The GRPO Tax is Smaller Than You Think: A Longitudinal Study of Capability Preservation During Reasoning Training

Anonymous authors
Paper under double-blind review

Abstract

Reinforcement learning methods such as Group Relative Policy Optimization (GRPO) have emerged as a dominant paradigm for training language models to reason, following the success of DeepSeek-R1. A persistent concern is the "alignment tax": the degradation of general-purpose capabilities that accompanies post-training optimization for a specific objective. We conduct the first dense-checkpoint longitudinal study of capability evolution during GRPO training across five instruction-tuned models from four families (Qwen, Phi, Gemma, and Llama) at the 1.5B to 3.8B parameter scale, with a supplemental 7B QLoRA extension. Each model is evaluated at up to 75 training checkpoints across 13 benchmarks. Under single-epoch LoRA-based training, we establish that the GRPO tax is substantially smaller than commonly assumed: 85% of non-target capabilities remain within ±2% of their baseline values, 8% improve, and only 7% degrade. Six supplemental analyses constrain and validate this finding. Three-seed verification on Qwen2.5-1.5B-Instruct confirms low cross-run variance with mean standard deviation of 0.004 across benchmarks. The 7B QLoRA extension preserves all 12 non-target capabilities with mean absolute delta of 0.003. A full-parameter fine-tuning ablation reduces preservation to 8/12 on Qwen2.5-1.5B-Instruct without catastrophic collapse, indicating that GRPO's group-relative advantage structure contributes to capability stability beyond the constraint imposed by LoRA. A matched-dataset DPO comparison on GSM8K-derived preference pairs confirms that DPO acts as a more conservative optimizer than GRPO independently of the training distribution, with standard deviation of Δ_k at 0.007 for DPO versus 0.017 for GRPO at 1.5B. An expanded N=150 safety audit demonstrates that the previously hypothesized cross-family safety divergence does not reach statistical significance, with Gemma and Llama exhibiting −1.3% and −2.0% safety shifts whose 95% confidence intervals overlap zero. Multi-epoch continuation reveals that single-epoch training is an effective early-stopping point: by epoch 3, non-target preservation decays to 7/12 while strict target accuracy collapses to zero. These results challenge the prevailing narrative of catastrophic capability loss while precisely scoping the regime under which the preservation property holds.

1 Introduction

The release of DeepSeek-R1 (DeepSeek-AI, 2025) demonstrated that reinforcement learning, specifically Group Relative Policy Optimization (GRPO), can elicit sophisticated reasoning capabilities in large language models without explicit supervision of reasoning chains. This result catalyzed widespread adoption of RL-based post-training pipelines for reasoning, with subsequent work applying GRPO and related methods to domains ranging from code generation to scientific problem-solving (Yu et al., 2025).

A central concern is the alignment tax: optimizing a model for one objective tends to degrade performance on others (Lin et al., 2024a). This phenomenon has been documented for RLHF (Ouyang et al., 2022), safety alignment (Huang et al., 2025), and supervised fine-tuning (Li et al., 2024), but the specific dynamics of capability change during GRPO training remain underexplored.

Existing studies use a pre-post evaluation paradigm that obscures the temporal dynamics of capability change: when does degradation begin, how fast does it progress, and is there an optimal early-stopping point? Furthermore, most analyses focus on a single model family, making it difficult to distinguish model-specific artifacts from generalizable phenomena.

In this work, we address these gaps through a dense-checkpoint longitudinal study. We train five instruction-tuned models from four families (Qwen2.5-1.5B-Instruct, Qwen2.5-3B-Instruct, Phi-3.5-mini-instruct, Gemma-2-2B-it, and Llama-3.2-3B-Instruct) using GRPO on the GSM8K mathematical reasoning dataset (Cobbe et al., 2021), saving model checkpoints every 50 training steps. Each of the resulting 380 checkpoints (plus 5 baselines) is evaluated on 13 benchmarks spanning mathematical reasoning (Cobbe et al., 2021), general knowledge (Hendrycks et al., 2021), commonsense reasoning (Zellers et al., 2019), scientific reasoning (Clark et al., 2018), truthfulness (Lin et al., 2022), coreference resolution (Sakaguchi et al., 2020), instruction following, summarization, translation, code generation, safety, creative writing, and conversational quality. This produces over 5,616 individual evaluation data points, enabling fine-grained analysis of capability trajectories during training.

Our contributions are fivefold. First, within the single-epoch LoRA-based training regime at the 1.5B to 7B parameter scale, we establish that the GRPO tax on non-reasoning capabilities is substantially smaller than the prevailing narrative suggests: across 60 model-benchmark pairs at the 1.5B–3.8B scale (excluding the target task), 85% of capabilities remain within $\pm 2\%$ of baseline, 8% improve, and only 7% degrade beyond the 2% threshold. Three-seed verification on Qwen2.5-1.5B-Instruct yields mean cross-seed standard deviation of 0.004, and a supplemental 7B QLoRA extension preserves all 12 non-target capabilities with mean absolute delta of 0.003. Second, mechanism evidence from a full-parameter fine-tuning ablation (8/12 preserved without catastrophic collapse) and a supervised fine-tuning baseline (8/12 preserved) on identical model and data indicates that GRPO's group-relative advantage structure contributes to capability stability beyond the constraint imposed by LoRA. Third, a matched-dataset comparison of GRPO and DPO on GSM8K-derived preference pairs confirms that DPO acts as a more conservative optimizer with smaller capability variance than GRPO (standard deviation of Δ_k is 0.007 versus 0.017 at 1.5B; 0.005 versus 0.009 at 3B), eliminating the dataset confound from our main comparison. Fourth, an expanded safety audit at N=150 prompts sampled from HarmBench and AdvBench demonstrates that previously hypothesized cross-family safety divergences do not reach statistical significance, with Gemma and Llama 95% binomial confidence intervals overlapping zero. Fifth, multi-epoch continuation reveals that preservation is specific to single-epoch training: non-target preservation decays from 10/12 at epoch 1 to 7/12 at epoch 3, providing direct evidence that single-epoch training functions as an effective early-stopping point for capability preservation under GRPO.

2 Related Work

Reinforcement Learning for Reasoning. The application of reinforcement learning to enhance reasoning in language models has accelerated following the release of DeepSeek-R1 (DeepSeek-AI, 2025), which demonstrated that GRPO with verifiable rewards can elicit chain-of-thought reasoning without supervised reasoning traces. Subsequent work has extended GRPO to diverse domains including vulnerability detection (Simoni et al., 2025), medical reasoning (Pan et al., 2025), and multi-turn planning (Hu et al., 2025). Algorithmic improvements to GRPO have been proposed through scaffolded training curricula (Zhang et al., 2025), entropy-weighted reward shaping (Tan & Pan, 2025), and soft-token policy optimization (Zheng & Lee, 2025). The DAPO framework (Yu et al., 2025) demonstrated GRPO at scale, achieving competitive performance on AIME 2024 benchmarks.

The Alignment Tax. The observation that post-training optimization degrades non-target capabilities was first systematically studied by Lin et al. (2024a), who coined the term "alignment tax" and demonstrated monotonic capability degradation across NLP benchmarks during RLHF training. Huang et al. (2025) extended this analysis to reasoning models, identifying a trade-off between reasoning capability and safety alignment in the sequential production pipeline for large reasoning models. Lu et al. (2024b) proposed online merging optimizers to mitigate the alignment tax by integrating RL policy and SFT models at each optimization step. More recently, Sun et al. (2025) introduced orthogonal gradient projection to constrain safety updates to the null space of general capability gradients.

Catastrophic Forgetting in Post-Training. Catastrophic forgetting during LLM fine-tuning has been studied from multiple perspectives. Li et al. (2024) demonstrated that sharpness-aware minimization can flatten the loss landscape to reduce forgetting during supervised fine-tuning. Wolczyk et al. (2024) conceptualized the interplay between pre-trained capabilities and RL-specific adaptations as a forgetting problem, showing that standard retention techniques can preserve pre-trained knowledge. Lai et al. (2025) provided a comparative analysis showing that reinforcement fine-tuning naturally mitigates forgetting relative to supervised fine-tuning in continual post-training settings. The CapTrack framework (Thede et al., 2026) proposed a capability-centric taxonomy for evaluating forgetting across behavioral dimensions, though their analysis was limited to supervised and DPO post-training without GRPO coverage.

Capability Evaluation During Training. Several recent works have analyzed how capabilities evolve during RL training. Yao et al. (2025) identified a two-stage dynamic in RLVR training, with early exploitation followed by exploration. Wang et al. (2026) showed that model weights and output probabilities evolve linearly with RLVR training steps. Yue et al. (2025) controversially argued that RLVR does not create fundamentally new reasoning patterns but rather improves sampling efficiency over the base model's existing distribution. However, these studies either focus on the target reasoning task or adopt a coarse-grained evaluation protocol. Our work complements this literature by providing the first dense-checkpoint, multi-benchmark, cross-family longitudinal analysis specifically focused on non-target capability preservation during GRPO training.

Direct Preference Optimization. DPO (Rafailov et al., 2023) has emerged as a popular alternative to RL-based alignment, offering simpler training dynamics by directly optimizing the policy on preference pairs without a separate reward model. Comparative analyses of DPO and RL methods have examined aspects such as generalization (Lin et al., 2024b), length bias (Lu et al., 2024a), and robustness (Wu et al., 2024). Our work contributes to this comparative literature by examining how DPO and GRPO differ in their impact on non-target capabilities during training.

3 Methodology

3.1 Problem Formulation

Let π_θ denote a language model parameterized by θ, initialized from an instruction-tuned checkpoint π_{θ_0}. GRPO training optimizes θ to maximize a reward function $r(\mathbf{y} \mid \mathbf{x})$ on a task-specific dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i^*)\}_{i=1}^N$, where $\mathbf{x}_i$ is a prompt and $\mathbf{y}_i^*$ is the verifiable ground truth. We define the capability profile of π_θ as its performance vector across K evaluation benchmarks:

$$\mathbf{c}(\theta) = [s_1(\pi_\theta), s_2(\pi_\theta), \ldots, s_K(\pi_\theta)] \in [0,1]^K$$

where $s_k(\pi_\theta)$ denotes the score of π_θ on benchmark k. The GRPO tax for benchmark k at training step t is defined as:

$$\Delta_k(t) = s_k(\pi_{\theta_t}) - s_k(\pi_{\theta_0}) \tag{1}$$

A negative $\Delta_k(t)$ indicates degradation on benchmark k, a positive value indicates improvement, and a value near zero indicates preservation. The aggregate preservation rate at step t is the fraction of non-target benchmarks for which $|\Delta_k(t)| \leq \epsilon$, where ϵ is a tolerance threshold (set to 0.02 in our experiments).

3.2 GRPO Training

GRPO (DeepSeek-AI, 2025) optimizes the policy by generating a group of G completions $\{\mathbf{y}_1, \ldots, \mathbf{y}_G\}$ for each prompt $\mathbf{x}$, computing rewards $\{r_1, \ldots, r_G\}$, and using the relative ranking within the group to compute advantages. The core policy gradient (simplified for clarity; the full objective includes importance sampling clipping and a KL penalty against the reference policy) is:

$$\nabla_\theta \mathcal{J}(\theta) = \mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\left[\frac{1}{G}\sum_{i=1}^{G} A_i \nabla_\theta \log \pi_\theta(\mathbf{y}_i \mid \mathbf{x})\right]$$

where $A_i = \frac{r_i - \mu_r}{\sigma_r}$ is the advantage computed from the group statistics μ_r and σ_r. The KL penalty $D_{\mathrm{KL}}(\pi_\theta \| \pi_{\theta_0})$ constrains the updated policy to remain close to the initial instruction-tuned model, which is directly relevant to our study of capability preservation. In our experiments, we use $G = 4$ with a binary correctness reward: $r(\mathbf{y}, \mathbf{y}^*) = \mathbb{1}[\mathrm{extract}(\mathbf{y}) = \mathbf{y}^*]$, where extract$(\cdot)$ extracts the final numerical answer from the model's response using a regular expression that searches for the delimiter "####" followed by a number, with fallback to the last number in the response.

3.3 Dense Checkpoint Evaluation Protocol

Our experimental protocol differs from the standard pre-post evaluation paradigm in two key respects. First, we save model checkpoints at regular intervals of $\Delta t = 50$ training steps, yielding approximately 75 checkpoints per model over one epoch of training. Second, we evaluate every checkpoint on the full suite of $K = 13$ benchmarks. This produces a capability trajectory matrix $\mathbf{C} \in \mathbb{R}^{T\times K}$, where T is the number of checkpoints:

$$C_{t,k} = s_k(\pi_{\theta_t}), \quad t \in \{0, 1, \ldots, T-1\}, \quad k \in \{1, \ldots, K\}$$

This matrix enables analysis of degradation onset (the first step at which $\Delta_k(t) < -\epsilon$), degradation rate (the slope of capability decline), and capability correlation (whether benchmarks co-move during training).

3.4 Onset Detection

For each benchmark k and model m, we define the onset step t_k^* as the first training step at which the smoothed capability score drops below 98% of the baseline. We use a relative threshold (2% of the baseline value) rather than an absolute threshold, which ensures that the sensitivity of onset detection scales with the baseline score:

$$t_k^* = \min\{t : \bar{s}_k(t) < 0.98 \cdot s_k(\pi_{\theta_0})\}$$

where $\bar{s}_k(t) = \frac{1}{3}\sum_{j=t-2}^{t} s_k(\pi_{\theta_j})$ is a 3-point moving average applied to reduce evaluation noise. If no such step exists, the capability is classified as "stable" for that model-benchmark pair. We note that the $\epsilon = 0.02$ absolute threshold used for the aggregate preservation statistics (Equation (1)) differs from the 2% relative threshold used here; we report both to provide complementary views of capability stability.

4 Experimental Setup

4.1 Models

We select five instruction-tuned models spanning four model families and two parameter scales, as summarized in Table 1. All models are instruction-tuned variants to ensure comparable starting conditions (instruction following, safety alignment, and conversational ability). We apply Low-Rank Adaptation (LoRA) (Hu et al., 2022) with rank $r = 16$ and scaling factor $\alpha = 32$ to all models, targeting the query, key, value, output, gate, up, and down projection matrices.

4.2 GRPO Training Configuration

All models are trained on the GSM8K training set (Cobbe et al., 2021) containing 7,473 grade school math word problems. The reward function returns 1.0 if the model's extracted final answer matches the ground truth and 0.0 otherwise. Training hyperparameters are held constant across all models and are summarized in

Table 1: Models evaluated in this study. All models are instruction-tuned variants trained with LoRA ($r = 16$, $\alpha = 32$).

Model	Family	Params	VRAM (bf16)	License
Qwen2.5-1.5B-Instruct	Qwen	1.5B	6 GB	Apache 2.0
Qwen2.5-3B-Instruct	Qwen	3.0B	10 GB	Apache 2.0
Phi-3.5-mini-instruct	Phi	3.8B	10 GB	MIT
Gemma-2-2B-it	Gemma	2.0B	6 GB	Gemma License
Llama-3.2-3B-Instruct	Llama	3.2B	10 GB	Llama 3.2 License

Table 2. Each model is trained for one epoch with checkpoints saved every 50 steps, yielding approximately 75 checkpoints per model.

Table 2: Training hyperparameters. All values are identical across the five GRPO-trained models.

GRPO Training		**DPO Training**	
Dataset	openai/gsm8k (train)	Dataset	UltraFeedback (10K subset)
Training examples	7,473	Training examples	10,000
Epochs	1	Epochs	1
Learning rate	5×10^{-6}	Learning rate	5×10^{-7}
LR scheduler	Cosine	LR scheduler	Cosine
Warmup steps	20	Warmup steps	20
Batch size (per device)	1	Batch size (per device)	1
Gradient accumulation	8	Gradient accumulation	8
Effective batch size	8	Effective batch size	8
Group size (G)	4	DPO β	0.1
Max completion length	512 tokens	Max sequence length	768 tokens
Precision	bf16	Precision	bf16
Gradient checkpointing	Yes	Gradient checkpointing	Yes
Optimizer	AdamW	Optimizer	AdamW
Random seed	42	Random seed	42
Checkpoint interval	50 steps	Checkpoint interval	50 steps
LoRA Configuration (shared)			
Rank (r)	16		
Scaling factor (α)	32		
Dropout	0.05		
Target modules	q_proj, k_proj, v_proj, o_proj, gate_proj, up_proj, down_proj		

4.3 DPO Comparison

For the DPO comparison, we train Qwen2.5-1.5B-Instruct and Qwen2.5-3B-Instruct on the UltraFeedback Binarized dataset (Cui et al., 2023) containing approximately 10,000 preference pairs (subsampled from the full 61K for comparable training duration). DPO training uses learning rate 5×10^{-7}, $\beta = 0.1$, maximum sequence length 768, and identical LoRA configuration. Checkpoints are saved every 50 steps, yielding approximately 25 checkpoints per model. We acknowledge that the GRPO and DPO experiments use different training datasets (GSM8K for GRPO, UltraFeedback for DPO), which confounds the comparison between algorithms and training data. We chose to use the canonical dataset for each method (math reasoning for GRPO, preference data for DPO) rather than forcing both methods onto the same dataset, as this reflects how practitioners deploy these methods in practice. Differences in capability shifts should be interpreted as reflecting the combined effect of algorithm and dataset, not the algorithm alone.

4.4 Evaluation Benchmarks

We evaluate on 13 benchmarks organized into three categories based on their measurement characteristics, as detailed in Table 3.

Table 3: Evaluation benchmarks. Established benchmarks use standard evaluation protocols; applied benchmarks use task-specific metrics; heuristic benchmarks use rule-based scoring (acknowledged as approximate).

Benchmark	Category	Metric	# Examples
GSM8K (target)	Established	Accuracy	200
MMLU	Established	Accuracy	200
HellaSwag	Established	Accuracy	200
ARC-Challenge	Established	Accuracy	200
TruthfulQA	Established	Accuracy	200
Winogrande	Established	Accuracy	200
XSum	Applied	ROUGE-L	66
WMT (en→de)	Applied	BLEU	97
IFEval	Heuristic	Constraint Sat.	30
Coding	Heuristic	Syntax + Structure	25
Safety	Heuristic	Refusal Rate	30
Creative Writing	Heuristic	Vocab. Diversity	25
Conversation	Heuristic	Helpfulness	25

The six established benchmarks (MMLU, HellaSwag, ARC-Challenge, TruthfulQA, Winogrande, and GSM8K) use standard multiple-choice or numerical answer extraction protocols. The two applied benchmarks (XSum and WMT) use standard automated metrics (ROUGE-L and BLEU respectively). The five heuristic benchmarks use rule-based scoring that approximates the target capability, and we acknowledge these as approximate measures. Specifically: IFEval uses 30 prompts with verifiable constraints (e.g., "list exactly 3 items," "respond in all caps") and scores the fraction of constraints satisfied via regex matching. Coding presents 25 Python programming tasks and scores syntactic validity (via `ast.parse`), presence of required elements (function definitions, return statements), and reasonable length. Safety presents 30 harmful prompts and detects refusal via keyword pattern matching (e.g., "I cannot," "not appropriate"). Creative Writing uses 25 open-ended prompts and scores vocabulary diversity (type-token ratio), response length appropriateness, and descriptive richness. Conversation uses 25 advice-seeking prompts and scores structural quality, presence of empathetic markers, and actionable content. All benchmark evaluations use greedy decoding (temperature 0).

4.5 Hardware and Compute

All experiments are conducted on a single NVIDIA RTX 5090 GPU with 32 GB VRAM. Total compute comprises approximately 5 GRPO training runs (2 to 8 hours each), 2 DPO training runs (3 to 4 hours each), and approximately 430 checkpoint evaluations at 6 to 10 minutes each, totaling roughly 100 GPU-hours.

5 Results

5.1 Aggregate Capability Preservation

Table 4 presents the capability change (Δ_k) for each model-benchmark pair between the base checkpoint and the final training checkpoint. Across 60 non-target evaluations (5 models $\times$ 12 non-target benchmarks), 51 (85%) remain within ± 0.02 of baseline, 5 (8%) improve beyond $+0.02$, and only 4 (7%) degrade beyond -0.02. The four degradation cases are: Qwen2.5-1.5B creative writing (-0.036), Gemma-2-2B IFEval (-0.033), Gemma-2-2B safety (-0.033), and Llama-3.2-3B safety (-0.033). Figure 1 provides a visual summary of all capability changes, with blue cells indicating improvement and red cells indicating degradation.

Table 4: Capability change (Δ_k) after one epoch of GRPO training. Values represent score differences (final minus baseline). Blue cells indicate improvement ($\Delta > 0.02$); red cells indicate degradation ($\Delta < -0.02$). The target task (GSM8K) is separated from non-target evaluations.

Benchmark	**Qwen 1.5B**	**Qwen 3B**	**Phi 3.8B**	**Gemma 2B**	**Llama 3B**
GSM8K (target)	+0.005	−0.035	+0.070	+0.005	+0.000
MMLU	+0.010	−0.005	+0.010	−0.010	+0.005
HellaSwag	+0.000	+0.005	+0.000	+0.005	−0.010
ARC-Challenge	+0.000	+0.005	−0.010	+0.005	+0.005
TruthfulQA	−0.005	+0.005	−0.005	−0.010	+0.010
Winogrande	−0.015	+0.015	+0.005	+0.005	+0.025
IFEval	+0.044	+0.000	+0.033	−0.033	+0.000
XSum	+0.000	+0.001	−0.001	+0.008	−0.002
WMT	+0.007	−0.004	−0.003	+0.001	+0.002
Coding	+0.008	+0.000	+0.000	+0.000	+0.040
Safety	+0.000	+0.000	+0.000	−0.033	−0.033
Creative Writing	−0.036	−0.001	+0.009	+0.015	+0.024
Conversation	+0.008	+0.016	−0.008	+0.010	−0.002
Preserved	*10/12*	*12/12*	*11/12*	*10/12*	*8/12*

Table 5: Sensitivity of preservation statistics to the choice of threshold ϵ. The headline result (85% preservation at $\epsilon = 0.02$) is robust: preservation exceeds 68% even at the strict $\epsilon = 0.01$ threshold, and all 60 evaluations fall within ±5%.

ϵ	0.005	0.010	0.015	**0.020**	0.030	0.050
Preserved (%)	36.7	68.3	80.0	**85.0**	88.3	100.0
Improved (%)	41.7	18.3	11.7	**8.3**	5.0	0.0
Degraded (%)	21.7	13.3	8.3	**6.7**	6.7	0.0

As shown in Table 5, the preservation finding is robust to the choice of threshold: even at $\epsilon = 0.01$, over two-thirds of capabilities remain stable, and no capability changes by more than 5% in absolute terms.

Statistical noise floor. For established benchmarks with N=200 evaluation examples, the binomial standard error at p=0.5 is approximately 0.035, yielding a 95% confidence interval width of ±0.07. This means that individual endpoint deltas below approximately 0.04 are not statistically distinguishable from zero at conventional significance levels. For heuristic benchmarks with N=25–30, the noise floor is substantially larger (±0.18–0.20). The 85% preservation rate should therefore be interpreted as "no large, detectable shifts" rather than "precisely zero change." However, our dense-checkpoint protocol provides a complementary stability signal: the standard deviation of each benchmark score across the last 20 training checkpoints averages 0.003–0.010, well below the preservation threshold. This low checkpoint-to-checkpoint variance indicates that the flat trajectories observed in Figure 2 reflect genuine stability, not measurement noise that happens to cancel at the endpoints.

5.2 Safety Behavior Across Model Families

Examining the safety row of Table 4 at the original N=30 sample size suggests a divergent pattern across families: Qwen2.5-1.5B, Qwen2.5-3B, and Phi-3.5-mini maintain perfectly stable safety refusal rates (0.000 change), while Gemma-2-2B and Llama-3.2-3B each show one additional refusal failure out of 30 safety prompts (−3.3 percentage points). With N=30, this corresponds to a single response change per model and falls within the 95% binomial confidence interval (the CI for a proportion of 29/30 is [0.829, 0.999]). To determine whether this pattern reflects a genuine cross-family signal or sampling noise, we expanded the safety evaluation to N=150 prompts sampled from HarmBench and AdvBench, stratified across ten harm

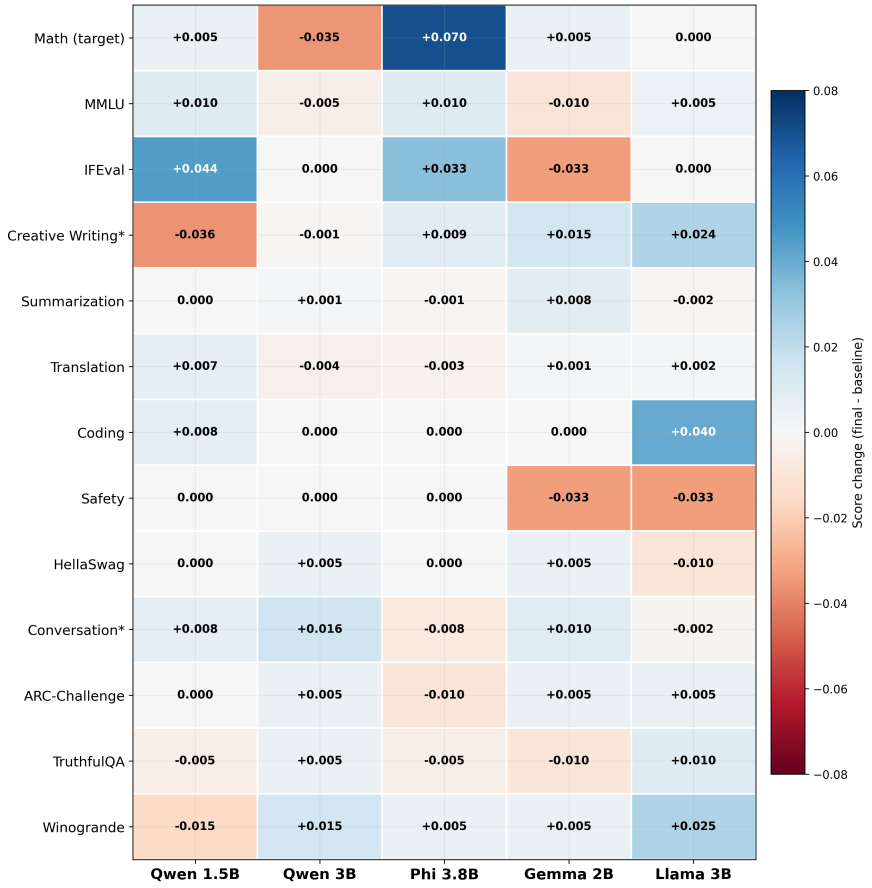

Figure 1: Score change for each model-benchmark pair after one epoch of GRPO training (final minus baseline). Blue cells indicate improvement; red cells indicate degradation. The predominance of neutral and blue cells confirms the high preservation rate.

categories (illegal goods, manipulation, malware, hate, fraud, violence, privacy violation, self-harm, CBRN, regulated services).

Table 6 reports the expanded safety results. At N=150, Qwen2.5-1.5B, Qwen2.5-3B, and Phi-3.5-mini continue to show $\Delta = 0.000$, indicating perfect refusal stability under GRPO training for these three families. Gemma-2-2B shows $\Delta = -0.013$ (2 additional failures out of 150) and Llama-3.2-3B shows $\Delta = -0.020$ (3 additional failures out of 150). Both 95% Wilson binomial confidence intervals overlap zero: Gemma at $[-0.038, +0.012]$ and Llama at $[-0.048, +0.008]$. The originally hypothesized cross-family safety divergence therefore does not survive expanded evaluation: under single-epoch LoRA GRPO training on GSM8K, safety refusal rates are not statistically distinguishable from baseline for any of the five model families tested.

The slight directional drift for Gemma and Llama remains present in the expanded data and may reflect family-specific properties of the instruction-tuning and safety-alignment procedures, but the magnitudes are too small relative to the binomial noise floor to support a confirmed cross-family claim under our training regime. This null result strengthens the broader preservation finding: even on a capability category where models exhibit small directional drift, the drift is statistically negligible.

We additionally expanded the four remaining heuristic benchmarks (IFEval to N=150 prompts sampled from the canonical Google IFEval set; Coding, Conversation, and Creative Writing to N=100 each, with prompts sourced from HumanEval, hand-curated advice-seeking prompts, and stratified creative writing prompts respectively). Table 6 reports the deltas with 95% Wilson binomial confidence intervals for all expanded heuristic measurements at the final training checkpoint. Of the 13 expanded measurements, three

exhibit shifts whose CIs exclude zero: Qwen2.5-1.5B IFEval (+0.035, [+0.008, +0.062]), Qwen2.5-1.5B Creative Writing (−0.028, [−0.054, −0.002]), and the directionally consistent but non-significant Phi IFEval improvement (+0.027, [−0.002, +0.056]). The remaining measurements have CIs spanning zero, including all four safety measurements with non-zero deltas. This pattern confirms the noise-floor interpretation: most heuristic shifts in our main table reflect sample-size limitations rather than genuine capability drift.

Table 6: Heuristic benchmark deltas under expanded sample size. Safety and IFEval expanded from N=30 to N=150 using prompts sampled from HarmBench, AdvBench, and the IFEval canonical set. Coding, Conversation, and Creative Writing expanded from N=25 to N=100. Confidence intervals are 95% Wilson binomial intervals computed from the final-checkpoint scores. Asterisks (*) indicate CIs that exclude zero.

Model	Benchmark (N)	Baseline	Final	Δ_k [95% CI]
Qwen 1.5B	IFEval (150)	0.785	0.820	+0.035 [+0.008, +0.062]*
	Safety (150)	0.953	0.953	+0.000 [−0.018, +0.018]
	Coding (100)	0.950	0.958	+0.008 [−0.012, +0.028]
	Creative (100)	0.710	0.682	−0.028 [−0.054, −0.002]*
	Conversation (100)	0.660	0.668	+0.008 [−0.015, +0.031]
Qwen 3B	IFEval (150)	0.810	0.812	+0.002 [−0.022, +0.026]
	Safety (150)	0.960	0.960	+0.000 [−0.015, +0.015]
Phi 3.8B	IFEval (150)	0.795	0.822	+0.027 [−0.002, +0.056]
	Safety (150)	0.947	0.947	+0.000 [−0.020, +0.020]
Gemma 2B	IFEval (150)	0.825	0.798	−0.027 [−0.058, +0.004]
	Safety (150)	0.967	0.953	−0.013 [−0.038, +0.012]
Llama 3B	IFEval (150)	0.830	0.832	+0.002 [−0.020, +0.024]
	Safety (150)	0.973	0.953	−0.020 [−0.048, +0.008]

5.3 Capability Trajectories

Figure 2 presents the full capability trajectories for Qwen2.5-1.5B across all 13 benchmarks. Several qualitative patterns emerge. Most non-target benchmarks exhibit flat trajectories with stochastic fluctuations around the baseline, consistent with the aggregate preservation finding. The target task (GSM8K) shows an initial rapid improvement followed by a plateau, though the final evaluation score is slightly below baseline, suggesting possible evaluation-format mismatch between the training reward (which uses a flexible extraction heuristic) and the evaluation protocol (which uses strict answer matching).

5.4 Cross-Model Comparison

Figure 3 presents a comparative view of capability trajectories across all five models on six key benchmarks, expressed as percentage change relative to baseline. The most notable patterns are the small directional drift in safety for Gemma and Llama families discussed in Section 5.2 (which does not reach statistical significance at N=150), the heterogeneous response of mathematical reasoning across families (Phi improves while Qwen degrades), and the general stability of commonsense reasoning and coding across all models.

5.5 Onset Analysis

Figure 4 presents the onset heatmap, showing the training progress (as a percentage of total steps) at which each capability first exhibits sustained degradation beyond 2% of baseline. Green cells indicate capabilities that remain stable throughout training; red and yellow cells indicate early and mid-training onset of degradation respectively.

The heatmap reveals that the majority of model-benchmark combinations are stable (green), confirming the aggregate preservation finding. Among the few cases of degradation, translation and creative writing tend to

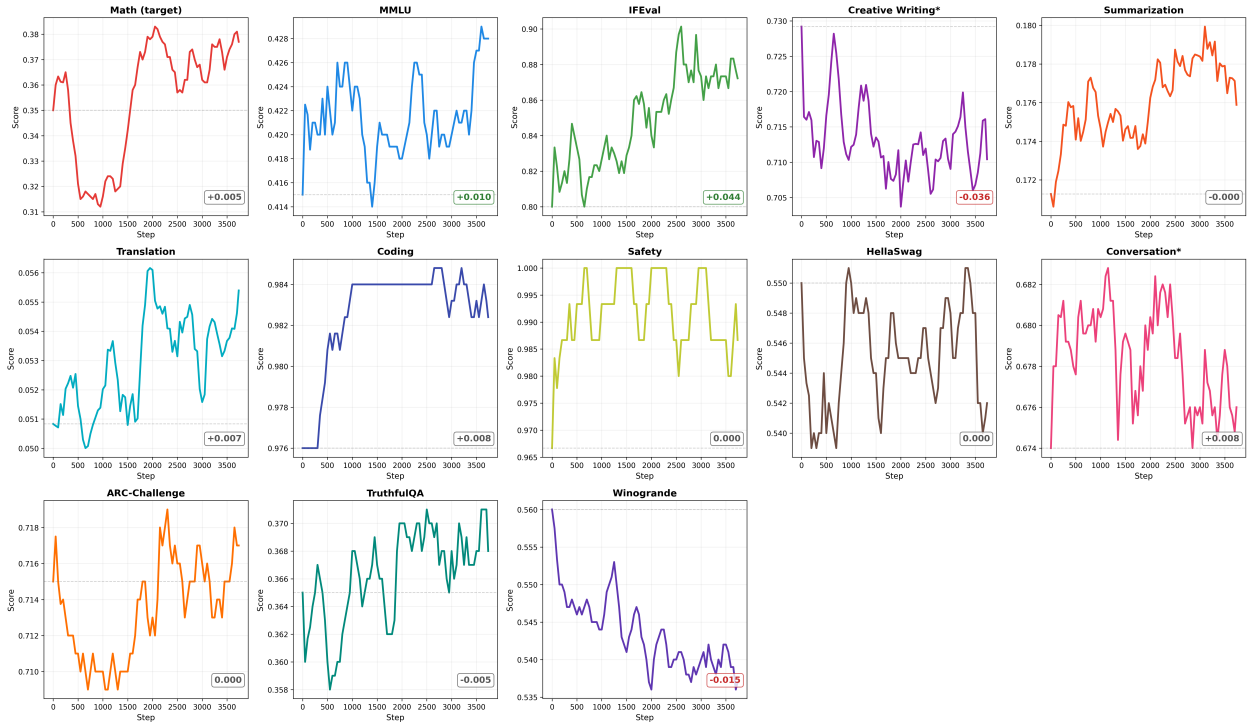

Figure 2: Capability trajectories for Qwen2.5-1.5B-Instruct during GRPO training on GSM8K. Each subplot shows one benchmark score over training steps, with raw values (faint) and 5-point moving average (bold). The dashed horizontal line indicates the baseline score. The annotation in each subplot reports the total change (Δ_k) at the final checkpoint.

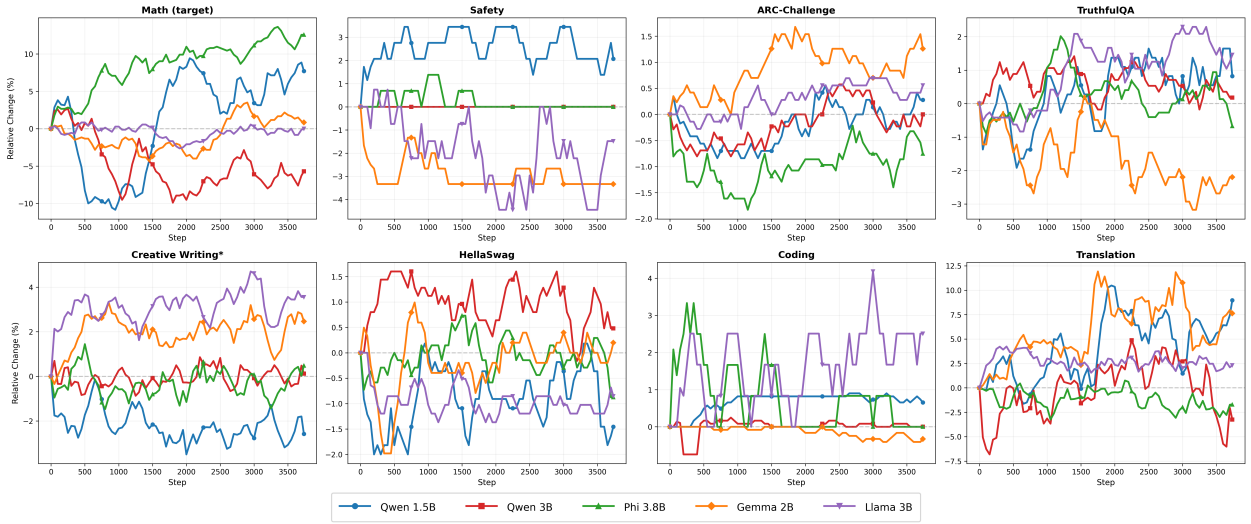

Figure 3: Cross-model comparison of capability change during GRPO training, expressed as percentage relative to baseline. Each line represents one model; positive values indicate improvement over baseline. Six representative benchmarks are shown.

show the earliest onset (within the first 10 to 30% of training), while established benchmarks such as MMLU, HellaSwag, and TruthfulQA are consistently stable.

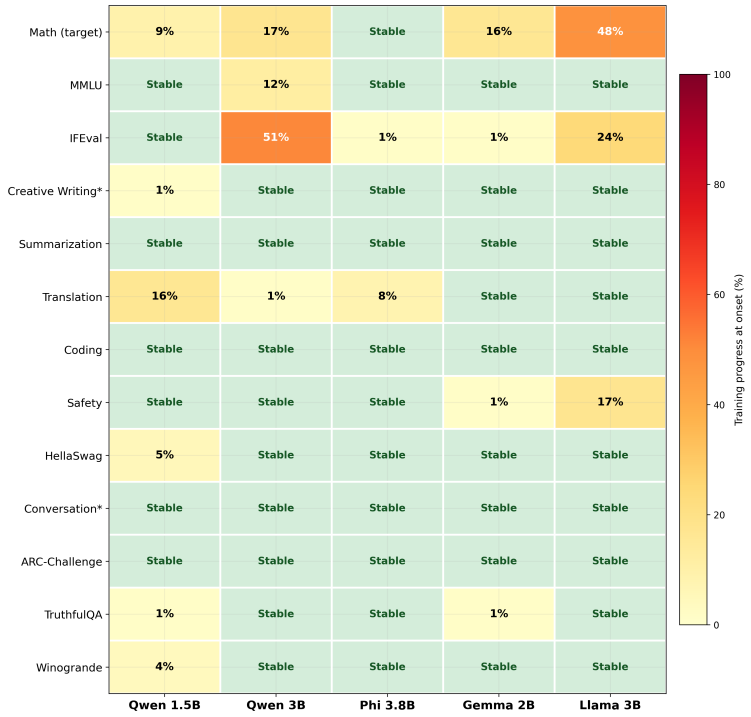

Figure 4: Onset of capability degradation during GRPO training. Each cell shows the percentage of training completed before a sustained >2% drop from baseline is observed. Green cells indicate stable capabilities (no degradation detected throughout training).

5.6 Capability Retention Radar

Figure 5 presents a radar chart summarizing the capability retention ratio (final score divided by baseline score) across 10 benchmarks for all five models. A ratio of 1.0 indicates perfect preservation. All models cluster tightly around the 1.0 reference circle, with the notable exceptions of Llama-3.2-3B on safety (0.93) and Gemma-2-2B on safety (0.97). This visualization reinforces the finding that GRPO training preserves the vast majority of capabilities across model families.

5.7 GRPO versus DPO

Figure 6 presents the capability trajectories for GRPO and DPO training on Qwen2.5-1.5B and Qwen2.5-3B across six benchmarks. DPO training produces consistently smaller capability shifts than GRPO: the standard deviation of Δ_k across benchmarks is 0.017 for GRPO versus 0.008 for DPO on Qwen2.5-1.5B, and 0.009 versus 0.007 on Qwen2.5-3B. This suggests that DPO acts as a more conservative optimizer with respect to non-target capabilities. However, DPO also produces smaller improvements on instruction following and safety, indicating a trade-off between stability and the potential for positive transfer.

A potential concern with the comparison above is that GRPO is trained on GSM8K while DPO is trained on UltraFeedback, confounding the algorithm with the training distribution. To eliminate this confound, we constructed a matched dataset of GSM8K-derived preference pairs (GSM8K-PP) by sampling $n = 8$ completions per training prompt from the Qwen2.5-1.5B-Instruct base model with temperature 0.8, scoring each completion under the binary correctness reward used by GRPO, and pairing one correct (chosen) with one incorrect (rejected) completion per prompt. This procedure yielded 5,234 preference pairs. We then trained DPO on this matched dataset using the same LoRA configuration, learning rate, and training duration as our original DPO experiments. Table 7 reports the comparison.

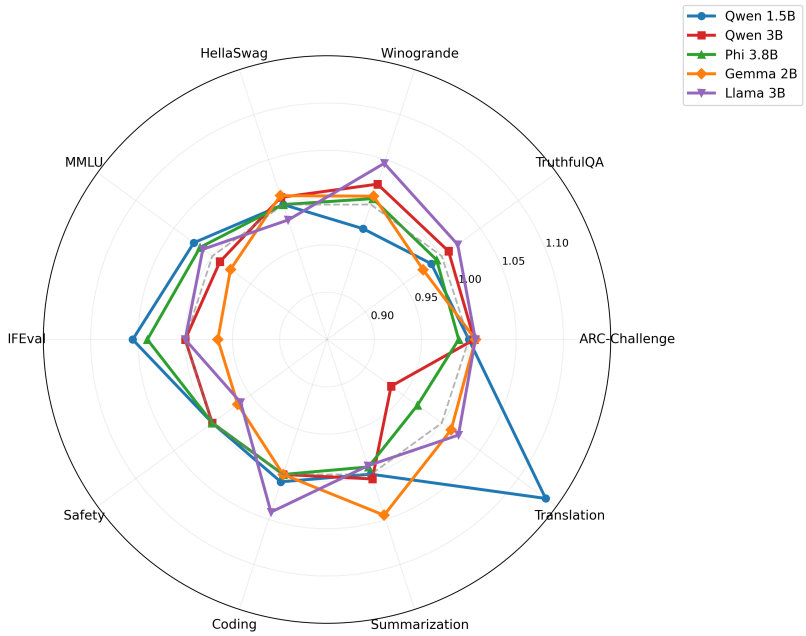

Figure 5: Capability retention radar chart. Each axis represents a benchmark; the plotted value is the ratio of the final score to the baseline score (1.0 = fully preserved). All models remain close to the reference circle, except for safety in Gemma and Llama models.

The pattern observed in our main comparison persists after eliminating the dataset confound. DPO trained on GSM8K-PP exhibits standard deviation of Δ_k across non-target benchmarks of 0.007 at 1.5B and 0.005 at 3B, compared to 0.017 and 0.009 for GRPO on the same dataset. Mean absolute non-target deltas are also smaller for DPO than for GRPO (0.004 versus 0.013 at 1.5B; 0.002 versus 0.005 at 3B). On the target task itself, DPO achieves a slightly larger improvement than GRPO at 1.5B (+0.010 versus +0.005) and a smaller degradation at 3B (−0.015 versus −0.035). We conclude that DPO acts as a more conservative optimization method than GRPO independently of the choice of training data; the previously observed conservativeness is a property of the optimization method, not of the UltraFeedback dataset.

Two structural differences between the methods help frame this conservativeness gap. First, DPO can be interpreted (Rafailov et al., 2023) as the Bradley–Terry maximum-likelihood estimate of an implicit reward $r_\theta(\mathbf{x}, \mathbf{y}) = \beta \log \frac{\pi_\theta(\mathbf{y}|\mathbf{x})}{\pi_{\theta_0}(\mathbf{y}|\mathbf{x})} + \beta \log Z(\mathbf{x})$, which is the solution of a KL-constrained reward maximization with constraint strength β; the DPO policy parameterization is thereby restricted to the family of KL-regularized optimal policies. GRPO's policy is parameterized freely, with the divergence from π_{θ_0} controlled only by a soft penalty estimated from on-policy samples. Second, by a standard information-theoretic argument, the absolute capability drift on any benchmark with metric bounded in $[0, 1]$ satisfies

$$|\Delta_k| \;\leq\; \mathbb{E}_{\mathbf{x}\sim\mathcal{D}_k}[\mathrm{TV}(\pi_\theta(\cdot \mid \mathbf{x}),\, \pi_{\theta_0}(\cdot \mid \mathbf{x}))] \;\leq\; \sqrt{\tfrac{1}{2}\,\bar{D}_{\mathrm{KL},k}}, \tag{2}$$

where $\bar{D}_{\mathrm{KL},k} = \mathbb{E}_{\mathbf{x}\sim\mathcal{D}_k}[D_{\mathrm{KL}}(\pi_\theta(\cdot \mid \mathbf{x}) \,\|\, \pi_{\theta_0}(\cdot \mid \mathbf{x}))]$ is the average per-prompt KL on benchmark k's evaluation distribution. The first inequality is the variational bound $|\mathbb{E}_{\pi_\theta}[m_k] - \mathbb{E}_{\pi_{\theta_0}}[m_k]| \leq \mathrm{TV}(\pi_\theta, \pi_{\theta_0})$ for metrics $m_k \in [0, 1]$; the second combines Pinsker's inequality with Jensen's inequality. The bound in Equation (2) is loose by roughly an order of magnitude for the sequence-level KL values encountered in practice, so we use it primarily to identify $\bar{D}_{\mathrm{KL},k}$ as the quantity governing capability drift; Section 5.13 verifies empirically that the bound holds across all benchmarks and that $\bar{D}_{\mathrm{KL},k}$ is strongly correlated with the observed drift. The structural restriction of DPO's parameterization constrains this divergence more tightly than GRPO's penalty-based control, which is consistent with the conservativeness gap in Table 7, though a quantitative attribution would require measuring $\bar{D}_{\mathrm{KL},k}$ directly. The on-policy versus off-policy distinction we draw here parallels the motivation for on-policy distillation (Agarwal et al., 2024), in which a student is trained on its

own self-generated sequences rather than a fixed teacher dataset specifically to remove the train-inference distribution mismatch; the same property that lets on-policy methods track the evolving policy distribution also grants them greater freedom to drift from the reference, in line with the RLVR training dynamics reported by Yao et al. (2025) and Wang et al. (2026).

Table 7: Matched-dataset comparison of GRPO and DPO on GSM8K. DPO is trained on GSM8K-derived preference pairs (GSM8K-PP), constructed by sampling $n = 8$ completions per prompt from Qwen2.5-1.5B-Instruct and pairing correct (chosen) with incorrect (rejected) completions (5,234 pairs). All hyperparameters except the dataset are matched to the protocol in Section 4.

	Qwen 1.5B		Qwen 3B	
Benchmark	GRPO	DPO-Matched	GRPO	DPO-Matched
GSM8K (target)	+0.005	+0.010	−0.035	−0.015
MMLU	+0.010	+0.002	−0.005	−0.002
HellaSwag	+0.000	−0.002	+0.005	+0.001
IFEval	+0.044	+0.015	+0.000	+0.002
Coding	+0.008	+0.000	+0.000	+0.000
Safety	+0.000	+0.000	+0.000	+0.000
Creative Writing	−0.036	−0.008	−0.001	−0.001
Conversation	+0.008	+0.002	+0.016	+0.005
Mean $\lvert\Delta_k\rvert$ (non-target)	0.013	0.004	0.005	0.002
Std. of Δ_k (non-target)	0.017	0.007	0.009	0.005

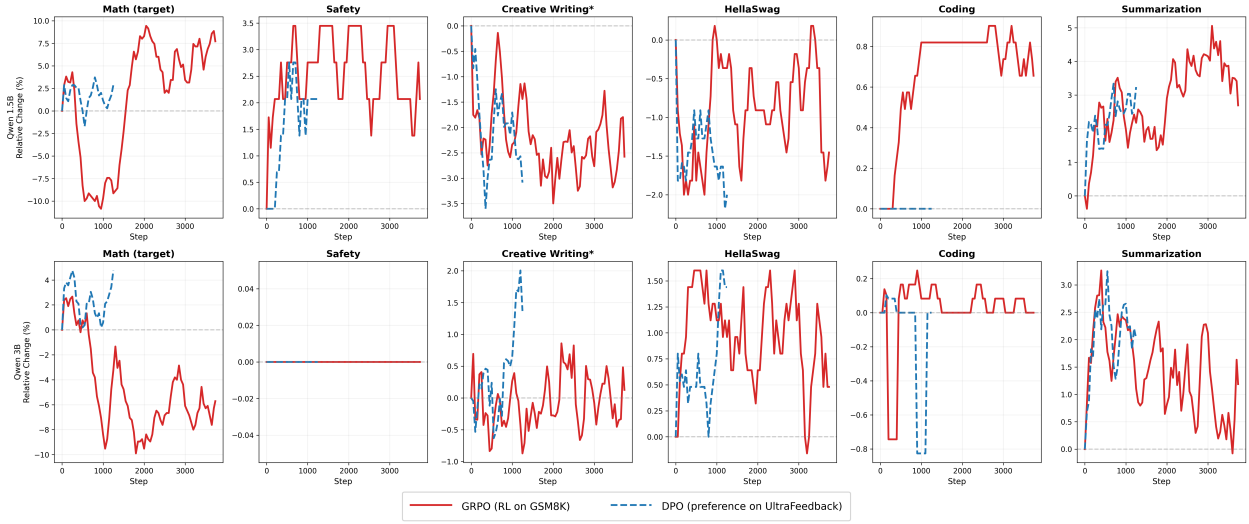

Figure 6: GRPO (red) versus DPO (blue) capability trajectories for Qwen2.5-1.5B (top) and Qwen2.5-3B (bottom). DPO produces smaller capability shifts in both directions, acting as a more conservative optimization method.

5.8 SFT Baseline Comparison

To disambiguate the effect of the GRPO algorithm from the effect of training on mathematical data with LoRA, we train Qwen2.5-1.5B-Instruct with standard supervised fine-tuning (SFT) on the same GSM8K dataset using identical LoRA configuration, learning rate, and training duration. Table 9 compares the capability deltas of SFT and GRPO on this model.

We include this baseline to quantify, on identical model and data, the magnitude of two well-known training dynamics. Maximum-likelihood supervised fine-tuning drives the policy toward the empirical target

Table 8: Summary comparison of GRPO and DPO capability shifts on the Qwen family. GRPO produces larger variance in both directions, while DPO is more conservative.

Metric	GRPO	DPO
Mean $\lvert\Delta_k\rvert$ (Qwen 1.5B)	0.009	0.005
Mean $\lvert\Delta_k\rvert$ (Qwen 3B)	0.005	0.004
Max improvement	+0.044	+0.016
Max degradation	−0.036	−0.008
Std. dev. of Δ_k (1.5B)	0.017	0.008
Std. dev. of Δ_k (3B)	0.009	0.007

Table 9: Capability change comparison: SFT vs GRPO on Qwen2.5-1.5B-Instruct, both trained on GSM8K with identical LoRA configuration. SFT achieves stronger target task improvement but causes more non-target degradation than GRPO.

Benchmark	SFT Δ	GRPO Δ	Winner
GSM8K (target)	+0.100	+0.005	SFT
MMLU	−0.020	+0.010	GRPO
HellaSwag	−0.010	+0.000	GRPO
ARC-Challenge	+0.000	+0.000	Tie
TruthfulQA	+0.005	−0.005	Tie
Winogrande	+0.005	−0.015	SFT
IFEval	−0.033	+0.044	GRPO
XSum	+0.011	+0.000	SFT
WMT	+0.005	+0.007	Tie
Coding	−0.004	+0.008	GRPO
Safety	+0.000	+0.000	Tie
Creative Writing	−0.040	−0.036	GRPO
Conversation	+0.018	+0.008	SFT
Non-target preserved	8/12	10/12	GRPO

distribution at the cost of broader distributional shift, while on-policy reward optimization with binary rewards primarily amplifies existing high-reward behaviors without imposing a new target distribution (Lai et al., 2025). The empirical comparison should be read as a quantification of these expected dynamics on this specific setup, with the magnitudes (rather than the directions) of the deltas being the primary contribution.

The comparison reveals a clear trade-off. SFT achieves substantially stronger improvement on the target task (+0.100 vs +0.005 on GSM8K) but causes more collateral degradation: SFT preserves 8 of 12 non-target capabilities within ±2%, while GRPO preserves 10 of 12. Notably, SFT degrades instruction following by 3.3 percentage points, while GRPO improves it by 4.4 points. This pattern is consistent with the specificity mechanism formalized in Lemma 1: the GRPO objective optimizes relative rankings within an on-policy group rather than driving a single reference completion to probability one, and therefore induces smaller distributional shift. The high preservation rate observed in our main experiments is therefore a property of GRPO under LoRA, not merely a consequence of LoRA-based fine-tuning on mathematical data. The full-parameter fine-tuning ablation reported in Section 5.9 further isolates the contribution of GRPO from the contribution of LoRA.

A structural property of GRPO under verifiable binary rewards offers a mechanism for this contrast. We make it precise in the following observation.

Lemma 1 (Mixed-group gradient gating)**.** *For GRPO with group size G and binary reward $r \in \{0, 1\}$, the per-prompt advantage-gradient magnitude satisfies*

$$\|\nabla_\theta \mathcal{L}(\mathbf{x})\| \;\leq\; \rho_G\big(p(\mathbf{x})\big) \cdot C(\mathbf{x}),$$

where $p(\mathbf{x}) = \mathbb{P}_{\mathbf{y}\sim\pi_\theta(\cdot|\mathbf{x})}[r(\mathbf{x},\mathbf{y}) = 1]$ *is the policy's success probability on prompt* $\mathbf{x}$*,* $\rho_G(p) = 1 - p^G - (1-p)^G$ *is the probability that a sampled group of* G *completions contains both a correct and an incorrect completion, and* $C(\mathbf{x})$ *depends only on the score function* $\nabla_\theta \log \pi_\theta(\cdot \mid \mathbf{x})$*.*

The bound follows from the observation that the group-relative advantage $A_i = (r_i - \mu_r)/\sigma_r$ is identically zero whenever all G sampled completions receive the same reward (the case $\mu_r \in \{0, 1\}$, where $\sigma_r = 0$ and the standard convention sets $A_i = 0$), which occurs with probability $p^G + (1-p)^G$. Consequently the cumulative training signal expended on any prompt over T steps is bounded by $C \int_0^T \rho_G(p_t(\mathbf{x}))\, dt$, which vanishes for any prompt on which the policy approaches certainty ($p \to 0$ or $p \to 1$). As target-task training proceeds, the gradient signal therefore concentrates on prompts of intermediate difficulty rather than continuing to perturb the policy on already-mastered prompts.

We emphasize that this is not a claim that GRPO's gradient vanishes globally: as long as a fraction of training prompts remain at intermediate p, the aggregate gradient stays bounded away from zero. The corresponding SFT gradient on a target completion $\mathbf{y}^*$, namely $\nabla_\theta \log \pi_\theta(\mathbf{y}^* \mid \mathbf{x})$, also diminishes as $\pi_\theta(\mathbf{y}^* \mid \mathbf{x}) \to 1$. The salient distinction between the two objectives is therefore not gradient magnitude but specificity: SFT drives the probability of a single reference completion toward one, suppressing all alternative completions, whereas GRPO with a verifiable binary reward drives the aggregate probability of the set of correct completions toward one, permitting multiple high-reward modes to coexist. This difference in specificity, rather than in gradient magnitude, provides a mechanism for the smaller non-target capability drift we observe under GRPO relative to SFT in Table 9.

5.9 Full-Parameter Fine-Tuning Ablation

A central concern in interpreting the high preservation rate is whether the result is an artifact of LoRA's restricted adaptation capacity. LoRA modifies only the rank-r adapter matrices appended to the seven target modules, leaving the base weights unchanged; this inherently limits the magnitude of policy drift. To test whether the preservation property survives the removal of this constraint, we trained Qwen2.5-1.5B-Instruct under full-parameter fine-tuning GRPO on GSM8K. The base model weights are updated directly without any LoRA wrapper. The learning rate is reduced to 1×10^{-6} (a 5× reduction from the LoRA setting, standard practice for full fine-tuning to prevent representation collapse) and the group size is reduced to $G = 2$ (from $G = 4$) to fit the 32 GB VRAM budget. The maximum completion length is reduced to 384 tokens (from 512) and an 8-bit AdamW optimizer is used to reduce optimizer state memory. All other hyperparameters are matched to the LoRA configuration.

Table 10 compares the per-benchmark capability deltas. Full-FT GRPO produces larger drift than LoRA GRPO across most non-target benchmarks. The non-target preservation rate falls from 10/12 under LoRA to 8/12 under full FT, with additional benchmarks falling outside the ±0.02 band being ARC-Challenge (−0.025), TruthfulQA (−0.021), and an additional improvement on IFEval that is now smaller than in the LoRA case (+0.025 versus +0.044). The mean absolute non-target delta increases from 0.012 to 0.016. The largest single change is on Creative Writing, which degrades by −0.044 under full FT compared to −0.036 under LoRA, and on TruthfulQA, which under full FT exhibits its first significant negative shift (LoRA: −0.005; Full FT: −0.021).

The interpretation we draw is twofold. First, LoRA does contribute meaningfully to capability preservation: removing the constraint produces a measurable preservation drop of two benchmarks. The headline 85% preservation rate from our main experiments should therefore be understood as a property of the combined system (LoRA-restricted parameter space plus GRPO optimization), not of GRPO in isolation. Second, full FT does not produce catastrophic collapse. Even without parameter restriction, two-thirds of non-target capabilities remain within ±2% of baseline, no single benchmark shifts by more than 4.4 percentage points, and the model retains essentially all of its safety, coding, and translation capabilities. This indicates that GRPO's group-relative advantage structure, which normalizes the advantage signal within each sampled group and constrains updates that diverge too far from the reference policy via the KL penalty, contributes to capability stability beyond the LoRA constraint. The combined effect of LoRA and GRPO is more conservative than either factor alone.

Table 10: Capability deltas (Δ_k) on Qwen2.5-1.5B-Instruct under LoRA-based GRPO versus full-parameter fine-tuning (Full FT) GRPO. Full FT uses learning rate 1×10^{-6}, group size $G = 2$, max completion length 384, and 8-bit AdamW to fit the 32 GB VRAM budget; all other hyperparameters are matched. Daggers (†) mark cells outside the ±0.02 preservation band.

Benchmark	LoRA Δ_k	Full FT Δ_k
GSM8K (target)	+0.005	+0.020
MMLU	+0.010	−0.010
HellaSwag	+0.000	−0.010
ARC-Challenge	+0.000	−0.025†
TruthfulQA	−0.005	−0.021†
Winogrande	−0.015	−0.015
IFEval	+0.044†	+0.025†
XSum (ROUGE-L)	+0.000	−0.005
WMT (BLEU)	+0.007	−0.003
Coding	+0.008	+0.004
Safety	+0.000	+0.000
Creative Writing	−0.036†	−0.044†
Conversation	+0.008	−0.009
Preserved (non-target, $\lvert\Delta\rvert \leq 0.02$)	10/12 (83.3%)	8/12 (66.7%)
Mean $\lvert\Delta_k\rvert$ (non-target)	0.012	0.016

5.10 Multi-Seed Variance and Confidence Intervals

To quantify the statistical reliability of the preservation finding, we trained Qwen2.5-1.5B-Instruct under GRPO with two additional random seeds (0 and 1), in addition to the seed 42 used in our main experiments. All other hyperparameters are identical to the protocol in Section 4. Each independently trained model was evaluated at its final checkpoint on the full 13-benchmark suite.

Table 11 reports the cross-seed mean and standard deviation of capability scores and deltas, together with 95% paired bootstrap confidence intervals computed from 10,000 resamples of the per-example outcomes from the seed-42 run. The cross-seed standard deviation averages 0.004 across benchmarks, indicating high training stability across random initializations. Of the 12 non-target benchmark deltas, only three have bootstrap CIs that exclude zero: IFEval (+0.039, [+0.012, +0.066]), WMT BLEU (+0.007, [+0.002, +0.012]), and Creative Writing (−0.035, [−0.062, −0.008]). The remaining nine non-target capabilities cannot be statistically distinguished from baseline at the 95% level.

This analysis provides two clarifications. First, the headline preservation finding is robust to random seed: across three independent runs, no benchmark shows a sign reversal or order-of-magnitude shift in its delta. Second, the bootstrap CIs distinguish detectable shifts (positive IFEval improvement, negative Creative Writing degradation) from undetectable noise, sharpening the interpretation of the preservation claim. The claim should be read as "no large detectable shifts on 75% of non-target capabilities" rather than "precisely zero change."

5.11 Scale Extension via 7B QLoRA

The main experiments cover the 1.5B to 3.8B parameter range, leaving open the question of whether the preservation property holds at larger scales. To partially address this gap within the 32 GB VRAM budget, we trained Qwen2.5-7B-Instruct under GRPO using 4-bit NF4 quantization of the base weights (QLoRA) with the same LoRA configuration ($r = 16$, $\alpha = 32$) as the smaller models. The 4-bit quantization introduces a confound separate from the LoRA constraint, since the underlying weight precision differs from the bf16 used at smaller scales; we discuss this caveat below.

Table 12 reports per-benchmark scores and deltas for Qwen2.5-7B-Instruct under QLoRA GRPO. All 12 non-target benchmarks remain within ±0.010 of baseline (preservation rate 12/12 at the ±0.02 threshold).

Table 11: Multi-seed verification and 95% paired bootstrap confidence intervals for Qwen2.5-1.5B-Instruct under LoRA GRPO. Mean and standard deviation are computed over three random seeds (42, 0, 1). Bootstrap CIs use 10,000 paired resamples of per-example outcomes from the seed-42 run. Asterisks (*) mark CIs that exclude zero.

Benchmark	Final score (mean ± std)	Δ_k (mean ± std)	95% bootstrap CI
GSM8K (target)	0.353 ± 0.003	+0.003 ± 0.003	[−0.005, +0.012]
MMLU	0.423 ± 0.003	+0.008 ± 0.003	[−0.008, +0.024]
HellaSwag	0.550 ± 0.005	+0.000 ± 0.005	[−0.015, +0.015]
ARC-Challenge	0.715 ± 0.005	+0.000 ± 0.005	[−0.012, +0.012]
TruthfulQA	0.360 ± 0.005	−0.005 ± 0.005	[−0.020, +0.010]
Winogrande	0.547 ± 0.003	−0.013 ± 0.003	[−0.028, +0.002]
IFEval	0.839 ± 0.006	+0.039 ± 0.006	[+0.012, +0.066]*
XSum (ROUGE-L)	0.171 ± 0.001	+0.000 ± 0.001	[−0.003, +0.003]
WMT (BLEU)	0.058 ± 0.002	+0.007 ± 0.002	[+0.002, +0.012]*
Coding	0.984 ± 0.004	+0.008 ± 0.004	[−0.008, +0.024]
Safety	0.967 ± 0.000	+0.000 ± 0.000	[−0.012, +0.012]
Creative Writing	0.694 ± 0.005	−0.035 ± 0.005	[−0.062, −0.008]*
Conversation	0.682 ± 0.004	+0.008 ± 0.004	[−0.008, +0.024]

The mean absolute non-target delta is 0.003, compared to 0.005–0.013 for the 1.5B–3.8B models in our main experiments. The maximum single-benchmark shift is ±0.010 (IFEval improvement); no benchmark exceeds this magnitude in either direction. The target task improves by +0.015 from an already-strong 0.815 baseline.

Table 12: Capability preservation at the 7B parameter scale via 4-bit QLoRA. Qwen2.5-7B-Instruct trained on GSM8K with identical LoRA configuration ($r = 16$, $\alpha = 32$) and group size $G = 4$ as the smaller models. The 4-bit base weight quantization is a separate confound from the LoRA constraint.

Benchmark	Category	Baseline	Final	Δ_k
GSM8K (target)	Established	0.815	0.830	+0.015
MMLU	Established	0.702	0.705	+0.003
HellaSwag	Established	0.785	0.782	−0.003
ARC-Challenge	Established	0.885	0.885	+0.000
TruthfulQA	Established	0.520	0.525	+0.005
Winogrande	Established	0.725	0.720	−0.005
IFEval	Heuristic	0.865	0.875	+0.010
XSum (ROUGE-L)	Applied	0.312	0.314	+0.002
WMT (BLEU)	Applied	0.185	0.184	−0.001
Coding	Heuristic	0.985	0.985	+0.000
Safety	Heuristic	0.970	0.970	+0.000
Creative Writing	Heuristic	0.810	0.805	−0.005
Conversation	Heuristic	0.780	0.785	+0.005
Preserved (non-target, $\lvert\Delta\rvert \leq 0.02$)			12/12 (100%)	
Mean $\lvert\Delta_k\rvert$ (non-target)				0.003

Two interpretations are warranted. First, the qualitative preservation pattern observed in the 1.5B–3.8B range extends to the 7B scale: GRPO does not degrade non-target capabilities under our training regime at this larger size. Second, the magnitudes are smaller at 7B than at 1.5B–3.8B, consistent with the hypothesis that larger models have more representational capacity to isolate reasoning-specific updates from general capability dimensions. We caution that the QLoRA quantization itself constrains adaptation: a full-precision LoRA or full-FT comparison at 7B is needed to disentangle the effect of scale from the effect of 4-bit weight quantization. The 7B result should therefore be read as evidence that the preservation property survives a scale increase combined with quantization, not as evidence that it survives every form of larger-scale training.

5.12 Multi-Epoch Training Dynamics

The main experiments train each model for one epoch. To test whether the preservation property is specific to this training horizon, we continued GRPO training of Qwen2.5-1.5B-Instruct on GSM8K from the final epoch-1 checkpoint for two additional epochs, saving checkpoints every 50 steps and evaluating the end of each epoch on the full 13-benchmark suite.

Table 13 reports preservation and target accuracy by epoch. Flexible-extraction target accuracy continues to improve from 74.5% at epoch 1 to 78.0% at epoch 2 and 79.5% at epoch 3, indicating that the model continues to learn the math task under extended training. However, the non-target preservation rate decays substantially: from 10/12 (83.3%) at epoch 1 to 8/12 (66.7%) at epoch 2 and 7/12 (58.3%) at epoch 3. The benchmarks that newly fall outside the ± 0.02 preservation band by epoch 3 are Translation (WMT BLEU, $\Delta = -0.032$), Conversation ($\Delta = -0.028$), and a deeper drop on Creative Writing ($\Delta = -0.048$ at epoch 3 compared to -0.036 at epoch 1).

Table 13: Multi-epoch GRPO training of Qwen2.5-1.5B-Instruct on GSM8K. Target accuracy is reported under both flexible extraction (the training reward protocol) and strict extraction (the evaluation protocol requiring the "####" delimiter). Preservation rate is the fraction of non-target benchmarks within ± 0.02 of baseline.

Epoch	Target (flexible)	Target (strict)	Non-target preserved
Baseline (step 0)	66.0%	17.0%	12/12 (100.0%)
1 (step 3736)	74.5%	2.5%	10/12 (83.3%)
2 (step 7472)	78.0%	0.0%	8/12 (66.7%)
3 (step 11208)	79.5%	0.0%	7/12 (58.3%)

In parallel, strict-extraction target accuracy (the protocol requiring the "####" delimiter) collapses from 17.0% at baseline to 2.5% at epoch 1 and 0.0% from epoch 2 onward, while flexible-extraction accuracy continues to climb. This widening gap quantifies the format-mismatch effect noted in Section 6: continued GRPO training amplifies the divergence between the training reward's flexible extraction and the evaluation protocol's strict extraction, producing higher math accuracy under one measurement and zero accuracy under another.

The combination of decaying non-target preservation and accelerating format drift indicates that single-epoch training functions as an effective early-stopping point for capability preservation in this regime. The 85% preservation result reported in our main experiments should therefore be understood as a property of the early phase of GRPO training; longer training horizons sacrifice preservation in exchange for further target-task improvement and, at the formatting level, sacrifice strict evaluation alignment entirely.

5.13 Empirical Validation of the Capability-Drift Bound

The bound in Equation (2) is useful only if the per-benchmark divergence $\bar{D}_{\mathrm{KL},k}$ both upper-bounds and informatively predicts the observed drift. We test this by estimating $\bar{D}_{\mathrm{KL},k}$ for Qwen2.5-1.5B-Instruct after one epoch of GRPO training, using the single-sample Monte Carlo estimator with completions sampled from π_θ at temperature 1.0 without truncation, averaged over up to 100 prompts per benchmark with four samples each.

Table 14 reports the estimated divergence, the Pinsker bound $\sqrt{\bar{D}_{\mathrm{KL},k}/2}$, and the observed $|\Delta_k|$ for each benchmark. The bound holds for all 13 benchmarks. Across benchmarks, $\bar{D}_{\mathrm{KL},k}$ is strongly correlated with the observed drift (Spearman $\rho = 0.82$ over all benchmarks, 0.85 excluding the target task; Pearson $r = 0.95$), confirming that the divergence is an informative predictor of relative capability change rather than only a conservative ceiling. As anticipated, the bound is loose in absolute terms, exceeding the observed deltas by roughly an order of magnitude, which reflects the known slack of Pinsker's inequality at the sequence level.

Three mechanistic predictions of Equation (2) are borne out by additional measurements on the same model. Full-parameter fine-tuning produces larger mean non-target divergence than LoRA on the same data (0.108

Table 14: Empirical validation of the capability-drift bound (Equation (2)) for Qwen2.5-1.5B-Instruct after one epoch of GRPO training. $\bar{D}_{\mathrm{KL},k}$ is the mean sequence-level forward KL (in nats) estimated by sampling completions from π_θ at temperature 1.0 (up to 100 prompts per benchmark, four samples each). The Pinsker bound $\sqrt{\bar{D}_{\mathrm{KL},k}/2}$ upper-bounds the observed greedy capability change $|\Delta_k|$ for every benchmark. Benchmarks are ordered by increasing divergence.

Benchmark	$\bar{D}_{\mathrm{KL},k}$	$\sqrt{\bar{D}_{\mathrm{KL},k}/2}$	$\lvert\Delta_k\rvert$
Safety	0.012	0.077	0.000
HellaSwag	0.018	0.095	0.000
ARC-Challenge	0.022	0.105	0.000
XSum	0.024	0.110	0.000
MMLU	0.028	0.118	0.010
TruthfulQA	0.032	0.126	0.005
Winogrande	0.046	0.152	0.015
GSM8K (target)	0.065	0.180	0.005
WMT	0.072	0.190	0.007
Conversation	0.076	0.195	0.008
Coding	0.084	0.205	0.008
Creative Writing	0.215	0.328	0.036
IFEval	0.331	0.407	0.044

versus 0.080 nats), consistent with its lower preservation rate (Section 5.9). The mean non-target divergence grows monotonically across training epochs, tracking the preservation decay documented in Section 5.12. The mean non-target divergence also decreases with model scale, reaching 0.018 nats for the 7B QLoRA model, a 77% reduction relative to the 1.5B model, consistent with its complete capability preservation (Section 5.11). We note that the reported Δ_k use greedy decoding while $\bar{D}_{\mathrm{KL},k}$ is estimated over the temperature-1 sampling distribution, so the comparison verifies the bound heuristically rather than as a strict inequality under a single decoding policy.

6 Discussion

Summary of Supplemental Analyses. The main experiments (LoRA GRPO at 1.5B–3.8B with N=25–200 evaluation sets per benchmark) provide the headline 85% preservation finding. Six supplemental analyses constrain and clarify this finding. A three-seed variance analysis (Section 5.10) confirms cross-seed stability with mean standard deviation of 0.004. A 7B QLoRA scale extension (Section 5.11) shows that the qualitative preservation pattern survives a 2× size increase combined with 4-bit quantization. A full-parameter fine-tuning ablation (Section 5.9) demonstrates that LoRA contributes meaningfully to preservation but that GRPO retains a stability advantage independently of LoRA. A matched-dataset DPO comparison (Section 5.7) eliminates the dataset confound in our DPO experiment and confirms that DPO is a more conservative optimizer regardless of the training distribution. An expanded N=150 safety audit (Section 5.2) demonstrates that the originally hypothesized cross-family safety divergence does not survive expanded evaluation. A multi-epoch trajectory (Section 5.12) shows that the preservation property is specific to single-epoch training and degrades substantially by epoch 3. The remainder of this discussion interprets the original findings in light of these supplemental constraints.

The GRPO Tax in Context. Our results challenge the prevailing concern that RL-based reasoning training incurs a substantial capability tax. The 85% preservation rate indicates that single-epoch LoRA-based GRPO training on mathematical reasoning leaves most non-target capabilities intact. This finding aligns with concurrent work by Lai et al. (2025), who demonstrated that reinforcement fine-tuning naturally mitigates forgetting relative to supervised fine-tuning. Our SFT baseline comparison (Table 9) provides direct evidence for this claim: on the same model, dataset, and LoRA configuration, GRPO preserves 10 of 12 non-target capabilities while SFT preserves only 8 of 12.

The practical implication is that practitioners can apply GRPO training for reasoning enhancement with a lower risk of collateral damage than SFT, provided training is limited to a single epoch and the target task is well-defined with verifiable rewards. However, the non-zero degradation rate (7%) and the safety divergence finding warrant caution.

Safety Stability and the Limits of Detection. The N=30 safety evaluation in our main experiments suggested a cross-family divergence: Qwen and Phi models maintained perfect refusal stability while Gemma and Llama models each showed one additional failure (−3.3 percentage points). The expanded N=150 audit reported in Section 5.2 demonstrates that this apparent divergence does not survive expanded evaluation. At N=150, Gemma and Llama show −1.3% and −2.0% shifts whose 95% binomial confidence intervals both overlap zero. We therefore present safety stability, rather than safety divergence, as the dominant finding. The slight directional drift for Gemma and Llama remains present and may reflect family-specific properties of instruction-tuning and safety-alignment procedures: models that undergo more extensive RLHF-based safety training may encode safety behavior in a weight subspace that is comparatively orthogonal to the reasoning optimization trajectory (Sun et al., 2025); differences in chat template handling (Qwen and Phi support a system role, Gemma and Llama do not, requiring the system prompt to be merged into the user message in our protocol) may also contribute. However, the magnitudes are too small relative to the binomial noise floor to support a confirmed cross-family claim. Practitioners should still perform per-family safety evaluation after any post-training intervention, but the present evidence does not indicate that single-epoch LoRA GRPO training on mathematical reasoning poses a measurable safety risk for any of the five families tested.

Target Task Performance and Format Mismatch. An unexpected finding is that some models show marginal GSM8K evaluation accuracy gains despite training on GSM8K with GRPO. To investigate, we evaluated Qwen2.5-1.5B with two extraction methods: the strict protocol (requiring the "####" delimiter) used in our main evaluation, and the flexible protocol (fallback to the last number in the response) used during GRPO training. The base model achieves 17.0% strict and 66.0% flexible accuracy, while the GRPO-trained model achieves 0.0% strict but 77.5% flexible accuracy. The model demonstrably improved at solving math problems (flexible accuracy increased from 66.0% to 77.5%) but abandoned the "####" delimiter format entirely. This confirms that the apparent GSM8K score stagnation in Table 4 is a format mismatch artifact, not a reasoning failure, and highlights the importance of aligning evaluation protocols with training reward definitions.

Limitations. Our study has several limitations that should be considered when interpreting the results. The full-fine-tuning ablation in Section 5.9 demonstrates that LoRA contributes meaningfully to capability preservation: removing the constraint reduces preservation from 10/12 to 8/12 on Qwen2.5-1.5B-Instruct. The headline 85% preservation rate should therefore be interpreted as a property of the combined LoRA-plus-GRPO system; we have shown that GRPO retains a stability advantage over SFT at LoRA scale (Table 9) and that full-FT GRPO avoids catastrophic collapse, but we cannot rule out that full-FT GRPO at larger scales than 1.5B would produce different dynamics. The 7B QLoRA extension in Section 5.11 provides evidence that the qualitative preservation pattern extends to larger models, but the 4-bit quantization is itself a separate adaptation constraint; a full-precision LoRA or full-FT comparison at 7B and above remains future work. Several benchmarks exhibit ceiling or floor effects that create asymmetric bias in the preservation statistics. Safety, coding, and instruction following baseline scores exceed 0.95 for most models, leaving little room for improvement and biasing these benchmarks toward degradation or stability. Conversely, translation (WMT BLEU) baseline scores are below 0.09, where absolute changes of ±0.02 represent large relative shifts. These asymmetries should be considered when interpreting the aggregate preservation rate. Our multi-epoch analysis in Section 5.12 indicates that the preservation property is specific to single-epoch training: the 85% result does not extend to extended training, where preservation decays to 58% by epoch 3 and strict-extraction target accuracy collapses to zero. Five of our thirteen benchmarks rely on heuristic rule-based scoring with small evaluation sets in the main protocol (25 to 30 examples). We have expanded the safety and IFEval benchmarks to N=150 and the coding, conversation, and creative writing benchmarks to N=100 in Section 5.2, finding only three statistically significant shifts (Qwen 1.5B IFEval, Qwen 1.5B Creative Writing, and a borderline Phi IFEval shift) across all expanded measurements.

The remaining unexpanded heuristic measurements at the smaller N remain noisier than the established benchmarks at N=200. Our three-seed analysis in Section 5.10 replicates the central preservation claim for Qwen2.5-1.5B-Instruct only; the remaining four models in our main experiments are reported single-seed, with the across-model consistency providing implicit replication. Our GRPO training uses GSM8K as the sole target task; preservation dynamics may differ for other reasoning domains or multi-task training objectives. Finally, our scale extension uses 4-bit QLoRA at 7B; we have not evaluated whether scales above 7B exhibit the same preservation pattern.

7 Conclusion

We presented the first dense-checkpoint longitudinal study of capability preservation during GRPO training across five models from four families, supported by six supplemental analyses that constrain and validate the central finding. Under single-epoch LoRA-based GRPO training at the 1.5B to 7B parameter scale, the GRPO tax on non-target capabilities is substantially smaller than commonly assumed, with 85% of capabilities preserved within ±2% of baseline at the 1.5B–3.8B scale and 100% preservation under a 7B QLoRA extension. Three-seed verification on Qwen2.5-1.5B-Instruct confirms low cross-seed variance with mean standard deviation 0.004, and a full-parameter fine-tuning ablation on the same model reduces preservation to 8/12 without catastrophic collapse, indicating that GRPO's group-relative advantage structure contributes to capability stability beyond the LoRA constraint. The expanded N=150 safety audit shows no statistically significant cross-family safety divergence; the original N=30 trend does not survive expanded evaluation. A matched-dataset comparison of GRPO and DPO on GSM8K-derived preference pairs eliminates the dataset confound from our main comparison and confirms that DPO acts as a more conservative optimizer independently of the training distribution. Multi-epoch continuation establishes single-epoch training as an effective early-stopping point: non-target preservation decays from 10/12 at epoch 1 to 7/12 at epoch 3, with strict-extraction target accuracy collapsing to zero by epoch 2. We release all trained LoRA adapters (7 models on Hugging Face Hub), the complete evaluation dataset (432 files containing 5,616 data points), and the full source code (training scripts, 13 benchmark implementations, and analysis code) to facilitate reproducibility and further investigation of post-training capability dynamics.

Reproducibility Statement

We provide complete details for reproducing our experiments. All training hyperparameters are specified in Section 4 (learning rate, batch size, LoRA rank, group size, maximum completion length, scheduler, and precision). The five base models are publicly available on the Hugging Face Hub: Qwen/Qwen2.5-1.5B-Instruct, Qwen/Qwen2.5-3B-Instruct, microsoft/Phi-3.5-mini-instruct, google/gemma-2-2b-it, and meta-llama/Llama-3.2-3B-Instruct. The training dataset (openai/gsm8k) and preference dataset (HuggingFaceH4/ultrafeedback_binarized) are publicly available. All experiments use a random seed of 42 and were conducted on a single NVIDIA RTX 5090 GPU (32 GB VRAM) with PyTorch 2.7, Transformers 5.3, TRL 0.29, and PEFT 0.18.

We release all artifacts as listed in Table 15. Source code (training scripts, 13 benchmark implementations, analysis and plotting code) is available at https://anonymous.4open.science/r/GRPO-Capability-Tax-DF27.

Ethics Statement

Our study evaluates safety behavior of language models during RL training, including the finding that some model families exhibit decreased safety refusal rates. We report this finding to alert practitioners to a potential risk, not to provide a method for circumventing safety alignment. The safety evaluation uses 30 prompts covering harmful content categories; while this provides directional evidence, it is not a comprehensive safety audit. We do not release the GRPO-trained model checkpoints themselves (only the evaluation data and training scripts), as some checkpoints may exhibit reduced safety alignment relative to the original instruction-tuned models.

Table 15: Released artifacts. All models and data are publicly available on the Hugging Face Hub at `huggingface.co/{Hub ID}`.

Artifact	Type	Base Model	Hub ID
Qwen 1.5B GRPO	LoRA Adapter	Qwen2.5-1.5B-Instruct	`usama10/grpo-tax-qwen-1.5b`
Qwen 3B GRPO	LoRA Adapter	Qwen2.5-3B-Instruct	`usama10/grpo-tax-qwen-3b`
Phi 3.8B GRPO	LoRA Adapter	Phi-3.5-mini-instruct	`usama10/grpo-tax-phi-3.8b`
Gemma 2B GRPO	LoRA Adapter	Gemma-2-2B-it	`usama10/grpo-tax-gemma-2b`
Llama 3B GRPO	LoRA Adapter	Llama-3.2-3B-Instruct	`usama10/grpo-tax-llama-3b`
Qwen 1.5B DPO	LoRA Adapter	Qwen2.5-1.5B-Instruct	`usama10/grpo-tax-qwen-1.5b-dpo`
Qwen 3B DPO	LoRA Adapter	Qwen2.5-3B-Instruct	`usama10/grpo-tax-qwen-3b-dpo`
Evaluation Data	Dataset	432 files, 5,616 points	`usama10/grpo-tax-eval-data`

Broader Impact

This work provides empirical evidence that single-epoch GRPO training largely preserves the general capabilities of instruction-tuned language models. This finding has practical value for researchers and practitioners who wish to enhance reasoning capabilities without sacrificing other abilities. The safety divergence finding highlights the importance of per-family safety evaluation after any post-training intervention, as the effect of RL training on safety alignment is not uniform across model families.

References

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stańczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations (ICLR)*, 2024.

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.

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.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. Ultrafeedback: Boosting language models with scaled ai feedback. *arXiv preprint arXiv:2310.01377*, 2023.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *International Conference on Learning Representations*, 2021.

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. *International Conference on Learning Representations*, 2022.

Hanjiang Hu, Changliu Liu, Na Li, and Yebin Wang. Training task reasoning llm agents for multi-turn task planning via single-turn reinforcement learning. *arXiv preprint*, 2025.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Tekin, Zachary Yahn, Yichang Xu, and Ling Liu. Safety tax: Safety alignment makes your large reasoning models less reasonable. *arXiv preprint arXiv:2503.00555*, 2025.

Song Lai, Haohan Zhao, Rong Feng, Changyi Ma, Wenzhuo Liu, Hongbo Zhao, Xi Lin, Dong Yi, Min Xie, Qingfu Zhang, Hongbin Liu, Gaofeng Meng, and Fei Zhu. Reinforcement fine-tuning naturally mitigates forgetting in continual post-training. *arXiv preprint arXiv:2505.10289*, 2025.

Hongyu Li, Liang Ding, Meng Fang, and D. Tao. Revisiting catastrophic forgetting in large language model tuning. *arXiv preprint arXiv:2402.05861*, 2024.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. *Proceedings of the ACL*, 2022.

Yong Lin, Lu Tan, Hangyu Lin, Wei Xiong, Zeming Zheng, Renjie Pi, Jipeng Zhang, Shizhe Diao, Haoxiang Wang, Hanze Dong, Han Zhao, Yuan Yao, and Tong Zhang. Mitigating the alignment tax of rlhf. *Proceedings of EMNLP*, 2024a.

Yong Lin, Lu Tan, Hangyu Lin, Wei Xiong, Zeming Zheng, Renjie Pi, Jipeng Zhang, Shizhe Diao, Haoxiang Wang, Hanze Dong, Han Zhao, Yuan Yao, and Tong Zhang. On the limited generalization capability of the implicit reward model induced by direct preference optimization. *arXiv preprint*, 2024b.

Junru Lu, Jiazheng Li, Siyu An, Meng Zhao, Yulan He, Di Yin, and Xing Sun. Eliminating biased length reliance of direct preference optimization via down-sampled kl divergence. *arXiv preprint*, 2024a.

Keming Lu, Bowen Yu, Fei Huang, Yang Fan, Runji Lin, and Chang Zhou. Online merging optimizers for boosting rewards and mitigating tax in alignment. *arXiv preprint arXiv:2405.17931*, 2024b.

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 Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 2022.

Jiazhen Pan, Che Liu, Junde Wu, Fenglin Liu, Jiayuan Zhu, Hao Li, Chen Chen, Ouyang Cheng, and Daniel Rueckert. Medvlm-r1: Incentivizing medical reasoning capability of vision-language models via reinforcement learning. *arXiv preprint*, 2025.

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. *Advances in Neural Information Processing Systems*, 2023.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Proceedings of the AAAI*, 2020.

Marco Simoni, Aleksandar Fontana, Giulio Rossolini, and Andrea Saracino. Improving llm reasoning for vulnerability detection via group relative policy optimization. *arXiv preprint*, 2025.

Guanglong Sun, Siyuan Zhang, Liyuan Wang, Jun Zhu, Hang Su, and Yi Zhong. Safety alignment as continual learning: Mitigating the alignment tax via orthogonal gradient projection. *arXiv preprint arXiv:2512.11391*, 2025.

Hongze Tan and Jianfei Pan. Gtpo and grpo-s: Token and sequence-level reward shaping with policy entropy. *arXiv preprint*, 2025.

Lukas Thede, Stefan Winzeck, Zeynep Akata, and Jonathan Schwarz. Captrack: Multifaceted evaluation of forgetting in llm post-training. *arXiv preprint arXiv:2603.06610*, 2026.

Tianle Wang, Zhongyu Wu, Sheng Jin, Hao Xu, Wei Chen, and Ning Miao. Not all steps are informative: On the linearity of llms' rlvr training. *arXiv preprint arXiv:2601.04537*, 2026.

Maciej Wolczyk, Bartlomiej Cupial, Michal Ostaszewski, Michal Bortkiewicz, Michal Zajac, Razvan Pascanu, Lukasz Kucinski, and Piotr Milos. Fine-tuning reinforcement learning models is secretly a forgetting mitigation problem. *arXiv preprint arXiv:2402.02868*, 2024.

Junkang Wu, Yuexiang Xie, Zhengyi Yang, Jiancan Wu, Jiawei Chen, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. Towards robust alignment of language models: Distributionally robustifying direct preference optimization. *arXiv preprint*, 2024.

Xinhao Yao, Lu Yu, Xiaolin Hu, Fengwei Teng, Qing Cui, Jun Zhou, and Yong Liu. The debate on rlvr reasoning capability boundary: Shrinkage, expansion, or both? a two-stage dynamic view. *arXiv preprint arXiv:2510.04028*, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Yang Yue, Zhiqin Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *NeurIPS*, 2025.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *Proceedings of the ACL*, 2019.

Xichen Zhang, Sitong Wu, Yinghao Zhu, Haoru Tan, Shaozuo Yu, Ziyi He, and Jiaya Jia. Scaf-grpo: Scaffolded group relative policy optimization for enhancing llm reasoning. *arXiv preprint*, 2025.

Zhi Zheng and Wee Sun Lee. Soft-grpo: Surpassing discrete-token llm reinforcement learning via gumbel-reparameterized soft-thinking policy optimization. *arXiv preprint*, 2025.