint*. Let $\mathcal{C}$ specify how the source model may be changed, and let $\Theta_{\mathcal{C}}(\phi; \theta_0)$ denote the resulting student, with $\phi = 0$ recovering the source function. We optimize

$$\widehat{\phi} \in \arg\min_{\phi\in\Phi_{\mathcal{C}}} \mathcal{L}_1(\Theta_{\mathcal{C}}(\phi; \theta_0)) + \lambda \mathcal{R}_{M,g,A,W}\left(\Theta_{\mathcal{C}}(\phi; \theta_0), \theta_0\right). \tag{8}$$

The carrier $\mathcal{C}$ includes both weight- and representation-level adaptation. Ordinary LoRA uses low-rank weight updates (Hu et al., 2022). PiSSA and MiLoRA, modify the low-rank initializations (Meng et al., 2024; Wang et al., 2025). PSOFT and OLoRA constrain the weight geometry (Wu et al., 2026). ReFT instead applies a low-rank intervention to intermediate representations (Wu et al., 2024). Full fine-tuning is recovered by allowing $\Phi_{\mathcal{C}}$ to span the entire parameter space.

The second term in Equation 8 measures the teacher on selected computations from the new inputs:

$$\mathcal{R}_{M,g,A,W}(\theta, \theta_0) = \mathbb{E}_{x\sim M} \frac{1}{|A_x|} \sum_{a\in A_x} d(g_\theta(C_W(x, a)), g_{\theta_0}(C_W(x, a))). \tag{9}$$

In our old-data-free setting, $M = \mu_1$. The set A_x determines which positions are constrained, $C_W(x, a)$ is the causal context of width W ending at a, g_θ is the matched output or hidden representation, and d is the corresponding divergence. In language-model post-training, Proximal Policy Optimization (PPO) and Group Relative Policy Optimization (GRPO) are often augmented with a KL penalty to a frozen reference policy (Schulman et al., 2017; Shao et al., 2024). Direct Preference Optimization (DPO) instead trains on preferred and rejected responses using log-probability ratios relative to a frozen reference model (Rafailov et al., 2023). Thus, setting $\lambda = 0$ removes only our additional matching term $\mathcal{R}_{M,g,A,W}$, i.e., post-training then has no explicit source-function constraint, whereas PPO, GRPO, or DPO may still depend on the teacher model through their own objectives (which might explain their retention advantage over other methods (Shenfeld et al., 2026)).

From dense distillation to WINDOWLORA. The most constrained member of Equation 9 matches the teacher at every position of each new input using the complete causal prefix:

$$M = \mu_1, \qquad A_x = \{1, \dots, T\}, \qquad W = T, \qquad g_\theta = p_\theta, \qquad d(u, v) = D_{\mathrm{KL}}(v \| u). \tag{10}$$

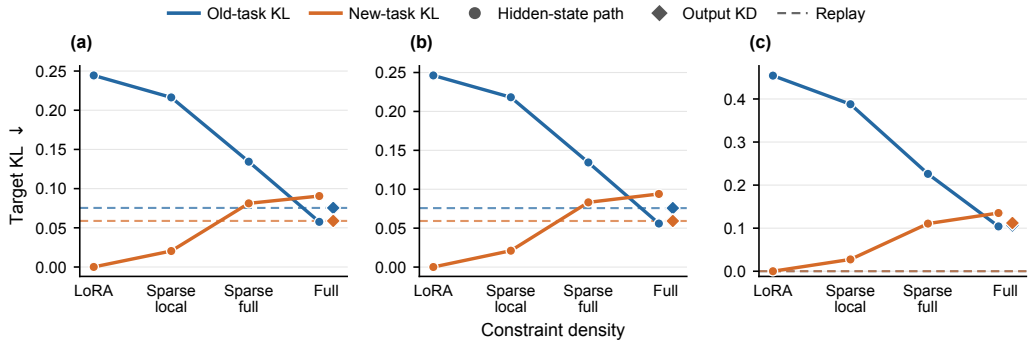


Figure 2: Retention–adaptation paths obtained by increasing the density of teacher constraints: (a) global conflict, (b) local conflict, and (c) compatible disjoint support. Under conflict, dense output distillation approaches the replay compromise; under disjoint support, replay performs well on both tasks while new-input matching induces an avoidable retention-adaptation tradeoff.

This recovers dense output distillation on the new-data distribution. Replacing p_θ with a hidden state $h_\theta^{(\ell)}$ and KL divergence with a representation distance gives its hidden-state counterpart. Starting from this endpoint, the preservation constraint can be relaxed in two independent ways: A_x can retain fewer matched positions, and W can restrict how far each position looks back. Our WINDOWLORA family follow the path from all-position full-prefix matching, through uniformly spaced full-prefix and local-window anchors, to LoRA. We refer to movement along this path simply as decreasing *constraint density* (see figure 1).

3.1 EVALUATING THE ADAPTED STUDENT

For an adapted student θ, we measure the old- and new-task target risks separately:

$$\mathcal{E}_k(\theta) = \mathbb{E}_{x\sim\mu_k} \frac{1}{|\mathcal{P}(x)|} \sum_{a\in\mathcal{P}(x)} D_{\mathrm{KL}}(q_k(\cdot \mid x_{\le a}) \,\|\, p_\theta(\cdot \mid x_{\le a})), \qquad k \in \{0,1\}. \tag{11}$$

The new-task risk $\mathcal{E}_1$ measures adaptation, while the old-task risk $\mathcal{E}_0$ measures retention. Importantly, both are evaluated against the known targets rather than against the teacher. A model can therefore match the teacher well on new inputs while still remaining far from the true old target on old inputs.

This WINDOWLORA family connects carrier-only adaptation to dense distillation through increasingly strong teacher constraints. Figure 2 shows that, for length-eight sequences, denser matching improves retention while reducing adaptation. Under global and local conflict, dense output distillation closely matches adapter replay because the new inputs cover the same model computations as the old inputs. The remaining tradeoff is intrinsic: the two tasks assign different targets to the same inputs. (Appendix B reports the remaining architectural protocol).

Under compatible disjoint support, replay remains near zero on both tasks, showing that there is no intrinsic tradeoff and that the carrier can represent a joint solution. Nevertheless, the old-data-free family exhibits a clear frontier: LoRA attains old/new KL $0.454/0.00005$, whereas dense distillation moves to $0.106/0.112$, substantially improving retention at the cost of adaptation. This preservation despite disjoint supports suggests that new inputs can probe computations shared with the old task. However, the remaining gap to replay shows that this transfer is incomplete and, without an additional coverage assumption, is not guaranteed. The length-16 and length-32 results in Appendix B.3 show the same pattern.

As formalized below, the decisive condition is not conflict itself but coverage: new-data matching can recover the old-task geometry only when its probes observe the old-sensitive directions of the adaptation carrier.

3.2 Relation to knowledge-adaptation priors

Knowledge-adaptation priors provide a useful criterion for deciding when such a substitute is exact (Khan & Swaroop, 2021). Let

$$\overline{\mathcal{L}}_0(\theta) = \sum_{z \in \mathcal{D}_0} \ell(z;\theta) + R_0(\theta), \qquad \theta_0 \in \arg\min_\theta \overline{\mathcal{L}}_0(\theta). \tag{12}$$

For the exponential-family generalized linear setting studied by Khan & Swaroop (2021), the full-memory prior is

$$\mathcal{K}_{\text{full}}(\theta;\theta_0) = \sum_{u \in \mathcal{D}_0} B_A(f_\theta(u), f_{\theta_0}(u)) + B_{R_0}(\theta,\theta_0), \tag{13}$$

where f_θ is the natural output parameter, B_A is the Bregman divergence induced by the log-partition function, and B_{R_0} is the weight-space divergence induced by the old regularizer. Under the assumptions of their result,

$$\nabla_\theta \mathcal{K}_{\text{full}}(\theta;\theta_0) = \nabla_\theta \overline{\mathcal{L}}_0(\theta). \tag{14}$$

The K-prior is therefore characterized by reconstruction of the missing old gradient, not merely by the use of a frozen teacher.

The family in Equation 9 is generally only a partial K-prior. It queries the teacher on μ_1 rather than on the old data, which creates a *coverage gap*. Hidden-state matching may induce a different geometry from the natural output loss, which creates a *matching gap*. Finally, the family does not automatically include $B_{R_0}(\theta,\theta_0)$, which creates a *completion gap*. Dense distillation closes none of these gaps automatically; it only increases contextual and positional coverage within the new probe distribution.

When the locations of target conflict are unknown, fixing a small set of anchors can create a permanent blind spot: the selected positions may never intersect the part of the sequence where preservation is needed. A natural alternative within our family is to sample m anchors uniformly at each update. As shown in Appendix B.4, the resulting sparse matching loss and its gradient are unbiased estimators of their all-position counterparts. In the shared-support conflict setting, where an exact teacher makes dense output matching equivalent to the old-task loss, random-anchor matching therefore provides an unbiased stochastic approximation to the replay preservation term, with variance decreasing as m increases.

More specifically, if conflict occurs at c of the T positions, a uniformly sampled set of m anchors intersects the conflict set with probability $1 - \binom{T-c}{m}/\binom{T}{m}$, and the probability of never observing a conflicting position over N independent updates is the corresponding miss probability raised to N. A fixed anchor set can miss the conflict permanently, whereas resampling eventually exposes it with high probability.

Prior functional-regularization methods already establish that preservation depends on where the reference function is queried. LwF evaluates the old model on current-task inputs, whereas FRCL and FROMP retain inducing or memorable inputs from previous tasks (Li & Hoiem, 2017; Titsias et al., 2020; Pan et al., 2021). We instead use a controlled sequence setting to separate this coverage limitation from unavoidable target conflict, and measure how contextual and positional relaxation moves a fixed adaptation carrier between LoRA and dense teacher matching.

4 Related Work

Retraining, replay, and continual adaptation. Continual fine-tuning can degrade old language-model capabilities, with the outcome depending on data order, similarity, and optimization (Luo et al., 2023; Li et al., 2024). Replay directly restores an old empirical loss but requires old data. Learning without Forgetting instead uses new-task inputs with old-model soft targets (Li & Hoiem, 2016), building on output distillation (Hinton et al., 2015). K-priors explain when a combination of function- and weight-space divergences reconstructs the past gradient and can recover retraining with a compact memory (Khan & Swaroop, 2021). Our paper imports this functional criterion into language-model post-training by adapt without claiming that the original theorem directly covers transformers.

Reference policies in post-training. RLHF commonly combines reward with a KL penalty to a fixed supervised reference model; PPO-ptx additionally mixes pretraining gradients (Ouyang et al., 2022). GRPO replaces PPO's learned value baseline with group-relative advantages while retaining, in its reference-regularized form, a fixed-policy KL term (Shao et al., 2024). DPO derives an offline preference objective from the optimum of KL-regularized reward maximization (Rafailov et al., 2023). We distinguish the fixed reference policy from the behavior policy used for importance ratios and clipping, and distinguish DPO's reference-relative margin from an explicit KL penalty.

Weight- and representation-efficient adaptation. LoRA learns a low-rank additive update (Hu et al., 2022); PiSSA initializes and performs adaptation in the principal singular subspace (Meng et al., 2024); empirically, this constraint can induce a capacity–retention tradeoff, with standard low-rank LoRA learning less in-domain than full fine-tuning but better preserving performance outside the target domain Biderman et al. (2024). MiLoRA does both on the minor singular subspace instead (Wang et al., 2025); PSOFT applies an orthogonal transformation within a principal subspace (Wu et al., 2026); and LoReFT learns low-rank representation interventions (Wu et al., 2024). These methods specify a carrier or feasible set for the effective update. Null-space continual-learning methods likewise restrict updates using feature covariance (Wang et al., 2021). Such constructions are defined in a weight or feature coordinate system; our analysis asks separately whether the realized network function remains stable during subsequent training.

Representation anchoring and drift. Recent work directly anchors internal representations during adaptation (Dalal et al., 2026), and state-aligned distillation compresses long-context behavior into adapters (Gao et al., 2026). WINDOWLORA instead uses identical local crops for teacher and student while a separate full-context branch learns the task. Representation measures such as CKA summarize network differences (Kornblith et al., 2019), but similarity need not imply functional equivalence (Hayne et al., 2024). We use paired activation drift as a evidence that a weight-space direction is functionally active; but this is more of diagnostics rather than causal evidence.

5 EXPERIMENTAL DESIGN

We next test whether the adaptation–retention tradeoffs identified in the controlled study also appear in pretrained language models. Our practical old-data-free variant, WINDOWLORA, uses rank-8 LoRA updates on the attention value and MLP up projections. It matches teacher and student hidden states at four uniformly spaced anchors,

$$A_L = \left\{ \frac{L}{4}, \frac{L}{2}, \frac{3L}{4}, L \right\}, \qquad W = \frac{L}{4}, \tag{15}$$

where W is the causal-window width.

5.1 MODELS AND PROTOCOLS

Qwen2.5-1.5B. We independently adapt Qwen2.5-1.5B (Qwen Team, 2025) to PIQA, SocialIQA, HellaSwag, WinoGrande, ARC-Easy, ARC-Challenge, and OpenBookQA (Bisk et al., 2020; Sap et al., 2019; Zellers et al., 2019; Sakaguchi et al., 2020; Clark et al., 2018; Mihaylov et al., 2018). Each run uses 1,024 training examples, 256 validation examples, one epoch (64 updates), and three seeds. We compare LoRA, WINDOWLORA, PiSSA, MiLoRA, PSOFT, and LoReFT, together with the reference-regularized post-training baselines PPO, GRPO, and DPO (Shenfeld et al., 2026; Nunez et al., 2026).

Llama-3.1-8B. We additionally evaluate transfer from Llama-3.1-8B (Llama Team, 2024). Commonsense-15K contains 15,119 examples from the seven tasks above and BoolQ, with 120 examples reserved for checkpoint selection. Evaluation covers all eight complete task sets. LoReFT is applied after block 16 with rank 4 and therefore uses a different parameter budget. A MetaMathQA-40K transfer study is reported in Appendix C.5. Full hyperparameters and checkpoint revisions appear in Appendix C.3.

5.2 FUNCTIONAL DRIFT

We measure activation drift on Qwen at all 28 layers. For paired source and adapted models, we record query, key, and value activations before positional encoding, together with the 1,536-dimensional MLP down-projection output before residual addition. Task probes use answer-prediction positions from 16 validation examples per task; retention probes use continuation positions from eight WikiText windows. We compute relative RMS differences, normalize them within each layer, and average across probe positions, layers, and task adapters.

We also project activation differences onto PCA bases fitted to source-model calibration activations. The leading subspace explains 90% of source activation energy, and its orthogonal complement defines the residual subspace. These activation-derived subspaces are distinct from the weight-SVD subspaces used by some adaptation methods; the residual component is therefore not an exact functional null space.

5.3 ADAPTATION AND RETENTION

For Qwen, adaptation is measured by answer-token and (EOS) NLL, averaged equally across the seven tasks within each seed. For Llama, we report deterministic generated-answer accuracy and answer-only NLL, averaged equally across the eight evaluation sets. Checkpoints are selected using full-sequence validation loss on the 120-example holdout.

Retention is measured by perplexity on fixed WikiText and Penn Treebank test prefixes (Merity et al., 2017; Marcus et al., 1993), using context length 1,024 and stride 512 so that each target token is scored once. We report each corpus separately and use their arithmetic mean only for visualization. Because perplexity captures only one aspect of retention, we report adaptation and retention separately rather than combining them into a single score.

6 RESULTS

6.1 WEIGHT SUBSPACES DO NOT DEFINE FUNCTIONAL SUBSPACES

Table 8 shows that weight-space constraints do not necessarily limit functional drift. On the Qwen retention probes, PiSSA and MiLoRA produce roughly three times as much activation drift as LoRA, while PSOFT produces about twice as much. The same pattern holds when source activations are separated into leading and residual PCA components (Appendix D.2): both components drift across attention and MLP representations. Consistent with Equation 17, the SVD of an individual weight matrix does not capture the end-to-end Jacobian geometry on protected inputs. Although these experiments do not dispute the optimization benefits of structural weight constraint, they challenge the stronger assumption that their selected weight directions are functionally harmless. LoReFT produces less than 1% drift across all measured components, showing evidence of distinction between weight geometry and functional change. These measurements remain diagnostic rather than causal.

6.2 ADAPTATION AND RETENTION REMAIN IN TENSION

Tables 1 and 2 report the two model protocols separately.

Qwen2.5-1.5B. LoRA achieves the strongest task adaptation, but increases both retention perplexities by approximately 1–2%. PiSSA, MiLoRA, and PSOFT adapt strongly but forget more, whereas LoReFT remains closer to the source with weaker adaptation. PPO, GRPO, and DPO also preserve source perplexity, but provide little improvement over the source task loss. Thus, apparent retention can reflect insufficient adaptation rather than a uniformly better solution.

Llama-3.1-8B. Under the Commonsense-15K protocol, WINDOWLORA reduces LoRA's retention perplexities by approximately 2%, while lowering generated-answer accuracy by 2.3 percentage points. LoReFT remains closest to the source but loses nearly 20 accuracy points relative to LoRA. This reveals the same adaptation–retention tradeoff under a second protocol, rather than providing a direct comparison across model sizes.

Table 1: Qwen2.5-1.5B results. Task NLL averages seven fixed-subset adaptations equally; retention uses WikiText-128K/PTB-32K. Values are mean ± sample SD over seeds 17/18/19; the base is deterministic. Lower is better. Policy methods optimize different objectives.

Family	Method	Training signal	Task NLL ↓	WikiText PPL ↓	PTB PPL ↓
Base	Original	—	3.1777	8.5216	16.3371
SFT	LoRA	SFT	2.1928 ± 0.0010	8.6261 ± 0.0078	16.5803 ± 0.0173
SFT	**WINDOWLORA**	SFT + local state	2.2115 ± 0.0014	8.5688 ± 0.0059	16.4446 ± 0.0024
SFT	PiSSA	SFT	2.2641 ± 0.0055	9.5707 ± 0.0904	18.8027 ± 0.0724
SFT	MiLoRA	SFT	2.2031 ± 0.0031	9.6444 ± 0.1065	18.4909 ± 0.2556
SFT	PSOFT	SFT	2.2439 ± 0.0014	8.8808 ± 0.0208	17.3840 ± 0.0398
SFT	LoReFT	SFT	2.2847 ± 0.0019	8.5231 ± 0.0003	16.3517 ± 0.0035
Policy	PPO	choice reward + KL	3.2474 ± 0.0659	8.5343 ± 0.0010	16.3438 ± 0.0023
Policy	GRPO	choice reward + KL	3.2414 ± 0.0704	8.5359 ± 0.0018	16.3439 ± 0.0014
Policy	DPO	correct/incorrect pair	3.1763 ± 0.0346	8.5334 ± 0.0013	16.3431 ± 0.0010

Table 2: Llama-3.1-8B Commonsense-15K transfer. Accuracy/NLL average eight full evaluation sets equally; retention uses model-specific tokenization. Adapted rows are mean ± sample SD over seeds 17/18/19; the base is deterministic. Protocols differ from Qwen.

Method	Accuracy ↑	Task NLL ↓	WikiText ↓	PTB ↓	Mean PPL ↓
Orig.	21.3028	1.8231	5.8767	9.9022	7.8894
LoRA	82.5035 ± 0.3423	0.0642 ± 0.0006	6.0406 ± 0.0092	10.2413 ± 0.0249	8.1409 ± 0.0140
WINDOWLORA	80.1990 ± 0.7160	0.0693 ± 0.0012	5.9290 ± 0.0023	9.9984 ± 0.0168	7.9637 ± 0.0094
LoReFT	63.4166 ± 0.7223	0.1095 ± 0.0014	5.8772 ± 0.0001	9.9095 ± 0.0017	7.8934 ± 0.0009

6.3 WINDOWLORA SHIFTS THE OBSERVED FRONTIER

On Qwen, WINDOWLORA recovers over half of LoRA's perplexity increase with higher task NLL, reducing retention-probe value and MLP drift by 36% and 38%, and task-probe MLP drift by 19%. Retention improves across all seven adapters, with similar trends on Llama.

Ablations confirm the stability–plasticity trade-off: stronger anchoring monotonically reduces drift but weakens adaptation. The best budgeted setting uses the final layer, $\lambda = 0.3$, four uniform anchors, and width $L/4$ (Appendix D.3–D.4); fixed anchors favor stability. Generation effects are mixed, but retention improves consistently. Thus, WINDOWLORA offers a useful operating point, not a dominant or replay-equivalent method: it uses new-data probes, and intermediate targets to approximate what K-prior (or a replay) offers.

7 CONCLUSION

This work shows that a retention–adaptation frontier alone does not explain why a model forgets. Controlled sequence tasks distinguish intrinsic trade-offs, where conflicting targets force even replay to compromise, from method-induced trade-offs, where old-data-free constraints only approximate the missing old-task objective. We formalize WINDOWLORA as a partial K-prior that interpolates between unconstrained LoRA and dense teacher matching by varying the positional and contextual coverage of local hidden-state anchors. Under conflict, denser constraints approach the replay compromise; with compatible disjoint support, replay solves both tasks while new-input matching retains a nonzero frontier, showing that incomplete probe coverage can itself impose adaptation costs.

Real-task results support this functional view where structured weight subspaces do not limit activation drift, whereas WINDOWLORA reduces drift and improves retention over LoRA on Qwen2.5-1.5B and Llama-3.1-8B, at a modest task-performance cost. Across the old-data-free adaptation methods we study, stronger adaptation is consistently accompanied by weaker retention, or vice versa. The source of this tradeoff differs: it is intrinsic when old and new targets conflict, but method-induced when preservation constraints incompletely cover the old task. WINDOWLORA provides a controllable bridge from unconstrained LoRA to dense teacher matching, making this frontier explicit and allowing practitioners to navigate it through constraint density and strength.

Reproducibility statement. Appendix C gives experimental and provenance details.

AI use statement. Generative AI assisted language editing and code drafting; the authors reviewed and take responsibility for all material and measurements.

REFERENCES

Dan Biderman, Jacob Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, et al. Lora learns less and forgets less. *arXiv preprint arXiv:2405.09673*, 2024.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7432–7439, 2020. doi: 10.1609/aaai.v34i05.6239.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. In *arXiv preprint arXiv:1803.05457*, 2018. URL https://arxiv.org/abs/1803.05457.

Dwip Dalal, Shivansh Patel, Chahit Jain, Jeonghwan Kim, Utkarsh Mishra, Alex Baratian, Hyeonjeong Ha, Heng Ji, Svetlana Lazebnik, and Unnat Jain. Generalizable VLA finetuning via representation anchoring and language-action alignment. *arXiv preprint arXiv:2607.13429*, 2026. URL https://arxiv.org/abs/2607.13429.

Wenhao Gao, Tianlong Wang, Wei Jia, Linhao Zhang, Aiwei Liu, Miao Fan, and Zhou Xiao. SADA: Bridging in-context learning and fine-tuning via state-aligned distillation adapters. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics*, pp. 22847–22862, 2026. doi: 10.18653/v1/2026.acl-long.1046. URL https://aclanthology.org/2026.acl-long.1046/.

Lucas Hayne, Heejung Jung, and R. McKell Carter. Does representation similarity capture function similarity? *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=YY2iA0hfia.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. In *NeurIPS Deep Learning and Representation Learning Workshop*, 2015. URL https://arxiv.org/abs/1503.02531.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Mohammad Emtiyaz Khan and Siddharth Swaroop. Knowledge-adaptation priors. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/a4380923dd651c195b1631af7c829187-Abstract.html.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 3519–3529, 2019. URL https://proceedings.mlr.press/v97/kornblith19a.html.

Hongyu Li, Liang Ding, Meng Fang, and Dacheng Tao. Revisiting catastrophic forgetting in large language model tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 4297–4308, 2024. doi: 10.18653/v1/2024.findings-emnlp.249. URL https://aclanthology.org/2024.findings-emnlp.249/.

Zhizhong Li and Derek Hoiem. Learning without forgetting. In *European Conference on Computer Vision*, pp. 614–629, 2016. doi: 10.1007/978-3-319-46493-0_37.

Zhizhong Li and Derek Hoiem. Learning without forgetting, 2017. URL https://arxiv.org/abs/1606.09282.

Llama Team. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*, 2023. URL https://arxiv.org/abs/2308.08747.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993. URL https://aclanthology.org/J93-2004/.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. PiSSA: Principal singular values and singular vectors adaptation of large language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-3846. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/db36f4d603cc9e3a2a5e10b93e6428f2-Abstract-Conference.html.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Byj72udxe.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2381–2391, 2018. doi: 10.18653/v1/D18-1260. URL https://aclanthology.org/D18-1260/.

Jeanmely Rojas Nunez, Viraj Sawant, Nathan Allen, Nomgondalai Amgalanbaatar, Yannis Zongo, Vasu Sharma, and Maheep Chaudhary. Mechanistic origins of catastrophic forgetting: why rl preserves circuits better than sft? *arXiv preprint arXiv:2605.28860*, 2026.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35, 2022. URL https://arxiv.org/abs/2203.02155.

Pingbo Pan, Siddharth Swaroop, Alexander Immer, Runa Eschenhagen, Richard E. Turner, and Mohammad Emtiyaz Khan. Continual deep learning by functional regularisation of memorable past, 2021. URL https://arxiv.org/abs/2004.14070.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025. URL https://arxiv.org/abs/2412.15115.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/a85b405ed65c6477a4fe8302b5e06ce7-Abstract-Conference.html.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8732–8740, 2020. doi: 10.1609/aaai.v34i05.6399.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 4463–4473, 2019. doi: 10.18653/v1/D19-1454. URL https://aclanthology.org/D19-1454/.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. URL https://arxiv.org/abs/1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. Rl's razor: Why online reinforcement learning forgets less. In *International Conference on Learning Representations*, volume 2026, pp. 59839–59864, 2026.

Michalis K. Titsias, Jonathan Schwarz, Alexander G. de G. Matthews, Razvan Pascanu, and Yee Whye Teh. Functional regularisation for continual learning with gaussian processes, 2020. URL https://arxiv.org/abs/1901.11356.

Hanqing Wang, Yixia Li, Shuo Wang, Guanhua Chen, and Yun Chen. MiLoRA: Harnessing minor singular components for parameter-efficient LLM finetuning. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4823–4836, 2025. doi: 10.18653/v1/2025.naacl-long.248. URL https://aclanthology.org/2025.naacl-long.248/.

Shipeng Wang, Xiaorong Li, Jian Sun, and Zongben Xu. Training networks in null space of feature covariance for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 184–193, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Wang_Training_Networks_in_Null_Space_of_Feature_Covariance_for_Continual_CVPR_2021_paper.html.

Fei Wu, Jia Hu, Geyong Min, and Shiqiang Wang. Efficient orthogonal fine-tuning with principal subspace adaptation. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=FSHrinMArK.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D. Manning, and Christopher Potts. ReFT: Representation finetuning for language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://openreview.net/forum?id=fykjplMc0V.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, 2019. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

A LIMITATIONS

Scope of the controlled study. The diagnostic separation between intrinsic and method-induced tradeoffs rests on small synthetic problems: binary strings of length $T \leq 32$, a fixed four-class oracle, and decoders with at most $\sim$137K parameters. The two relationships we construct are idealized extremes. Conflict is an exact $\pm\delta$ perturbation on a shared support, and compatibility is a clean first-token partition. Natural task pairs likely combine partial support overlap, graded target disagreement, and shared features whose relevance differs across tasks. Our attribution of the conflict frontier to an intrinsic cause also assumes that finite-capacity, finite-step replay approximates the unrestricted optimum in Equation 2. Finally, the agreement between dense output distillation and replay under conflict holds by construction when $\mu_0 = \mu_1$ and the teacher realizes q_0. It calibrates the endpoint of the family; it is not independent evidence that distillation preserves well in general.

Theoretical connection. The K-prior result holds for exponential-family generalized linear models; we use it as an interpretive criterion, not as a guarantee for transformers. WINDOWLORA omits the weight-space term B_{R_0}, and its hidden-state MSE does not correspond to the Bregman divergence of the natural output loss, so it is a partial K-prior on all three gaps we identify. The central coverage condition, that probes must observe the old-sensitive directions of the carrier, is stated qualitatively. We do not measure those directions or verify coverage in either the synthetic or the pretrained setting.

Drift is diagnostic, not causal. Activation drift is a layer-normalized relative RMS on a small probe set (16 task examples per task and eight WikiText windows). The PCA residual subspace is not a functional null space. We do not show that reducing value or MLP drift causes better retention; interventional experiments would be needed to establish this.

Retention and adaptation measurement. Retention is measured only by perplexity on WikiText and PTB prefixes. This misses reasoning, factual recall, in-context learning, instruction following, and safety behavior, which may degrade differently. Both source models are base checkpoints, so the results may not extend to post-training instruction-tuned models. Qwen adaptation uses 1,024 examples and 64 updates per task. In this short regime, LoRA changes retention perplexity by only 1–2%, and differences between methods are correspondingly small. Adaptation is evaluated on fixed 256-example validation subsets rather than full benchmarks. The coefficient λ was selected on the same subsets used for the reported results, which may bias results in favor of WINDOWLORA. Error bars are sample standard deviations over three seeds, not confidence intervals, and we apply no multiple-comparison correction.

Capacity and optimization. The Qwen methods differ in trainable parameters (18.46M for the LoRA family, 0.110M for PSOFT, 0.012M for LoReFT) and in learning rate (2×10^{-4}, 10^{-3}, and 4×10^{-3}, respectively). PiSSA and MiLoRA reuse LoRA's learning rate even though their initializations place the update in directions with very different scales. Their larger drift may therefore reflect an untuned step size rather than an inherent property of principal- or minor-subspace adaptation. Likewise, LoReFT's low drift and weak adaptation cannot be separated from its roughly $1500\times$ smaller budget and single intervention site.

B CONTROLLED SYNTHETIC STUDY

B.1 TASK CONSTRUCTION

The controlled study separates two sources of an adaptation–retention frontier. In the global- and local-conflict conditions, the old and new tasks share the same input support but assign different targets. Replay must therefore compromise. In the compatible-disjoint condition, the first token partitions the two input supports while both tasks follow the same target rule. Replay can consequently achieve low risk on both tasks. Table 3 illustrates the three constructions for length-eight strings; the oracle distribution ρ and perturbations δ_{global} and δ_{local} are defined in Section 3.

Table 3: Examples of the controlled task relationships for length-eight strings. Conflict conditions use the same inputs with opposing target perturbations. Compatible disjoint support partitions inputs by their first token while retaining a shared target rule.

Condition	Old inputs and targets	New inputs and targets	Replay optimum
Global conflict	$x = 00110110$, $q_0 = \rho - \delta_{\text{global}}$	$x = 00110110$, $q_1 = \rho + \delta_{\text{global}}$	Nonzero compromise
Local conflict	$x = 00110110$, $q_0 = \rho - \delta_{\text{local}}$	$x = 00110110$, $q_1 = \rho + \delta_{\text{local}}$	Nonzero compromise
Compatible disjoint	$x = 00110110$, $q_0 = \rho$	$x = 10110110$, $q_1 = \rho$	Low risk on both tasks

B.2 Architecture and optimization

The oracle is a two-layer causal decoder with width 16, four attention heads, and feed-forward width 32. Students use four attention heads and vary in depth and width as summarized in Table 4. LoRA uses rank eight and is applied to the attention value and MLP up projections in every block. The adaptation methods are optimized for 600 updates, while replay and other reference solutions use 1,200 updates. Unless stated otherwise, results are averaged over five optimization seeds.

Table 4: Decoder configurations used in the synthetic study. Parameter counts are reported for sequence length $T = 8$; learned positional embeddings cause the base count to increase slightly for longer sequences. LoRA parameter counts do not depend on sequence length.

Student	Layers	d_{model}	Heads	d_{ff}	Base parameters	LoRA parameters
Small	2	16	4	32	5,124	1,280
Medium	3	32	4	64	26,980	3,840
Large	4	64	4	128	136,580	10,240

For $T = 8$, training evaluates all 256 binary strings. For $T \in \{16, 32\}$, we sample 512 complete strings for training and 512 distinct strings for evaluation. All runs use AdamW with learning rate 3×10^{-3}, zero weight decay, gradient clipping at one, and cosine decay to 10% of the initial learning rate. The five optimization seeds are 71–75.

B.3 Coefficient and sequence-length studies

Constraint density determines which teacher computations are matched, whereas λ determines the strength of those constraints. We separate these effects by sweeping λ while holding the carrier, anchors, and context width fixed. Figure 3 shows that increasing λ continuously exchanges new-task fit for old-task retention. Under conflict, the path approaches the replay compromise. Under compatible disjoint support, replay remains near zero on both tasks, whereas new-input matching traces an avoidable frontier caused by incomplete old-task supervision.

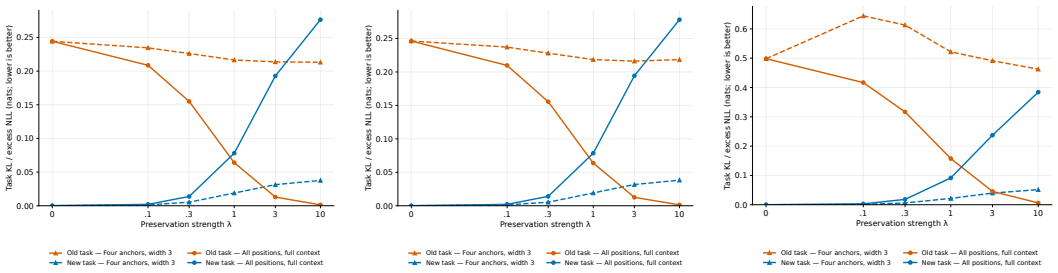
Figures 4 and 5 repeat the density study at longer sequence lengths. Both reproduce the main pattern: conflict creates an intrinsic replay compromise, whereas compatible disjoint support admits a joint solution that old-data-free matching does not fully recover.

B.4 Random-anchor coverage

The matching family permits the anchor set A_x to be sampled rather than fixed. Uniform resampling gives an unbiased stochastic approximation to all-position matching and avoids permanent blind spots from a fixed sparse set.

For an input x, let $\mathcal{P}(x)$ denote its valid causal positions and $T_x = |\mathcal{P}(x)|$. For a fixed context width W, representation g, and divergence d, define

$$r_a(\theta, x) = d(g_\theta(C_W(x, a)), g_{\theta_0}(C_W(x, a))) \,. \tag{16}$$



(a) Global conflict (b) Local conflict (c) Compatible disjoint support

Figure 3: Retention–adaptation frontiers obtained by varying the preservation weight λ. Each curve holds the adaptation carrier and matching configuration fixed while changing only the strength of teacher matching.

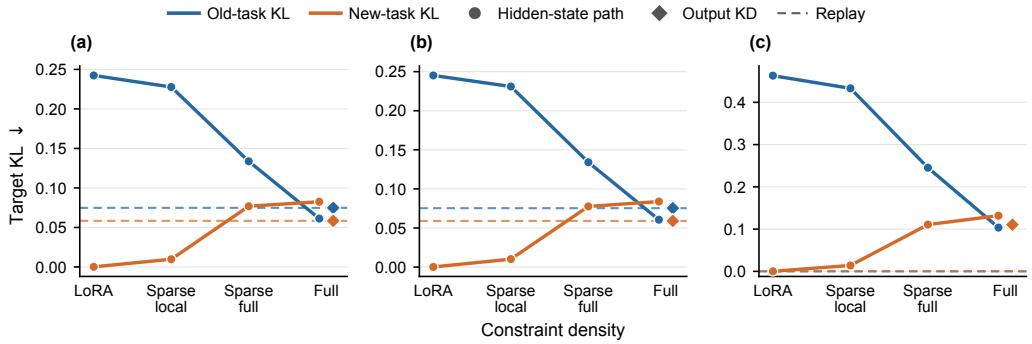


Figure 4: Retention–adaptation paths at $T = 16$ as constraint density increases: (a) global conflict, (b) local conflict, and (c) compatible disjoint support.

The all-position objective is

$$\mathcal{R}_{\text{all}}(\theta) = \mathbb{E}_{x\sim\mu_1} \frac{1}{T_x} \sum_{a\in\mathcal{P}(x)} r_a(\theta, x). \tag{17}$$

Let A_x contain $m \le T_x$ positions sampled uniformly without replacement at each optimization step, and define

$$\widehat{\mathcal{R}}_m(\theta) = \mathbb{E}_{x\sim\mu_1} \frac{1}{m} \sum_{a\in A_x} r_a(\theta, x). \tag{18}$$

Lemma 1 (Unbiased random-anchor matching)**.** *For every θ,*

$$\mathbb{E}_A[\widehat{\mathcal{R}}_m(\theta)] = \mathcal{R}_{\text{all}}(\theta). \tag{19}$$

If differentiation and expectation can be exchanged, then

$$\mathbb{E}_A[\nabla_\theta \widehat{\mathcal{R}}_m(\theta)] = \nabla_\theta \mathcal{R}_{\text{all}}(\theta). \tag{20}$$

Conditional on x, let $g_a = \nabla_\theta r_a(\theta, x)$, $\overline{g}_x = T_x^{-1} \sum_{a\in\mathcal{P}(x)} g_a$, and $\widehat{g}_x = m^{-1} \sum_{a\in A_x} g_a$. Then

$$\mathbb{E}_A\left[\|\widehat{g}_x - \overline{g}_x\|_2^2\right] = \frac{1 - m/T_x}{m} \frac{1}{T_x - 1} \sum_{a\in\mathcal{P}(x)} \|g_a - \overline{g}_x\|_2^2. \tag{21}$$

Thus, increasing the number of sampled anchors reduces gradient variance and recovers the dense gradient when $m = T_x$.

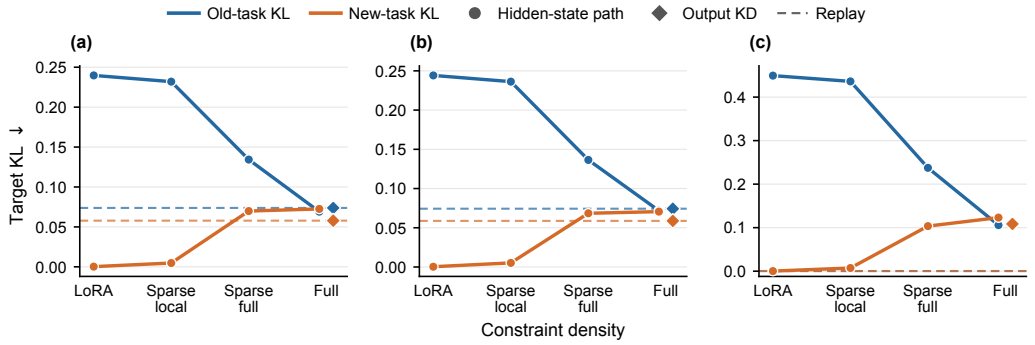
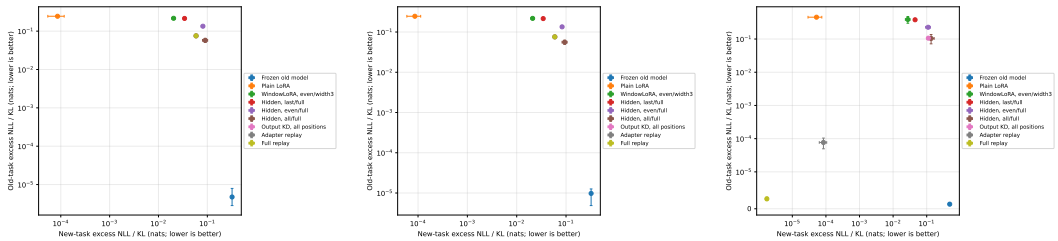
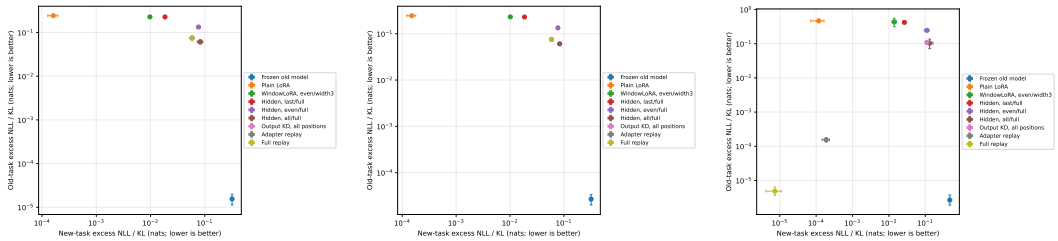


Figure 5: Retention–adaptation paths at $T = 32$ as constraint density increases: (a) global conflict, (b) local conflict, and (c) compatible disjoint support.



(a) $T = 8$: global conflict (b) $T = 8$: local conflict (c) $T = 8$: compatible disjoint



(d) $T = 16$: global conflict (e) $T = 16$: local conflict (f) $T = 16$: compatible disjoint

Figure 6: Per-condition synthetic results for $T = 8$ and $T = 16$. Columns correspond to global conflict, local conflict, and compatible disjoint support.

Proof. Each position has inclusion probability m/T_x. Conditional on x,

$$\mathbb{E}_A\left[\frac{1}{m}\sum_{a\in A_x} r_a(\theta, x)\right] = \frac{1}{m}\sum_{a\in\mathcal{P}(x)} \frac{m}{T_x} r_a(\theta, x) \tag{22}$$

$$= \frac{1}{T_x}\sum_{a\in\mathcal{P}(x)} r_a(\theta, x). \tag{23}$$

Taking expectation over x proves Equation 19; applying the same argument to the gradients proves Equation 20. Equation 21 is the finite-population variance of a sample mean drawn uniformly without replacement. □

Connection to shared-support conflict. Suppose $\mu_0 = \mu_1$, the teacher realizes the old target, and r_a is the output KL from teacher to student. All-position matching then equals the old risk:

$$\mathcal{R}_{\text{all}}(\theta) = \mathcal{L}_0(\theta). \tag{24}$$

Consequently,

$$\mathbb{E}_A[\widehat{\mathcal{R}}_m(\theta)] = \mathcal{L}_0(\theta), \qquad \mathbb{E}_A[\nabla_\theta \widehat{\mathcal{R}}_m(\theta)] = \nabla_\theta \mathcal{L}_0(\theta). \tag{25}$$

Table 5: How common post-training methods relate to the shared preservation constraint. *None* denotes no explicit comparison with the source function.

Method	Adaptation signal	Explicit anchor	Anchor probes	Update geometry
Full SFT	answer likelihood	none	—	all trainable weights
LoRA	answer likelihood	none	—	low-rank weight updates
PiSSA	answer likelihood	none	—	principal-component initialization
MiLoRA	answer likelihood	none	—	minor singular-component updates
Adam-NSCL	sequential task loss	layer-input features	previous-task features	feature-covariance null space
LoReFT	answer likelihood	none	—	low-rank representation interventions
PSOFT	answer likelihood	none	—	orthogonal principal-subspace update
Output KD/LwF	task loss	output divergence	transfer inputs	chosen carrier
WINDOWLORA	full-context SFT	final-state distance	local windows	LoRA
Retention replay	task + old empirical loss	old risk	retained old data	chosen carrier
DPO	pairwise preference	reference margin	preference pairs	chosen carrier
PPO/GRPO-KL	policy objective	reference-policy KL	prompts/rollouts	chosen carrier

Random-anchor output matching is therefore an unbiased stochastic approximation to the replay preservation term in this calibrated setting. When replay uses weight γ, setting $\lambda = \gamma$ matches its expected objective.

Probability of observing positional conflict. Let $\mathcal{C}_x \subseteq \mathcal{P}(x)$ contain the c_x positions at which the old and new targets disagree. A uniformly sampled set of m anchors misses all conflicting positions with probability

$$\Pr(A_x \cap \mathcal{C}_x = \varnothing) = \frac{\binom{T_x - c_x}{m}}{\binom{T_x}{m}}, \tag{26}$$

and therefore observes at least one conflicting position with probability

$$\Pr(A_x \cap \mathcal{C}_x \neq \varnothing) = 1 - \frac{\binom{T_x - c_x}{m}}{\binom{T_x}{m}}. \tag{27}$$

Under independent resampling for N steps, the probability of never observing conflict is

$$\left[\frac{\binom{T_x - c_x}{m}}{\binom{T_x}{m}}\right]^N. \tag{28}$$

This follows by counting the $\binom{T_x - c_x}{m}$ non-conflicting anchor sets among all $\binom{T_x}{m}$ possible sets and using independence across steps. A fixed sparse set can miss conflict permanently; resampling removes this deterministic blind spot. It is not universally superior to targeted anchors when the important positions are known.

C REAL-MODEL METHODS AND REPRODUCIBILITY

C.1 METHOD MAP

C.2 QWEN2.5-1.5B PROTOCOL

Data contract. The base checkpoint is `Qwen/Qwen2.5-1.5B` at revision `8faed761d45a263340a0528343f099c05c9a4323`. The primary comparison contains 189 task-specific checkpoints: nine methods, seven tasks, and seeds 17, 18, and 19. Each task uses training rows 0–1,023 and validation rows 0–255. Every run completes one epoch and 64 optimizer updates. The artifact checks the complete $9 \times 7 \times 3$ matrix against the packaged comparison records before generating the tables.

Loss and aggregation. The answer-only causal objective masks prompt and padding tokens and scores all answer tokens and EOS. Validation NLL is weighted by the number of scored tokens within each task. Macro NLL averages the seven tasks equally within a seed and then reports the mean and sample SD across seeds. Retention uses context length 1,024 and stride 512, masking overlap tokens already scored in the preceding window. WikiText contains 131,072 evaluated tokens and PTB contains 32,768. When shown, combined retention is the arithmetic mean of the two corpus perplexities.

Audit and uncertainty. The transferred bundle contains SHA-256 source maps, 3,849 valid JSON files, 189 primary performance records, 189 matching activation records, and a 27-test activation audit. Portable regeneration reconstructs the performance macro table and checks drift aggregation. Error bars are sample SD across three training seeds, not confidence intervals, and exclude uncertainty from fixed data subsets and diagnostic probes. The coefficient λ was selected on the same validation subsets used for the reported validation results.

C.3 Adaptation-method configurations

Qwen parameter-efficient methods. LoRA, WindowLoRA, PiSSA, and MiLoRA use rank 16, scale 32, all linear attention and MLP projections, 18.46M trainable parameters, and learning rate 2×10^{-4}. PSOFT uses rank 32, 0.110M trainable parameters, and learning rate 10^{-3}. LoReFT uses a rank-four intervention after block 14, 0.012M trainable parameters, and learning rate 4×10^{-3}. These methods therefore do not share a common trainable-parameter budget.

Policy-based methods. PPO and GRPO use an eight-action answer-choice policy, correctness reward, clipping coefficient 0.2, and exact categorical KL to the base policy with coefficient 0.04. DPO pairs each correct answer with a seeded incorrect answer and uses $\beta = 0.1$.

C.4 WindowLoRA configuration and ablation grid

For each padded batch of length L, Qwen anchors are placed at $\lfloor kL/4 \rfloor$ for $k = 1, \ldots, 4$, with window width $\lfloor L/4 \rfloor$. Padding endpoints are discarded. Teacher and student receive identical crops and absolute position indices; the teacher disables adapters and is detached. The regularizer is the mean endpoint MSE over the final normalized hidden dimensions. Teacher targets may be cached, and all student windows are evaluated in one local batch. Task and regularization losses share one backward pass and optimizer step.

The default coefficient is $\lambda = 0.3$. It is selected subject to a maximum task-NLL increase of 1% on average and 2% on any individual task. The main Qwen results therefore reflect validation-based tuning rather than evaluation on untouched test sets. For Llama, we apply the same four-window rule and coefficient to representations after block 32.

The Qwen ablations use $\lambda \in \{0, 0.1, 0.3, 1, 3, 10\}$, compare middle- and final-layer anchoring, window widths $L/8$, $L/4$, and $L/2$, and fixed versus entropy-based anchor selection. Secondary studies include three-epoch BoolQ, GSM8K and XSum generation, and a one-seed SFT/PPO/GRPO pilot. These studies are not part of the 189-checkpoint primary matrix.

C.5 Llama-3.1-8B protocol

The base checkpoint is `meta-llama/Llama-3.1-8B` at revision `d04e592bb4f6aa9cfee91e2e20afa771667e1d4b`. Commonsense-15K contains 15,119 mixed examples. A seed-42 split reserves 120 examples for checkpoint selection and uses the remaining 14,999 for three epochs, or 705 updates. The evaluation sets contain 3,270 BoolQ, 1,838 PIQA, 1,954 SocialIQA, 10,042 HellaSwag, 1,267 WinoGrande, 2,376 ARC-Easy, 1,172 ARC-Challenge, and 500 OpenBookQA examples. Metrics average tasks equally within each seed and then report mean and sample SD over seeds 17, 18, and 19.

LoRA and WindowLoRA use rank 16 and scale 32 on all linear projections. LoReFT uses a rank-four intervention after block 16 and therefore has a different parameter budget. Two ARC-Easy test prompts exactly match training prompts; excluding them changes LoRA/WindowLoRA/LoReFT ARC-Easy accuracy from $91.5685/90.3058/79.9944\%$ to $91.5614/90.2977/79.9775\%$. Exact matching does not rule out semantic or pretraining contamination.

The MetaMathQA-40K transfer trains LoRA and WindowLoRA on all 40,000 examples for two epochs and evaluates all 1,319 GSM8K and 5,000 MATH problems. It was stopped after seed 17; LoReFT, additional seeds, and an original-model math baseline were not completed. Table 6 therefore supports only a single-seed transfer observation.

Table 6: Llama-3.1-8B MetaMathQA-40K transfer, seed 17 only. Accuracy and NLL average GSM8K and MATH equally. No seed uncertainty is estimated.

Method	Accuracy ↑	Task NLL ↓	WikiText ↓	PTB ↓	Mean PPL ↓
LoRA	51.1759	0.9046	6.1107	10.1049	8.1078
WindowLoRA	50.6652	0.8711	5.9170	9.9631	7.9400

Llama retention uses the same text lengths and sliding-window protocol as Qwen, retokenized with the Llama tokenizer. We do not measure Llama $Q/K/V$/MLP drift or activation subspaces, so Qwen and Llama provide separate rather than pooled evidence.

D Additional Qwen Results

D.1 Per-task and activation results

Table 7: Per-dataset answer-token NLL for the primary comparison. Each entry is mean ± sample SD across three seeds; Avg. averages the seven task means equally. These are fixed 256-example validation subsets.

Method	PIQA	SIQA	Hella	Wino	ARC-E	ARC-C	OBQA	Avg.
Original	–	–	–	–	–	–	–	3.1777
LoRA	2.2840 ± 0.0009	2.4408 ± 0.0077	2.2894 ± 0.0057	2.0029 ± 0.0031	1.3672 ± 0.0006	1.8656 ± 0.0011	3.0999 ± 0.0114	2.1928 ± 0.0010
WindowLoRA	2.2927 ± 0.0014	2.4501 ± 0.0083	2.3146 ± 0.0040	2.0222 ± 0.0023	1.3843 ± 0.0031	1.8915 ± 0.0052	3.1254 ± 0.0115	2.2115 ± 0.0014
PiSSA	2.3605 ± 0.0012	2.5188 ± 0.0102	2.3396 ± 0.0074	2.0401 ± 0.0208	1.4864 ± 0.0266	1.9343 ± 0.0079	3.1692 ± 0.0197	2.2641 ± 0.0055
MiLoRA	2.3033 ± 0.0016	2.4480 ± 0.0094	2.2988 ± 0.0100	2.0038 ± 0.0046	1.3833 ± 0.0146	1.8705 ± 0.0099	3.1137 ± 0.0069	2.2031 ± 0.0031
PSOFT	2.3155 ± 0.0037	2.4713 ± 0.0107	2.3438 ± 0.0027	2.1328 ± 0.0100	1.4075 ± 0.0070	1.9073 ± 0.0051	3.1292 ± 0.0026	2.2439 ± 0.0014
LoReFT	2.3234 ± 0.0018	2.4997 ± 0.0060	2.4078 ± 0.0019	2.1342 ± 0.0103	1.4378 ± 0.0043	1.9701 ± 0.0045	3.2198 ± 0.0062	2.2847 ± 0.0019
PPO	2.8410 ± 0.1177	3.3859 ± 0.0948	2.8176 ± 0.0170	4.2054 ± 0.0509	1.9188 ± 0.0129	2.5210 ± 0.0548	5.0424 ± 0.3228	3.2474 ± 0.0659
GRPO	2.8446 ± 0.1051	3.4753 ± 0.1088	2.7793 ± 0.0204	4.2071 ± 0.0436	1.9458 ± 0.0113	2.4616 ± 0.0587	4.9763 ± 0.3670	3.2414 ± 0.0704
DPO	2.6881 ± 0.0111	3.6971 ± 0.1875	2.9749 ± 0.0036	4.1831 ± 0.0260	2.2758 ± 0.0604	2.5061 ± 0.0527	3.9091 ± 0.0142	3.1763 ± 0.0346

Table 8: General activation drift (relative RMS, percent) from the original model. Q/K/V are measured before RoPE; MLP is the complete 1,536-dimensional down-projection output before residual addition. Values average 28 layers and seven task models within seed, then report mean ± sample SD across three seeds.

	Task probes				Retention probes			
Method	Q	K	V	MLP	Q	K	V	MLP
LoRA	18.71 ± 0.39	18.57 ± 0.48	35.47 ± 0.76	48.02 ± 1.02	4.45 ± 0.14	4.54 ± 0.15	9.24 ± 0.31	12.62 ± 0.37
WindowLoRA	15.28 ± 0.06	14.68 ± 0.09	28.75 ± 0.24	39.04 ± 0.24	2.82 ± 0.06	2.86 ± 0.07	5.89 ± 0.15	7.85 ± 0.17
PiSSA	27.62 ± 0.05	28.14 ± 0.15	52.07 ± 0.08	69.52 ± 0.33	13.43 ± 0.52	13.69 ± 0.56	27.84 ± 1.08	38.60 ± 2.18
MiLoRA	24.02 ± 0.14	24.62 ± 0.24	45.76 ± 0.24	59.33 ± 0.23	12.84 ± 0.31	13.32 ± 0.38	27.02 ± 0.57	36.30 ± 0.67
PSOFT	19.56 ± 0.07	19.26 ± 0.08	36.18 ± 0.05	50.44 ± 0.04	9.86 ± 0.17	10.47 ± 0.15	19.15 ± 0.25	26.13 ± 0.28
LoReFT	12.14 ± 0.13	10.87 ± 0.11	20.48 ± 0.28	28.38 ± 0.16	0.38 ± 0.01	0.36 ± 0.00	0.65 ± 0.02	0.91 ± 0.02
PPO	8.65 ± 0.16	7.92 ± 0.12	15.31 ± 0.27	22.55 ± 0.93	0.83 ± 0.00	0.83 ± 0.00	1.71 ± 0.01	2.33 ± 0.01
GRPO	8.93 ± 0.19	8.17 ± 0.14	15.78 ± 0.30	23.32 ± 1.09	0.85 ± 0.01	0.86 ± 0.01	1.76 ± 0.01	2.41 ± 0.02
DPO	8.08 ± 0.18	7.37 ± 0.19	14.25 ± 0.38	20.68 ± 0.53	0.78 ± 0.01	0.79 ± 0.01	1.62 ± 0.02	2.21 ± 0.04

The policy methods also have an auxiliary length-normalized answer-choice evaluation. Relative to the source model, PPO, GRPO, and DPO improve SocialIQA from 47.27% to $58.33/58.33/58.72\%$, ARC-Easy from 72.27% to $83.85/83.46/81.90\%$, and ARC-Challenge from 41.02% to $55.99/55.34/54.69\%$. Because this evaluator was not run on the six SFT-based carriers, it is not used as a nine-method leaderboard.

D.2 Leading and residual activation drift

For each layer and component, we fit a PCA basis to source-model activations and split it at 90% explained energy. The leading subspace contains the smallest set of directions reaching this cutoff; the residual subspace is its orthogonal complement, not an exact functional null space. Figure 7 reports component-relative RMS drift in both parts, while Figure 8 reports the fraction of drift energy in the residual part. The fraction should be interpreted with the absolute magnitude because a stable share can coexist with large total drift.

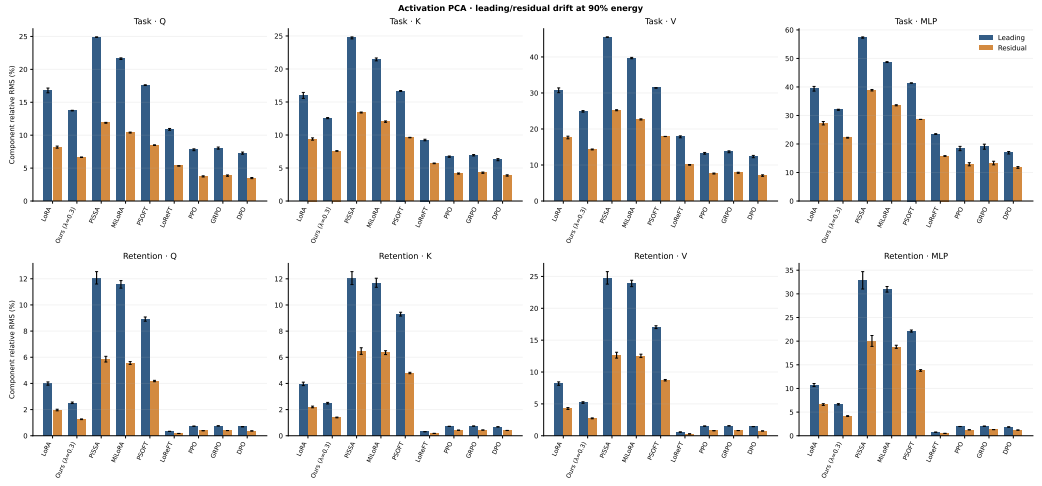


Figure 7: Qwen activation drift in leading and residual source-activation subspaces at a 90% source-energy cutoff. Columns show pre-RoPE $Q/K/V$ and the MLP down-projection output; rows show task and retention probes. Values average 28 layers and seven task adapters within each seed, followed by the mean and sample SD over three seeds.

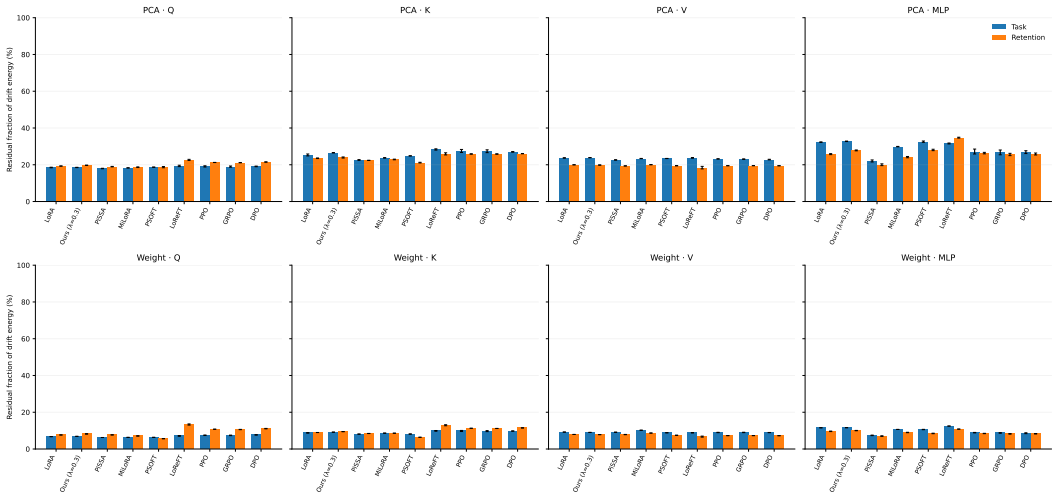


Figure 8: Fraction of activation-drift energy in the residual source subspace. Activation and weight subspaces are derived from their respective source representations.

D.3 Anchor strength and layer

The complete coefficient/layer study covers seven tasks and three seeds. Under the declared mean NLL increase of at most 1% and worst-task increase of at most 2%, it selects final-layer $\lambda = 0.3$ and middle-layer $\lambda = 3$. Both increase mean NLL by approximately 0.9%. The final-layer anchor leaves 12.24% of LoRA's final-state drift, whereas the middle-layer anchor leaves 46.65% at the final state and 7.65% at its own layer. Because state scales differ across layers, equal values of λ do not imply equal regularization strength.

A separate three-epoch BoolQ study gives the same qualitative pattern: anchoring lowers local hidden-state MSE from 0.5925 ± 0.1623 to 0.0260 ± 0.0057 and WikiText perplexity from 8.6823 ± 0.0591 to 8.5063 ± 0.0066, while validation NLL changes from 0.3767 ± 0.0491 to 0.3845 ± 0.0269. The paired NLL change varies in sign across seeds, so this is not evidence of a task-performance improvement.

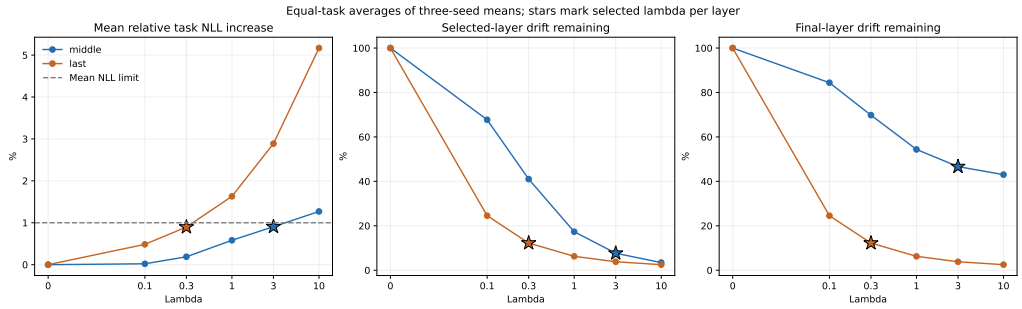


Figure 9: Coefficient sweep for middle- and final-layer anchors. Stars mark the configurations selected under the declared task-NLL budget.

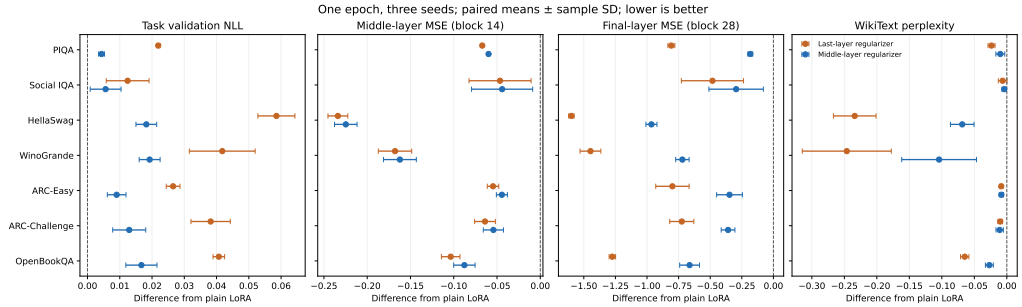


Figure 10: Per-task comparison of no anchoring and middle- or final-layer anchoring at $\lambda = 1$. Raw MSE is compared only within a layer because hidden-state scales differ.

D.4 Window width and anchor selection

Window-width and position-selection ablations use PIQA and HellaSwag with three seeds. Relative to $L/4$, the shorter $L/8$ window improves task NLL by 0.0159 but worsens mean retention perplexity by 0.0130. The longer $L/2$ window raises task NLL by 0.0146 and improves retention perplexity by only 0.0037. Full-vocabulary entropy selection improves task NLL by 0.0051 over fixed positions but worsens retention perplexity by 0.0242 and approximately doubles hidden-state drift. Top-10 entropy reduces drift relative to full entropy but still does not outperform fixed positions on retention. These two-task results support fixed $L/4$ windows as a practical compromise, not a universal optimum.

E Secondary Transfer and Policy Studies

E.1 Generative tasks

GSM8K and XSum use independent 1,024/256/256 train/validation/test subsets and three seeds. On GSM8K, all six adapted methods improve test NLL but reduce generated-answer accuracy relative to the source model. Compared with LoRA, WindowLoRA changes accuracy from 59.51% to 57.94% and mean retention perplexity from 12.4863 to 12.4406. On XSum, it changes ROUGE-L from 27.13 to 27.43 and retention perplexity from 12.5066 to 12.4678. LoReFT remains closest to source retention on both tasks.

E.2 Answer-choice policy pilot

The answer-choice pilot uses seed 17, 512 training prompts, and 256 held-out questions per task and is therefore separate from the primary matrix. Adding the hidden-state anchor reduces retention perplexity by 0.0105, 0.0239, and 0.0285 for SFT, PPO, and GRPO, respectively, while changing accuracy by -0.20, -1.17, and -1.37 percentage points. Training overhead is 1.75–$1.76\times$.

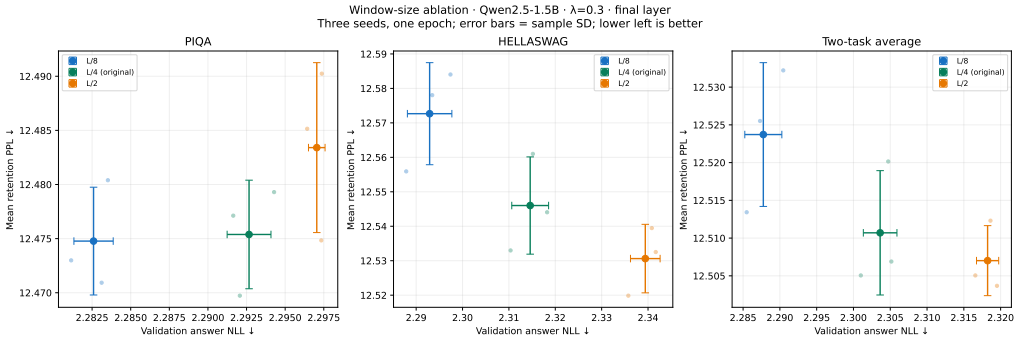


Figure 11: Window-width ablation on PIQA and HellaSwag. All configurations use one epoch, final-layer $\lambda = 0.3$, four fixed anchors, and three seeds.

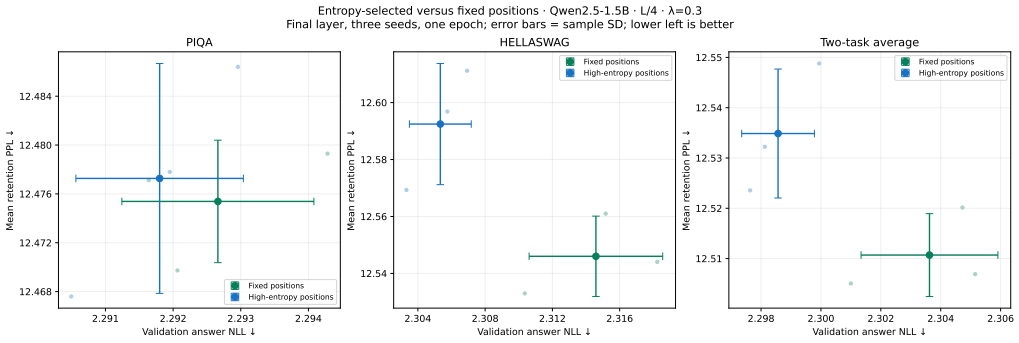


Figure 12: Fixed-quarter versus full-vocabulary entropy anchor selection. Lower-left is better; evaluation drift always uses fixed-quarter windows.

Bootstrap intervals reflect question sampling at one training seed and are not adjusted for multiple comparisons.

F Evidence Boundaries

The experiments establish distinct retention and drift profiles across update carriers, along with retention gains over matched LoRA at a small task cost. They do not establish that measured V or MLP drift causally determines forgetting, nor do fixed Qwen subsets constitute full-benchmark evaluation. Llama contributes a three-seed commonsense transfer and a single-seed math study, but not matched activation measurements or a controlled model-size comparison. Direct replay, DPO with anchoring, unrestricted RL generation, and activation-patching experiments remain open.

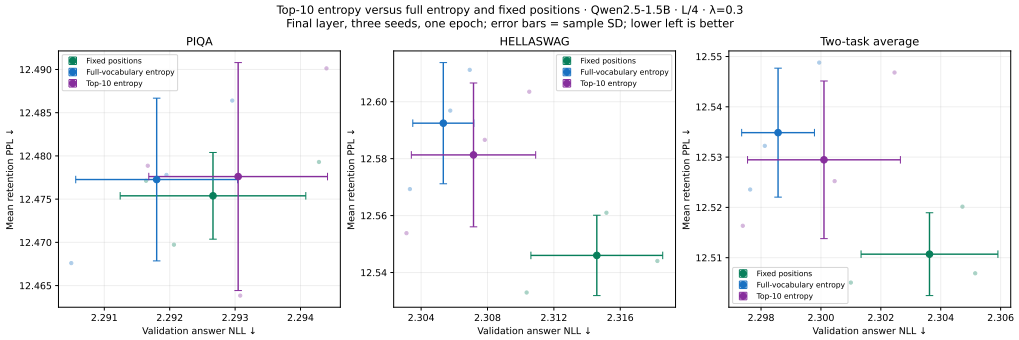


Figure 13: Fixed, full-vocabulary entropy, and top-10 entropy anchor selection. Top-10 improves on full entropy but not on fixed anchors.

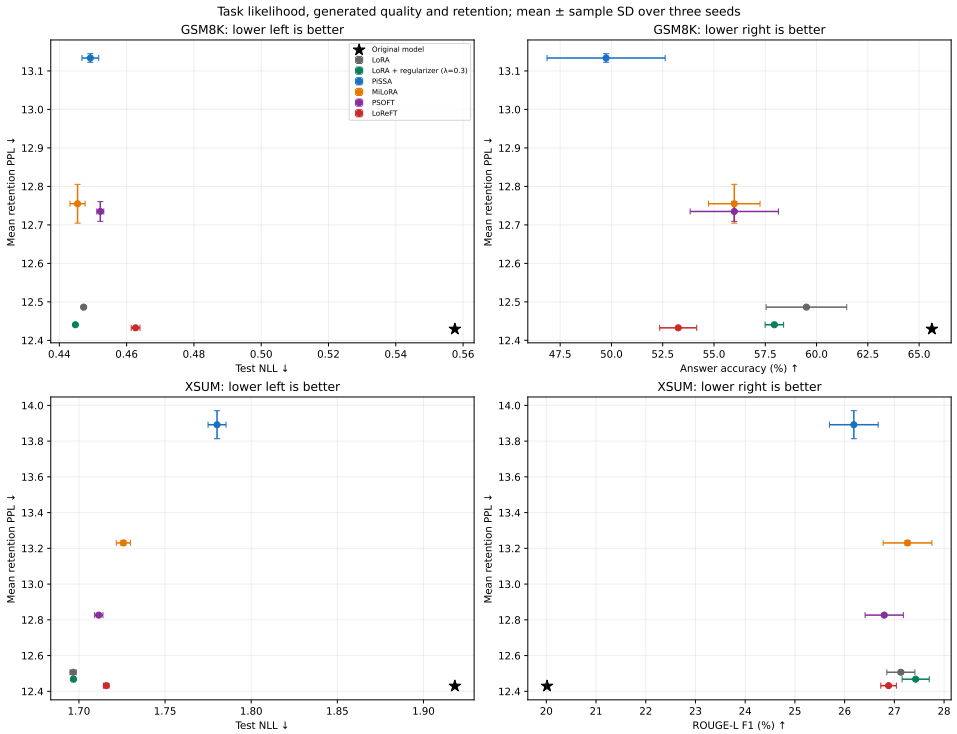


Figure 14: Generation quality, teacher-forced NLL, and retention on GSM8K and XSum. Each trained method uses three seeds.

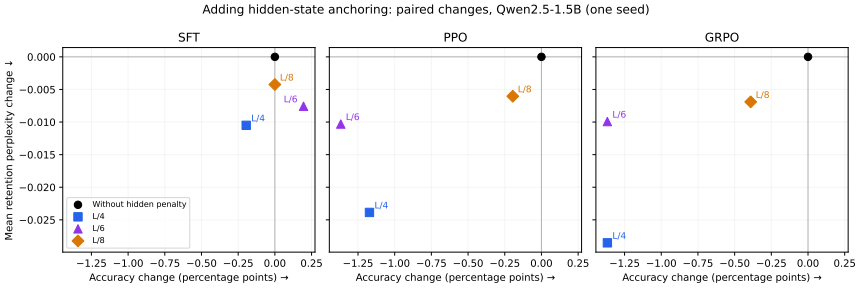


Figure 15: One-seed answer-choice SFT/PPO/GRPO pilot using $L/4$, $L/6$, and $L/8$ hidden-state anchors. This protocol differs from the primary matrix.