REASONING GAINS AND CAPABILITY STABILITY IN SHORT-HORIZON GRPO: A PAIRED KL AUDIT

Anonymous authors
Paper under double-blind review

ABSTRACT

Does a reference-policy KL penalty protect capabilities during reasoning reinforcement learning, once target-task learning and training exposure are measured together? We compare six 300-update LoRA runs of Qwen2.5-1.5B-Instruct, pairing three seeds across KL coefficients zero and 0.02. Greedy accuracy on 300 held-out GSM8K questions rises from 58.7% to 68.7–72.3%. Meanwhile, likelihood accuracy remains at 77.7–78.2% on 600 BoolQ questions (77.8% initially) and 73.2–74.6% on 299 ARC-Challenge questions (74.6% initially). The paired KL effects on final BoolQ change are $+0.17$, -0.33, and 0.00 percentage points. Thus this coefficient has no consistent protective effect across the three seeds. Item-level predictions distinguish aggregate stability from preservation of individual decisions: two KL runs finish at the initial BoolQ accuracy despite four or five gains canceling the same number of losses. An exact enumeration of compatible pairings shows how much statistical evidence is lost when only accuracy totals are retained. We also specify why a penalty measured on arithmetic responses does not bound changes on another prompt distribution. The evidence supports substantial arithmetic learning with small changes on two likelihood tasks in this short-horizon setting, while leaving broader preservation and stronger KL interventions unresolved.

1 INTRODUCTION

Reasoning reinforcement learning changes a model through a narrow reward signal, while deployment often relies on behavior outside that reward's task. GRPO (Shao et al., 2024) and reasoning-model reports (Guo et al., 2025) make this tension practically relevant. A capability monitor can detect a change, but its interpretation depends on the task, the scoring rule, and the training intervention. Flat aggregate scores can hide item turnover; small declines can reflect only a few changed decisions. Neither observation alone identifies a general preservation or forgetting mechanism.

We study this question through a paired intervention on one open-weight model. Six runs use three seeds and two KL coefficients, with 300 updates per run. Each checkpoint is assessed on generated arithmetic answers, binary reading likelihood, and multiple-choice science likelihood. The experiment produces 7,200 training completions and evaluates the same questions at five checkpoints. The arithmetic gain distinguishes this setting from an adaptation that barely changes measured target-task performance. The other-task results nevertheless remain small and do not consistently favor either penalty setting.

The study links three quantities that are often reported separately: target-task learning, the between-penalty contrast, and item-level gains and losses. Matching update exposure makes the first two comparable; retaining paired predictions makes the third observable. We then enumerate the pairings compatible with each pair of accuracy totals to quantify the evidence that would be lost by reporting margins alone. The overlap bounds are elementary; their role is to make the measurement consequence explicit in completed, paired reasoning-RL runs.

Phan et al. (2026) investigate broader general-capability forgetting and introduce replay with dynamic objective reweighting. We do not compare against that intervention. Our question is whether one reference penalty produces a consistent retention effect under a fixed, effective short-horizon

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.

adaptation. A penalty evaluated on arithmetic rollouts is not itself a guarantee about another prompt distribution. This distinction motivates reporting its measured effect, rather than treating the presence of a KL term as evidence of protection.

2 Related work and the scope of retention claims

Continual learning and task interference. Catastrophic forgetting is a change in performance on previously supported tasks as training adapts a model to a new objective. Elastic weight consolidation constrains important parameters (Kirkpatrick et al., 2017), learning without forgetting preserves earlier predictions through distillation (Li & Hoiem, 2018), and gradient episodic memory uses stored examples to restrict interfering updates (Lopez-Paz & Ranzato, 2017). These methods supply different information about past tasks. A penalty estimated only on arithmetic rollouts does not supply that same information. Consequently, our KL comparison is not a test of whether replay or task-aware continual-learning methods can prevent forgetting.

Adaptation capacity and learning exposure. LoRA parameterizes a matrix update with low-rank factors (Hu et al., 2022). Biderman et al. (2024) compare its learning and retention behavior with full fine-tuning, making adaptation strength an important part of a retention comparison. A smaller measured decline can accompany less learning on the new task. We therefore report arithmetic change together with the other-task monitor and keep the update count fixed across penalty settings. Rank alone cannot establish protection: a low-rank matrix can have arbitrarily large norm, and its two learned factors do not fix a global update subspace.

Reference penalties and policy optimization. Reference-policy penalties appear in instruction-following training from human feedback (Ouyang et al., 2022). Trust-region policy optimization (Schulman et al., 2015) imposes a different optimization structure from a sampled soft penalty, while PPO uses a clipped surrogate (Schulman et al., 2017). Our experiment uses one on-policy update for each new group, so neither a multi-epoch PPO guarantee nor a hard KL constraint applies. We use AdamW with zero weight decay; decoupled weight decay is a distinct mechanism from the reference penalty (Loshchilov & Hutter, 2019). This distinction prevents attributing an effect to "regularization" without identifying which term is active.

What capability evaluations cover. BoolQ assesses natural yes/no reading questions (Clark et al., 2019); GSM8K assesses arithmetic word problems (Cobbe et al., 2021). Their complementarity does not make them a complete capability inventory. ARC offers science questions (Clark et al., 2018), MMLU spans many subject areas (Hendrycks et al., 2021), and HELM emphasizes evaluating models across scenarios and metrics (Liang et al., 2023). These works illustrate the distinction between a well-defined task score and a broad claim about general knowledge. Our conclusions refer to the measured tasks and protocols, not to all capabilities represented by those suites.

Instruction-following evaluation can operationalize constraints with explicit verifiers (Zhou et al., 2023); a loose word-count or keyword heuristic does not necessarily verify the instruction it names. Safety is a further distinct axis. Fine-tuning can affect safety alignment even without malicious training intent (Qi et al., 2024), but a few refusal keywords do not measure that phenomenon adequately. The present reading and science evaluations do not measure safety preservation or instruction compliance.

Replication and paired comparisons. Training variability and evaluation uncertainty require different sampling units (Henderson et al., 2018; Agarwal et al., 2021). Statistical-testing guidance in NLP (Dror et al., 2018) is especially relevant when the same questions are scored before and after adaptation. A problem-level bootstrap (Efron, 1979) preserves those paired observations; it does not turn three training seeds into hundreds of independent training replications. McNemar's paired comparison (McNemar, 1947) depends on which questions change correctness, not just the two accuracy totals. The missing-discordance analysis below formalizes why aggregate monitor counts cannot recover that evidence.

3 PAIRED TRAINING AND EVALUATION

3.1 INTERVENTION AND DATA SEPARATION

Every run starts afresh from Qwen2.5-1.5B-Instruct (Qwen et al., 2024). We freeze base weights and train rank-eight LoRA adapters on query and value projections, with scaling factor 16 and zero adapter dropout. The KL coefficient is either $\beta = 0$ or 0.02. Three seeds, 20260921–20260923, define paired runs with identical prompt schedules and generation-seed schedules. Training uses a hash-selected set of 512 GSM8K training questions; evaluation uses disjoint benchmark splits. The pairing controls initialization and schedules, but policies can produce different rollouts after their updates diverge.

Each update samples four completions for one prompt. Full-distribution sampling uses temperature one, top-p one, top-k zero, repetition penalty one, and a 512-token output ceiling. We use the native chat template and request a boxed numerical answer. A label-independent parser extracts the last balanced boxed expression, falling back to an explicit answer field. Exact rational normalization merges equivalent fractions and decimals. Unparsed outputs receive zero reward; the checker does not verify every reasoning step or arbitrary symbolic equivalence. Training and evaluation use the same normalization rule. The same reward implementation is used from the first update in every run.

3.2 IMPLEMENTED UPDATE

The reward advantages subtract the four-sample mean and divide by the population standard deviation plus 10^{-8}. Unanimous groups have zero advantages. For completion y_j of length T_j, the token-averaged loss is

$$\mathcal{L}(\theta) = \frac{1}{4}\sum_{j=1}^{4}\frac{1}{T_j}\sum_{t=1}^{T_j}\left[-A_j \log \pi_\theta(y_{jt} \mid x, y_{j,<t}) + \beta k_{jt}\right], \tag{1}$$

$$k_{jt} = \exp(d_{jt}) - d_{jt} - 1, \qquad d_{jt} = \log \pi_{\text{ref}}(y_{jt} \mid x, y_{j,<t}) - \log \pi_\theta(y_{jt} \mid x, y_{j,<t}). \tag{2}$$

The reference disables adapters. We apply one on-policy update per fresh group, using AdamW with learning rate 10^{-5}, zero weight decay, and gradient-norm clipping at one. This implementation is a group-relative score-function update with a sampled reference penalty. It is not a multi-epoch clipped-PPO experiment, and the penalty is not a hard trust-region constraint. Token averaging also differs from a sequence-sum gradient; normalization choices can matter in reasoning RL (Liu et al., 2025).

All runs complete 300 updates. The number of mixed-reward groups is 200, 204, 195, 202, 196, and 201 in seed-major, penalty-minor order. Thus 65.0–68.0% of updates have a nonzero reward contrast before possible cancellation in parameter space. Each run samples 1,200 completions, for 7,200 in total. Equal update and completion counts do not imply equal realized token cost.

3.3 THREE EVALUATION ENDPOINTS

We evaluate at updates 0, 50, 100, 200, and 300 on fixed item sets. GSM8K uses 300 test questions, greedy generation, and the numerical checker above. BoolQ uses 600 validation questions and selects Yes or No by conditional log likelihood. We limit the passage to 768 tokens while preserving the question. ARC-Challenge uses all 299 validation questions and selects among the supplied answer labels by conditional log likelihood. These are three distinct endpoint protocols: generated arithmetic correctness, constrained reading decisions, and constrained science decisions. Their values are not interchangeable with scores from another prompting or grading implementation.

Every evaluation retains per-item predictions and labels, and the two likelihood tasks retain option scores. The initialized adapter's effective update is zero; identical baseline vectors are reused only after checking the evaluation IDs and zero adapter output. Baseline accuracy is 176/300 for GSM8K, 467/600 for BoolQ, and 223/299 for ARC-Challenge. None of the test or validation labels enters the training update. These repeated evaluations measure trajectories on a fixed monitor set; they are not independent test sets for choosing a best checkpoint.

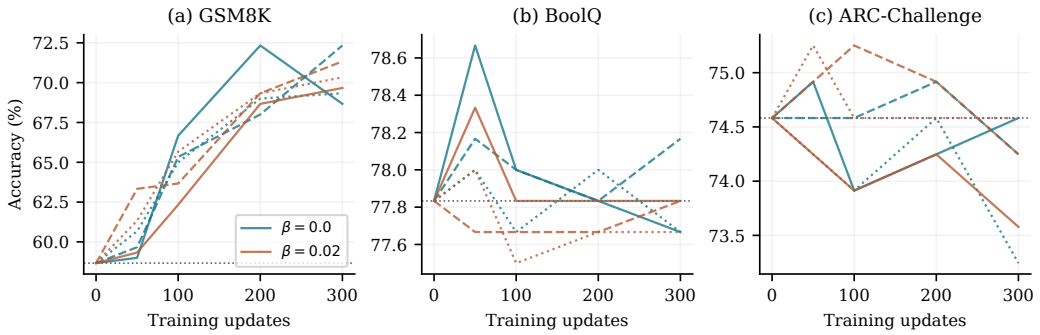

Figure 1: All six training trajectories. Color denotes the KL coefficient; solid, dashed, and dotted lines denote the three seeds. Horizontal dotted lines mark baseline accuracy. These are individual runs, not confidence bands. The tasks use different evaluation protocols and sample sizes.

Table 1: All 300-update endpoints. Counts use the same questions in each run. The baseline row is shared. Seed suffixes identify seeds 20260921–20260923.

Seed	β	GSM8K / 300	BoolQ / 600	ARC / 299
Baseline	—	176	467	223
21	0	206	466	223
21	0.02	209	467	220
22	0	217	469	222
22	0.02	214	467	222
23	0	208	466	219
23	0.02	211	466	223

4 Final outcomes and training trajectories

4.1 Target-task learning is substantial in this setting

Every run improves arithmetic accuracy from 58.7% to 68.7–72.3%, gaining 30–41 net correct answers on the fixed 300-question set. Figure 1 shows that the gain develops over training, rather than being inferred from a single changed item. The retention measurements therefore accompany a 10.0–13.7-point improvement on held-out arithmetic questions. They do not describe an intervention that leaves its target-task score essentially unchanged.

At update 300, the KL-minus-no-KL arithmetic differences are $+1$, -1, and $+1$ percentage points. The mean is $+0.33$ points, with inconsistent direction across seeds. Earlier checkpoint contrasts also change sign: their mean is $+1.56$ at update 50, -1.78 at 100, and -0.67 at 200. These are repeated measurements of the same six runs, not four independent studies. Choosing the checkpoint with the most favorable penalty contrast would change the intended fixed-endpoint comparison.

4.2 The primary retention contrast does not consistently favor KL

The primary endpoint is the between-penalty difference in before-to-after BoolQ accuracy at update 300, paired by seed. The three effects are $+1/600$, $-2/600$, and $0/600$, or $+0.17$, -0.33, and 0.00 percentage points. Their mean is -0.056 points. Table 1 shows that the underlying final scores differ from baseline by at most two answers. The experiment does not establish a consistently protective effect of this coefficient. It also does not establish equivalence: a non-inferiority margin and a powered preservation design were not specified.

ARC is a secondary endpoint. Its KL contrasts are $-3/299$, $0/299$, and $+4/299$, averaging $+0.11$ percentage points. One seed favors no KL, one ties, and one favors KL. The small reading changes and these science results do not support a broad catastrophic-forgetting claim either. The supported observation is coexistence of arithmetic improvement with small changes on two particular likelihood monitors over this training horizon.

Table 2: Training diagnostics over all 300 updates. The logged penalty is the sampled token-averaged quantity k_{jt}, not an exact sequence-level KL or an other-task divergence. Token counts include all four sampled responses per update.

Seed	β	Generated tokens	Mean penalty	Mixed groups
21	0	260715	0.00164	200/300
21	0.02	257478	0.00135	204/300
22	0	256910	0.00118	195/300
22	0.02	255319	0.00113	202/300
23	0	254967	0.00107	196/300
23	0.02	252775	0.00141	201/300

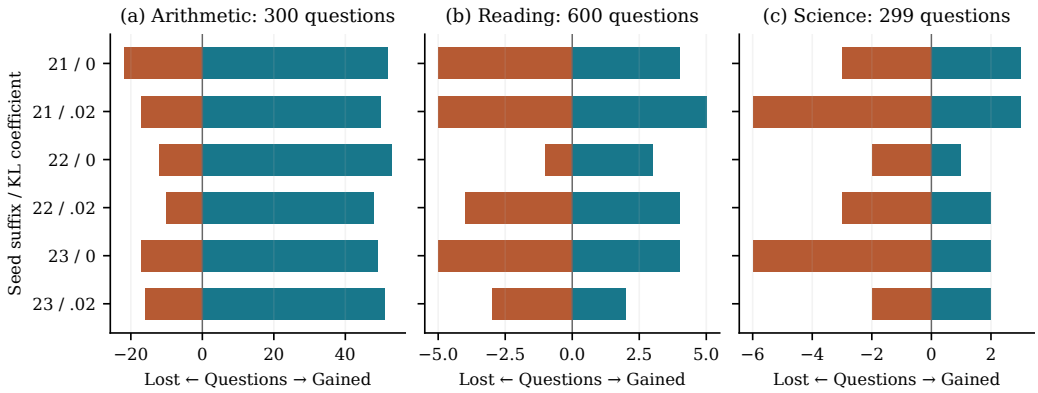

Figure 2: Correctness transitions between initialization and update 300. Gains are previously incorrect items answered correctly after training; losses are previously correct items answered incorrectly. Net accuracy subtracts losses from gains and can hide both. No seed–question cells are treated as independent training replications.

4.3 The logged penalty does not show uniform separation

The mean sampled penalty is lower with KL in the first two seed pairs but higher in the third. The corresponding last-50-update averages are also not uniformly ordered. This diagnostic matters for interpreting the intervention: choosing a nonzero coefficient does not demonstrate a consistently separated functional-divergence regime in these runs. The policy changes its sampled contexts and completions as it learns, so the averages are not evaluated on one common fixed distribution. They cannot establish which policy is closer to the reference on every prompt.

The pre-clipping gradient norm exceeds one in only one of 1,800 updates. Clipping therefore rarely changes these updates, but the small number of interventions does not identify a causal role for clipping. Generated training tokens range from 252,775 to 260,715 despite equal completion counts. These measurements delimit the regime actually tested: short adaptation with a small logged penalty and effective arithmetic learning. Extrapolation to longer training, larger coefficients, or larger models remains unsupported.

4.4 Flat scores can conceal changed decisions

Let g and ℓ be the counts of gained and lost questions. Net accuracy change is $(g - \ell)/n$, whereas correctness turnover is $(g + \ell)/n$. The first no-KL arithmetic run gains 52 questions and loses 22: its ten-point net improvement accompanies changes on 74/300 questions. Across the six runs, arithmetic turnover ranges from 58/300 to 74/300, or 19.3–24.7%. An improved aggregate score does not mean every previously correct item remains correct, and turnover is not necessarily harmful when gains exceed losses.

On BoolQ, the first and second KL runs both end at baseline accuracy but each changes decisions: five gains cancel five losses in the first, and four cancel four in the second. BoolQ turnover ranges from four to ten questions across runs. On ARC, the first no-KL run gains and loses three each, while the final KL run gains and loses two each. Both have unchanged aggregate accuracy. Figure 2 reports these distinctions directly, without assigning a general capability label to an individual gain or loss.

5 Estimands and the limits of aggregate monitors

Let $c_{s,\beta,t,i}$ be binary correctness for seed s, penalty β, checkpoint t, and item i. Define

$$d_{s,\beta,t} = \frac{1}{n}\sum_i (c_{s,\beta,t,i} - c_{s,\beta,0,i}), \qquad \tau_{s,t} = d_{s,0.02,t} - d_{s,0,t}.$$

The primary replication unit for τ is a training seed. A question bootstrap conditional on trained models probes evaluation-set composition, not training randomness. Pooling seed–question cells as independent would ignore shared questions across seeds and shared weights across questions. We report all three seed contrasts rather than giving a misleadingly precise pooled test.

The item-level results also allow a direct comparison between paired evidence and what could be inferred from reporting only accuracy totals. We use the current runs for that calculation, retaining the observed transitions while enumerating the alternative pairings compatible with each pair of totals.

6 What accuracy totals leave unidentified

Let a and b be baseline and endpoint successes on the same n items. The number h correct at both endpoints can take any integer in

$$\max(0, a + b - n) \le h \le \min(a, b).$$

For each compatible overlap, the discordant counts are $a - h$ and $b - h$. A two-sided exact McNemar/binomial test uses these discordances. Without their observed pairing, there is a range of possible p-values, not a single test.

For the first no-KL run at update 300, arithmetic successes increase from 176 to 206 out of 300. Compatible pairings yield exact p-values from 1.86×10^{-9} to 0.0493. The actual pairing has 52 gains and 22 losses, so it identifies one value within that range. For the same run's BoolQ scores at update 50, the totals are 467 and 472 out of 600; the possible range is 0.0625–0.8045, with six observed gains and one loss. For the third no-KL run's final ARC scores, 223 and 219 out of 299, the possible range is 0.125–0.8103. These examples use measured totals, but the enumerated pairings are hypothetical. They quantify information lost by retaining only margins.

The comparison concerns a fixed pair of checkpoints. It does not turn a question-level test into evidence from independent training replications. Even when every compatible pairing lies on one side of a nominal threshold, the effect's size, its uncertainty across seeds, and its relevance to capability preservation remain separate questions. A null test also differs from a non-inferiority or equivalence design, which requires a specified tolerance and enough observations to resolve it.

Why the range is sharp. For any integer overlap h in the stated interval, construct a 2×2 correctness table with entries h, $a - h$, $b - h$, and $n - a - b + h$. Every entry is nonnegative and the margins are exactly a and b. Conversely, any paired table with those margins must have that form. Enumerating the interval therefore exhausts all possible pairings; no narrower test-value range follows from the margins alone. Figure 3 displays this identified set. Its width reflects missing joint observations, which additional resampling of the same totals cannot recover.

What the observed pairings add. The arithmetic example has 74 discordant items, not merely 30 net gains. The paired data locate its uncertainty within the range allowed by the margins. For BoolQ, a small net change can arise from very few transitions, so a task-level confidence statement must retain their frequency. This information is available from the same evaluations: it requires saving predictions, not another training run. Its value is separate from task coverage, which still limits any claim about broader capabilities.

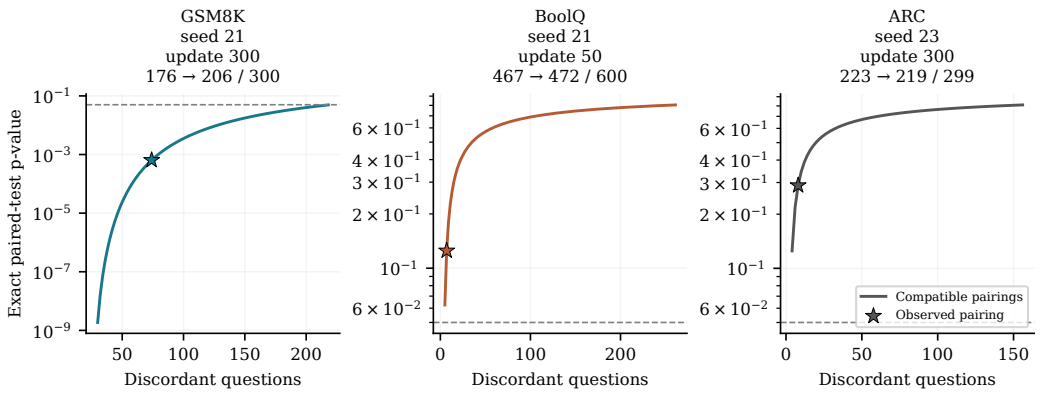

Figure 3: Information lost when paired predictions are reduced to marginal success counts. Curves enumerate every compatible pairing for three examples from the current runs; stars locate the observed pairing. Labels identify the seed suffix and update. The horizontal line marks 0.05. These are conditional paired-test values, not intervals over training seeds.

6.1 Design consequences of paired measurement

The observed gain–loss decomposition also determines the sampling variance of a paired accuracy difference. If independent evaluation questions have probabilities p_g and p_ℓ of a gain and loss, the per-question difference Z takes values 1, -1, and 0. Hence

$$\mathbb{E}Z = p_g - p_\ell, \qquad \mathrm{Var}(Z) = p_g + p_\ell - (p_g - p_\ell)^2.$$

For n IID questions, the variance of their mean is this expression divided by n. Accuracy change determines the difference $p_g - p_\ell$, but not the sum $p_g + p_\ell$. This explains why marginal accuracy totals are insufficient for a precise paired analysis even when the net decline is known exactly. The IID calculation is a design model, not a guarantee for a hand-selected monitor or for correlated seed–question observations.

An approximately flat score can therefore have low or high paired uncertainty depending on turnover. If almost no items change, the paired differences are mostly zero. If many gains cancel many losses, the mean can still be near zero while its sampling variance is larger. Increasing the number of checkpoints on the same models does not automatically reduce either uncertainty: those measurements share training history and questions. A time-course plot is useful for locating changes, but selecting its largest decline and applying a single-checkpoint interval would ignore the selection step.

What a preservation claim would require. A practically meaningful tolerance must be chosen in the units of a specific endpoint. For example, a margin in reading-question accuracy does not translate automatically into a margin in generated arithmetic accuracy, code execution, or safety behavior. A non-inferiority analysis asks whether a suitable lower confidence bound on change exceeds the negative tolerance, while an equivalence analysis asks whether an appropriate interval lies inside a two-sided band. Neither follows from a test that fails to reject zero change. Training-seed variability must also enter a claim about the training procedure, rather than only the two evaluated checkpoints.

Our short-horizon results do not implement that design. The individual seed changes and paired transition counts remain useful descriptive outputs: they state exactly how many monitored decisions changed, under which scoring rule, and after how much training. They also expose where a larger evaluation would be most informative. A small interval around an invalid proxy would still not establish the capability named by that proxy, so measurement validation precedes a power calculation for a broad capability claim.

Separating training exposure from evaluation. The six runs compare penalty settings at the same number of updates, with the same prompt schedule and sampled completions per update. They therefore hold training exposure fixed in these units. Generated token counts and realized policy changes still differ. A comparison at equal achieved arithmetic accuracy would answer a different

Table 3: Interpretation depends on the evaluation endpoint. These distinctions apply to both positive and negative changes; no single row measures general capability retention.

Endpoint	Directly measured	Not established by a flat score
Generated arithmetic	Numeric-answer correctness under fixed decoding	Unchanged reasoning strategies or broad mathematical competence
Reading likelihood	Relative probability of two allowed labels	Unchanged free-form responses or general knowledge
Science likelihood	Relative probability of supplied answer labels	Unchanged free-form scientific explanations
Sampled reference penalty	Divergence diagnostic on training responses	Proximity on another prompt distribution
Paired correctness	Gains and losses on the same items	Stability on unmeasured questions or across training seeds

question, because it could require different checkpoints or additional training in each condition. Our fixed-endpoint comparison avoids choosing those checkpoints after inspecting retention.

The evaluation also holds questions and scoring rules fixed. BoolQ and ARC compare probabilities of supplied answer labels; GSM8K scores a generated numerical answer. Stable label rankings can coexist with changes in the underlying probabilities or in free-form responses. Conversely, a small probability change can flip a near-tied label ranking. Reported accuracy therefore describes the decisions made by each evaluation protocol, rather than the entire conditional response distribution.

7 Interpreting regularization and capability change

7.1 What the two-coefficient comparison identifies

Within each seed, the two runs share initialization and the exogenous prompt and generation-seed schedules. The intended contrast changes only β. After the first differing update, however, responses, rewards, and gradients can diverge. This is part of the total effect of the training intervention; forcing later responses to remain identical would define another algorithm. Pairing reduces variation from the shared setup without removing stochastic variation along the subsequent training paths.

The three paired seed differences estimate behavior at two coefficient values and one learning rate. They do not identify a dose–response curve or a KL-by-learning-rate interaction. Such an interaction requires multiple learning rates at matched exposure and a contrast comparing their penalty effects. The small observed differences instead motivate measuring stronger divergence separation before attributing capability stability to the reference penalty.

7.2 Regularization is not a hard trust-region guarantee

A KL penalty encourages reference-policy proximity but does not by itself bound each update's KL divergence by η/β. Such a bound needs additional assumptions and an appropriate optimization argument. Similarly, GRPO's behavior-policy importance ratio and its fixed reference-policy KL anchor are conceptually distinct; the two policies must not be conflated when documenting an implementation.

LoRA constrains each matrix update's rank, not its norm. Jointly trained factors B, A can change the update subspace; rank r does not mean a fixed global r-dimensional parameter space. Factor rescaling can change parameter-gradient norms without changing the effective matrix update. Therefore raw gradient norms from differently parameterized updates are not directly comparable measures of functional drift.

Finally, average sampled KL and entropy estimates depend on their sampling distribution and estimator. Small logged values do not prove proximity on all capability prompts, and low entropy is not by itself a causal explanation for forgetting. Interventions and function-space measurements would be needed to test these hypotheses. The observed runs cannot explain discrepancies with larger-scale RL studies that differ in tasks, duration, and implementation.

7.3 When a reference bound can transfer across prompts

The distinction between a prompt-distribution average and guarantees on other tasks can be stated without assuming a particular optimizer. Suppose training prompts follow P and a capability evaluation follows Q. A small value of

$$\mathbb{E}_{x\sim P}\big[D_{\mathrm{KL}}\big(\pi_\theta(\cdot \mid x)\|\pi_{\mathrm{ref}}(\cdot \mid x)\big)\big]$$

places no bound on conditional policies at prompts outside the support of P. For example, on a two-prompt domain, let P concentrate on the first prompt and Q on the second. Two policies can agree exactly at the training prompt and assign arbitrarily different answer probabilities at the evaluation prompt. The displayed average is then zero regardless of the evaluation change. This is a counterexample to a distribution-free implication, not a claim that neural-network updates realize arbitrary independent prompt policies.

A sufficient coverage condition makes the missing assumption explicit. Write $K(x) = D_{\mathrm{KL}}(\pi_\theta(\cdot \mid x)\|\pi_{\mathrm{ref}}(\cdot \mid x))$. If $Q \ll P$ and $dQ/dP \le C$ almost surely, then nonnegativity gives

$$\mathbb{E}_Q K(x) = \mathbb{E}_P\left[\frac{dQ}{dP}(x)K(x)\right] \le C\,\mathbb{E}_P K(x).$$

This change-of-measure bound is useful only when the coverage constant C and the training expectation are controlled. Neither is certified here. Moreover, our measured penalty is sampled and token-averaged, whereas $K(x)$ is exact sequence-level KL. Applying this sufficient condition to the implemented loss would require both a coverage argument and an estimator analysis; a small logged penalty alone supplies neither.

This observation clarifies what the experiment can test. Its measured penalty contrast is an empirical property of the chosen training and evaluation protocols. It is not a proof that a particular β constrains arbitrary capability loss. Low-rank adaptation, small update norms, and a small sampled penalty may all be useful diagnostics, but none alone supplies the missing relationship between the training and evaluation distributions.

8 Limitations

The paired training comparison fixes update counts, seeds, and item-level outputs, but covers one 1.5B model and 300 updates. Its three evaluation protocols do not measure broad capability preservation. All paired predictions are available here, but the aggregate-count analysis shows why reporting only accuracy would discard useful information. Only two penalty coefficients and one learning rate are tested; the experiment does not identify a KL-by-learning-rate interaction. These restrictions limit both positive and negative conclusions: flat scores cannot establish equivalence, while isolated additional errors cannot establish catastrophic forgetting.

Generality of the measurement argument. The overlap bounds and gain–loss identities apply to any pair of binary correctness vectors, independently of the training algorithm. The empirical results are narrower: they concern the specified Qwen checkpoint, adapters, training schedules, and monitors. This separation matters when applying the audit elsewhere. A larger model or a stronger reasoning intervention may produce a different retention pattern while still requiring the same paired evidence to interpret it. The gain–loss distinction remains relevant across settings, but its generality does not expand this experiment's task coverage or training replication.

9 Conclusion

Across six paired 300-update runs, arithmetic accuracy improves by 10.0–13.7 percentage points while reading and science likelihood scores change little. The KL contrasts vary across seeds and do not establish a consistent protective effect. Paired outputs reveal the item gains and losses hidden by aggregate scores, while the overlap analysis characterizes what such totals alone can support. Short-horizon arithmetic adaptation in this setting changes target-task accuracy much more than the two monitored likelihood decisions. Broader preservation claims require additional tasks, training regimes, and a specified tolerance for acceptable capability loss.

Reproducibility Statement

The supplement provides the data and code needed to reproduce the reported analyses, following Pineau et al. (2021). The paired-training section specifies the train/evaluation split, adapter configuration, loss, update count, sampling distribution, and evaluation endpoints. The results table reports every seed and penalty setting in that study. The measurement analysis identifies where aggregate monitor counts cannot recover paired outcomes or validate a capability measure.

AI Use Statement

OpenAI Codex was used to aid and polish manuscript writing and to assist with literature retrieval and discovery, including finding related work. The authors are responsible for the final manuscript and citations.

References

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc Bellemare. Deep Reinforcement Learning at the Edge of the Statistical Precipice. In *Advances in Neural Information Processing Systems*, volume 34, pp. 29304–29320. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/hash/f514cec81cb148559cf475e7426eed5e-Abstract.html.

Dan Biderman, Jacob Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, Cody Blakeney, and John P. Cunningham. LoRA Learns Less and Forgets Less. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=aloEru2qCG.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of NAACL-HLT*, pp. 2924–2936, 2019. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300/.

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. *arXiv preprint arXiv:1803.05457*, 2018. URL https://arxiv.org/abs/1803.05457.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. URL https://arxiv.org/abs/2110.14168.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. The Hitchhiker's Guide to Testing Statistical Significance in Natural Language Processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1383–1392, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1128. URL https://aclanthology.org/P18-1128/.

B. Efron. Bootstrap Methods: Another Look at the Jackknife. *The Annals of Statistics*, 7(1):1–26, 1979. doi: 10.1214/aos/1176344552. URL https://doi.org/10.1214/aos/1176344552.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645:633–638, 2025. doi: 10.1038/s41586-025-09422-z. URL https://www.nature.com/articles/s41586-025-09422-z.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep Reinforcement Learning That Matters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018. doi: 10.1609/aaai.v32i1.11694. URL https://ojs.aaai.org/index.php/AAAI/article/view/11694.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring Massive Multitask Language Understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

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.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114. URL https://doi.org/10.1073/pnas.1611835114.

Zhizhong Li and Derek Hoiem. Learning without Forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2018. doi: 10.1109/TPAMI.2017.2773081. URL https://doi.org/10.1109/TPAMI.2017.2773081.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, et al. Holistic Evaluation of Language Models. *Transactions on Machine Learning Research*, 2023. URL https://openreview.net/forum?id=iO4LZibEqW.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-Like Training: A Critical Perspective. In *Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=5PAF7PAY2Y.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient Episodic Memory for Continual Learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/hash/f87522788a2be2d171666752f97ddebb-Abstract.html.

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

Quinn McNemar. Note on the Sampling Error of the Difference Between Correlated Proportions or Percentages. *Psychometrika*, 12(2):153–157, 1947. doi: 10.1007/BF02295996. URL https://doi.org/10.1007/BF02295996.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll 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 F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022. doi: 10.52202/068431-2011. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html.

Hoang Phan, Xianjun Yang, Yuanshun Yao, Jingyu Zhang, Shengjie Bi, Xiaocheng Tang, Madian Khabsa, Lijuan Liu, and Deren Lei. Beyond reasoning gains: Mitigating general-capability forgetting in large reasoning models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 34368–34398, 2026. doi: 10.18653/v1/2026.findings-acl.1717. URL https://aclanthology.org/2026.findings-acl.1717/.

Joelle Pineau, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Lariviere, Alina Beygelzimer, Florence d'Alche Buc, Emily Fox, and Hugo Larochelle. Improving reproducibility in machine learning research (a report from the neurips 2019 reproducibility program). *Journal of Machine Learning Research*, 22(164):1–20, 2021. URL https://jmlr.org/papers/v22/20-303.html.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To! In *International Conference on Learning Representations*, volume 2024, pp. 30988–31043, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/83b7da3ed13f06c13ce82235c8eedf35-Abstract-Conference.html.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. URL https://arxiv.org/abs/2412.15115.

John Schulman, Sergey Levine, Philipp Moritz, Michael Jordan, and Pieter Abbeel. Trust Region Policy Optimization. In *International Conference on Machine Learning*, volume 37, pp. 1889–1897, 2015. URL https://proceedings.mlr.press/v37/schulman15.html.

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.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-Following Evaluation for Large Language Models. *arXiv preprint arXiv:2311.07911*, 2023. URL https://arxiv.org/abs/2311.07911.