

How Much Supervised Fine-Tuning Is Enough? Dissecting the Impact of SFT Extent on Downstream Reinforcement Learning Performance

Shirley Kokane
Carnegie Mellon University

Abstract

Large Language Models (LLMs) are typically adapted to downstream tasks through Supervised Fine-Tuning (SFT), where Cross Entropy (CE) loss is used to maximize the likelihood of ground-truth tokens. While effective for aligning with observed data and establishing strong domain grounding, SFT alone often limits the model’s ability to explore alternative plausible outputs, potentially narrowing its decision space. Conversely, Reinforcement Learning (RL) enables reward-driven exploration and refinement but relies heavily on a well-aligned initialization to achieve stable and sample-efficient training. In real-world deployments, practitioners frequently grapple with the question: How much SFT is sufficient, and when should RL take over? This work investigates the complementary trade-off between SFT and RL approaches such as GRPO, focusing on how both stages jointly influence model exploration and exploitation. We provide a quantitative analysis across multiple domains to understand how varying the extent of SFT influences a model’s exploratory behavior and downstream RL effectiveness. Our findings aim to guide in determining optimal SFT checkpoints (CKPTs) that balance domain alignment with exploration capacity, thereby maximizing final domain performance while minimizing overfitting after the joint post-training. Our data and code are open source at [GitHub](#).

1 Introduction

Foundation model development has been empowering applications in language, vision, robotics and beyond (Li et al., 2019; Sun et al., 2019). These models usually go through a multi-step process until its finally applied to utility in a specific domain. Initially a base (or a pre-trained model) is usually created by pre-training on large-scale text and image data (Radford et al., 2018; Dai and Le, 2015; Devlin et al., 2019). The primary goal of pre-training is to encode the world’s (in a narrow

sense, the Internet’s) knowledge into the model, the model is trained to predict the next token over many different sequences prior to it. However, while a base model can be very knowledgeable, the training objective of simply predicting the next token also makes it less useful in most applications.

In order to make the LLM actually helpful in answering questions, we usually take a base model and go through post-training. Post-training aims to align model behavior with desirable properties such as helpfulness, honesty, and harmlessness, while also enhancing task-specific capabilities beyond those acquired during pre-training (Qiu et al., 2025; Parthasarathy et al., 2024). Prior work has explored improvements to individual post-training methods, for example by augmenting the standard SFT objective with entropy regularization to preserve output diversity (Li et al., 2024), or by leveraging self-generated data through self-play to iteratively extend performance beyond the limits of static human-annotated datasets (Chen et al., 2024). Other studies have examined the comparative effects of SFT and RL on properties such as generalization and memorization (Shenfeld et al., 2025; Chen et al., 2025). However, in practical deployments (Touvron et al., 2023b), these methods are typically applied in sequence as part of a unified post-training pipeline. Despite this widespread practice, there remains limited systematic understanding of how one stage influences the effectiveness of the subsequent stage, or how to monitor and optimize this interaction. In particular, relatively little work has jointly analyzed the combined effects of SFT and RL within a single framework.

In this work, we demonstrate that both SFT and RL are necessary to achieve strong domain expertise, while also identifying how to determine the optimal extent of SFT for a given dataset. We argue that post-training outcomes depend on the model’s pretraining knowledge and the entropy of the supervision data, and we introduce quantitative

indicators to characterize this interaction. Through controlled ablations on diversity, entropy, and accuracy, we empirically analyze how the combined SFT+RL pipeline shapes final model performance.

Our main contributions are:

- We systematically quantify the effect of SFT on token-level probability concentration and reduced generative stochasticity is based on the inherent capacity of the base model and also the dataset complexity.
- We study the extent of model drift from the pretrained model with its changes in output diversity, entropy, and exploration capacity.
- Through comprehensive empirical findings, we propose practical indicators for determining when it is more beneficial to transition from SFT to RL fine-tuning.

2 Related Work

Pretraining vs. Post-training Modern LLMs rely on a two-stage training paradigm. Pretraining on massive web-scale corpora enables models to acquire broad linguistic, factual, and reasoning capabilities by optimizing next-token prediction objectives (Radford et al., 2019; Brown et al., 2020; Chowdhery et al., 2023; Touvron et al., 2023a; Achiam et al., 2023). However, pretrained models often lack task alignment, controllability, and domain specialization. Post-training methods have therefore emerged as critical stages for aligning pretrained models with human preferences and downstream task requirements (Ouyang et al., 2022; Bai et al., 2022). Recent work has shown that strong pretrained representations significantly improve post-training efficiency and downstream performance scaling (Kaplan et al., 2020; Hoffmann et al., 2022; Wei et al., 2022), while other studies demonstrate that post-training is essential for instruction following, safety, and domain adaptation even when pretraining is extensive (Sanh et al., 2021; Chung et al., 2024; Touvron et al., 2023a). In this work, we propose how to optimally monitor the jointly used post-training methods RL and SFT.

SFT vs. RL Prior studies indicate that SFT is limited by its reliance on fixed supervision distributions, which can restrict exploration and propagate annotation biases, often leading to sharper and less stochastic output distributions (Gururangan et al., 2020; Sanh et al., 2021; Chung et al., 2024). In contrast, RL enables optimization directly against re-

ward signals, allowing models to discover solutions beyond demonstration data and to explicitly penalize undesirable responses (Stiennon et al., 2020; Bai et al., 2022; Rafailov et al., 2023; Chu et al., 2025). RL also introduces on-policy sampling, which exposes models to their own failure modes and provides corrective training signals that are absent in SFT. Recent work has also explored orthogonal improvements to each paradigm individually: for SFT, methods such as entropy-augmented objectives and diversity-enhanced supervision aim to improve generalization and mitigate distributional collapse (Li et al., 2024; Chen et al., 2024), while for RL, classical policy-gradient approaches such as REINFORCE (Williams, 1992) and emerging reward-shaping techniques seek to improve learning stability, and alignment quality.

Nevertheless, empirical studies consistently demonstrate that models trained using staged pipelines combining SFT followed by RL achieve superior performance, controllability, and alignment compared to models trained using either paradigm in isolation (Ouyang et al., 2022; Bai et al., 2022). We aim to establish a principled framework for determining how to optimally balance imitation-based learning and reinforcement-based optimization by analyzing the consequences of over-reliance on either stage in isolation.

Through comprehensive empirical analysis, we show that there exists an optimal post-SFT regime in which models are best suited for subsequent RL alignment. We characterize this regime using a proposed metric, the Rigidity Index (an inverse proxy for entropy), and identify an inflection point in the accuracy–rigidity curve where rigidity begins to increase faster than accuracy. We demonstrate that initializing RL from CKPTs prior to this point yields limited performance gains, while initializing beyond it produces only marginal accuracy improvements at the cost of substantial losses in exploration capacity and output diversity.

Beyond identifying the optimal SFT regime for RL initialization, our analysis reveals several previously underexplored dynamics of staged post-training. We show that model rigidity serves as a predictive indicator of downstream RL improvement potential, with excessive rigidity limiting achievable gains under fixed RL budgets. Furthermore, diversity losses incurred during heavy SFT persist throughout RL training and are not recovered by reward-driven optimization. We also observe reduced sensitivity to sampling temperature

in highly fine-tuned models, indicating constrained generative support even under stochastic decoding. Finally, by analyzing KL divergence relative to the pretrained model, we demonstrate that greater departure from the base model will lead to a greater distributional shift and would not necessarily translate to improved task performance.

3 Experimental SetUp

3.1 Tasks and Dataset

We consider two domains: Math and Science. For each domain, the available data is partitioned into three disjoint subsets: one used for training with SFT method, one used for the corresponding RL training and one used as a test set. Benchmark datasets and the partitioned test set remain fixed across all experiments, enabling consistent performance comparison and qualitative analysis across different training configurations.

In the Math domain, models are trained using the DeepMath (He et al., 2025) dataset (7k samples) and evaluated on the held-out DeepMath test set (2k samples) as well as two additional benchmarks: Olympiad Bench (He et al., 2024), and MATH-500 (Hendrycks et al., 2021).

In the Science domain, models are trained on OpenScience (NVIDIA Corporation, 2025) data (4k samples) and evaluated on the OpenScience held-out test set (1k samples), and GPQA-Diamond (Rein et al., 2024). All experiments are conducted using two model variants: DeepSeek-R1-Distill-Qwen 1.5B (DQ-1.5B) and DeepSeek-R1-Distill-Qwen 7B (DQ-7B) (Guo et al., 2025).

Different training hyperparameters, including learning rate schedules, weight decay, and advantage estimation strategies, can significantly influence the trade-off between downstream task performance and model rigidity. To systematically study this relationship, we train multiple model variants across diverse hyperparameter settings and select models that outperform less fine-tuned CKPTs in domain accuracy while minimizing collapse in token-level stochasticity. We identify the Pareto frontier between task performance and model rigidity, representing the set of models for which any further improvement in task accuracy necessarily results in significant increase in rigidity.

3.2 Metric Definitions

Domain accuracy alone does not fully capture the effects of extensive SFT on model behavior. To

better understand these impacts, we additionally evaluate two complementary metrics: the KL divergence with respect to the pretrained model and a newly proposed metric, the Rigidity Index. KL divergence quantifies the extent to which fine-tuning causes the model to deviate from the pretrained distribution, serving as a proxy for the loss of generalization and out-of-distribution robustness.

3.2.1 Rigidity Index

We quantify model rigidity at decoding time using *Rigidity Index*, which measures the concentration of probability mass among the top- k tokens at each decoding step. Intuitively, higher rigidity corresponds to lower entropy and reduced exploratory capacity during generation.

Let $\mathbf{z}_{b,t} \in \mathbb{R}^V$ denote the model logits over a vocabulary of size V for batch element b at decoding step t . We first compute log-probabilities using the log-softmax:

$$\log p_{b,t}(v) = \log \frac{\exp(z_{b,t,v})}{\sum_{v'=1}^V \exp(z_{b,t,v'})}. \quad (1)$$

Let $\mathcal{K}_{b,t}$ denote the set of the top- k tokens with the highest log-probabilities at step t . We define the normalized top- k probability distribution as:

$$\tilde{p}_{b,t}(v) = \frac{p_{b,t}(v)}{\sum_{v' \in \mathcal{K}_{b,t}} p_{b,t}(v')} \quad \text{for } v \in \mathcal{K}_{b,t}. \quad (2)$$

The entropy over the top- k distribution is then computed as:

$$H_{b,t} = - \sum_{v \in \mathcal{K}_{b,t}} \tilde{p}_{b,t}(v) \log \tilde{p}_{b,t}(v). \quad (3)$$

We normalize the entropy by its maximum value $\log k$ and define the Rigidity Index as:

$$R_{b,t} = 1 - \frac{H_{b,t}}{\log k}. \quad (4)$$

Finally, the overall Rigidity Index is the average over all decoding steps and batch elements:

$$\text{Rigidity} = \frac{1}{BT} \sum_{b=1}^B \sum_{t=1}^T R_{b,t}. \quad (5)$$

3.2.2 Inflection Point Calculation

Since accuracy and the Rigidity Index operate on different numerical scales, we first apply *min-max normalization* to rescale both metrics to a common

range, enabling meaningful comparison of their relative growth dynamics. Let $\tilde{\mathcal{A}}(t)$ and $\tilde{\mathcal{R}}(t)$ denote the normalized accuracy and rigidity at SFT step t , respectively.

We define their marginal growth rates as:

$$g_{\mathcal{A}}(t) = \frac{d\tilde{\mathcal{A}}(t)}{dt}, \quad g_{\mathcal{R}}(t) = \frac{d\tilde{\mathcal{R}}(t)}{dt}. \quad (6)$$

The first inflection point t^* is defined as the earliest training step satisfying:

$$t^* = \min \{t \mid g_{\mathcal{R}}(t) > g_{\mathcal{A}}(t)\}. \quad (7)$$

This criterion marks the transition where rigidity grows faster than accuracy, indicating diminishing marginal utility from further supervised fine-tuning. We also highlight the region surrounding the inflection point, which represents the most promising range of SFT CKPTs for subsequent RL tuning.

3.2.3 Type Token Ratio (TTR)

We calculate the diversity in the model trajectories using TTR. TTR is defined as the ratio of the number of unique tokens to the total number of tokens across all generated solutions. We should observe an increase in TTR to indicate higher diversity in trajectories. Let $\mathcal{S} = \{s_1, \dots, s_N\}$ denote a set of N generated solutions, where each solution $s_i = (t_{i,1}, \dots, t_{i,|s_i|})$ is a sequence of tokens. The Type-Token Ratio is computed as:

$$\text{TTR} = \frac{|\bigcup_{i=1}^N \{t_{i,j} \mid j = 1, \dots, |s_i|\}|}{\sum_{i=1}^N |s_i|} \quad (8)$$

4 The Effect of SFT

In domain-specific SFT where each input is paired with a single annotated response, the optimization objective reinforces only one trajectory in the output space. This leads to progressive concentration of probability mass around the supervised target, resulting in reduced entropy and increased divergence from the pre-trained base distribution.

4.1 Accuracy vs Rigidity Tradeoff

As expected, increasing the number of supervised fine-tuning (SFT) steps leads to consistent improvements in domain-specific accuracy. In Figure 1 and 5, we observe a near-linear relationship between model accuracy and the corresponding decline in exploration capacity, as measured by the Rigidity Index. Although early training epochs exhibit

some variability in rigidity, an overall linear increase emerges with respect to the number of SFT steps. Additionally, we identify the first inflection point in each graph, defined as the stage at which the rate of increase in rigidity surpasses the rate of accuracy improvement. In practice, optimization pipelines often prioritize maximizing accuracy, which can inadvertently favor heavily SFT-trained models for subsequent fine-tuning stages. By identifying this inflection point, we analyze how models trained before and after this transition differ in their behavior and adaptability during downstream RL training. The blue asterisks highlight the maximum accuracy achieved after RL when starting from each corresponding SFT CKPT. As shown in the figure, the RL performance gradually levels off and reaches its peak near the inflection point.

For mathematical tasks at 1.5B scale, SFT produces a smooth accuracy trajectory, with late onset of rigidity dominance and sustained RL effectiveness near the transition region. In contrast to math, science tasks at 1.5B scale exhibit early rigidity growth relative to accuracy gains, indicating premature distributional compression that constrains subsequent RL improvements. For mathematical tasks at 7B scale, the model maintains plasticity over a larger SFT regime, enabling sustained accuracy improvements before rigidity growth dominates. Although scaling to 7B substantially improves science accuracy and delays rigidity onset, the specialization pressure remains stronger than in mathematical domains, leading to earlier transition into a rigidity-dominated regime.

The earlier transition observed in science benchmarks may be partly attributed to the smaller training corpus (7k in math versus 4k in science). A reduced dataset accelerates coverage of the empirical distribution, leading to faster entropy exhaustion during fine-tuning. Consequently, subsequent updates primarily sharpen existing output modes rather than acquiring new capabilities, causing rigidity to increase disproportionately relative to accuracy. This effect is especially pronounced in smaller models, where limited capacity further amplifies data scarcity.

4.2 Dataset Entropy Exhaustion and SFT Saturation

We also observe that supervised finetuning saturation occurs at similar training durations across both 1.5B and 7B parameter models especially for Math Domain. While larger models possess greater

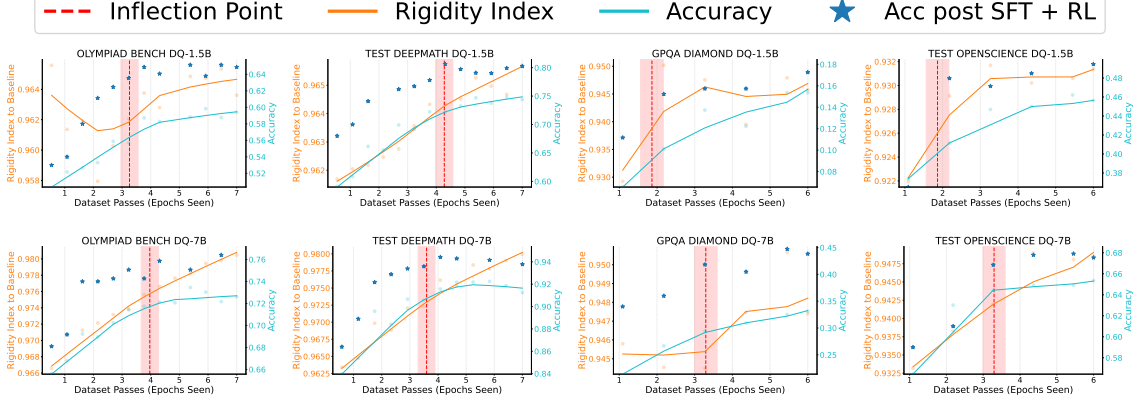


Figure 1: Orange markers and curves represent the Rigidity Index, while light blue markers and curves denote model accuracy for DQ-1.5B (Top row) and DQ-7B (Bottom row) during supervised fine-tuning on Math and Science tasks. The red vertical line indicates the identified inflection point, where rigidity growth begins to exceed accuracy improvement. Blue asterisks denote the maximum accuracy achieved after subsequent RL training when initialized from the corresponding SFT CKPTs.

representational capacity, SFT primarily operates as a distribution sharpening mechanism over pre-trained representations rather than learning new domain knowledge from scratch. Hence, the point of diminishing accuracy returns is governed more strongly by dataset entropy and supervision diversity than by model capacity. Since both models are trained on identical supervision data, they exhaust dataset signal at comparable training steps. Larger models often exhibit faster adaptation to supervised targets due to improved representation reuse and optimization efficiency, which can further explain why saturation does not necessarily occur later for higher-capacity models. Our observations therefore suggest that the transition from beneficial alignment to rigidity-driven over-specialization is primarily data-limited rather than capacity-limited.

Let $\mathcal{D} = \{(x, y)\}$ denote a supervised dataset drawn from an underlying data distribution $\pi_\beta(y|x)$. Supervised fine-tuning (SFT) optimizes the model policy $\pi_\theta(y|x)$ by minimizing the cross-entropy loss:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\mathbb{E}_{x \sim \mathcal{D}, y \sim \pi_\beta(\cdot|x)} [\log \pi_\theta(y|x)] \quad (9)$$

Let us define the conditional entropy of the dataset distribution as:

$$H_{\mathcal{D}}(Y|X) = -\mathbb{E}_{x \sim \mathcal{D}} \left[\sum_y \pi_\beta(y|x) \log \pi_\beta(y|x) \right] \quad (10)$$

During SFT training over steps t , the model distribution π_{θ_t} progressively approaches π_β . We define the *dataset entropy exhaustion ratio* as:

$$\Gamma(t) = 1 - \frac{\mathbb{E}_{x \sim \mathcal{D}} [H(\pi_{\theta_t}(\cdot|x))]}{H_{\mathcal{D}}(Y|X)} \quad (11)$$

where

$$H(\pi_{\theta_t}(\cdot|x)) = -\sum_y \pi_{\theta_t}(y|x) \log \pi_{\theta_t}(y|x) \quad (12)$$

measures the conditional entropy of the model output distribution. As training progresses, $\Gamma(t)$ increases, indicating that the model distribution collapses toward the low-entropy supervision distribution. This condition formalizes that the training distribution is proportional to the entropy in the data distribution.

5 The Effect of RL

5.1 Experimental Setup

To establish a uniform comparison across models trained with varying levels of supervised fine-tuning, we evaluate RL performance using the GRPO method (Shao et al., 2024) with a static trajectory-level reward assigned to each query.

To isolate the effect of RL training, we deliberately avoid incorporating KL divergence penalties or entropy regularization during advantage estimation. This allows us to evaluate the pure contribution of RL optimization without explicit constraints that enforce proximity to the reference policy.

Similar to the SFT stage, RL training exhibits inherent stochasticity. To account for this variability, we conduct multiple independent RL runs for each SFT-initialized model and select trajectories that demonstrate the most significant improvement

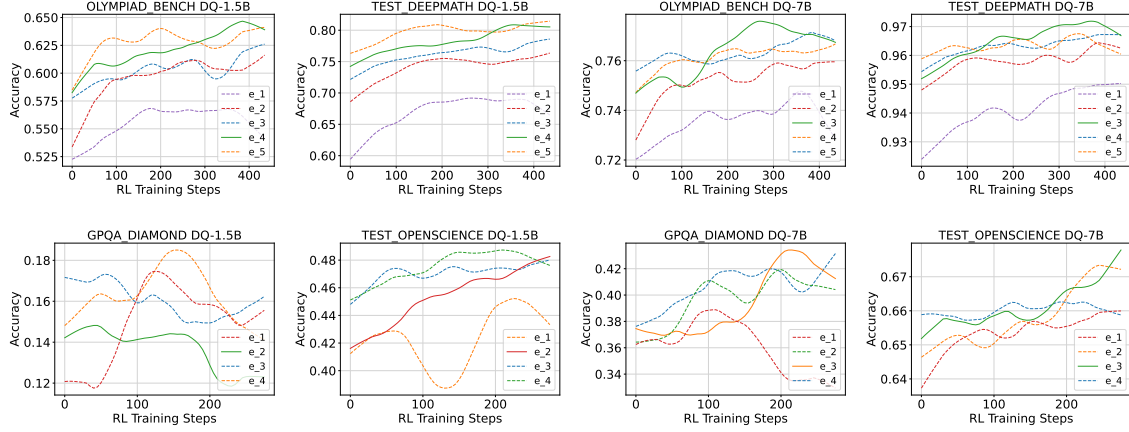


Figure 2: RL performance on different SFT CKPTs of DQ-1.5B and DQ-7B across Math (Top row) and Science (Bottom row) domains. SFT CKPT yielding the highest post-RL performance is highlighted with a solid line.

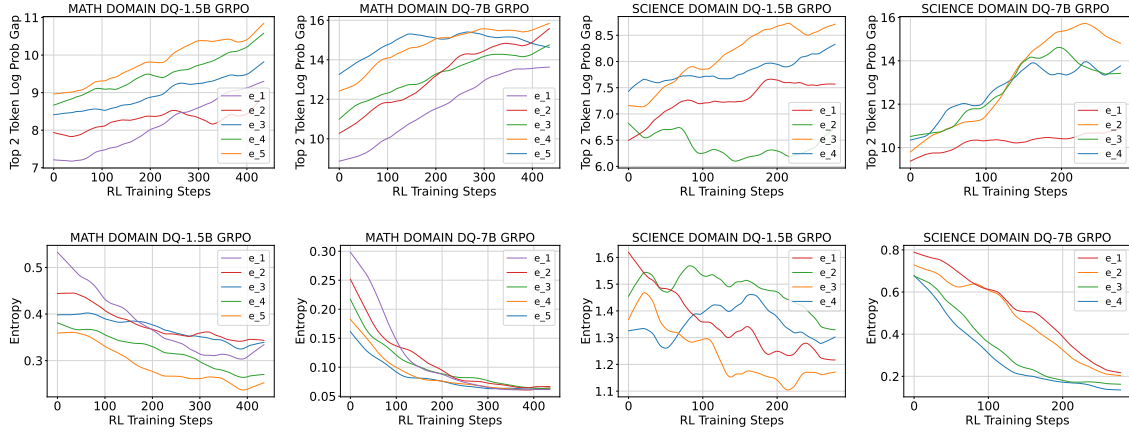


Figure 3: (Top row) Shows the evolution of the average log-probability gap between the top two tokens during RL training when initialized from different SFT CKPTs. As the number of SFT steps increases prior to RL, the probability gap widens more rapidly, indicating increased distribution sharpness and reduced exploratory behavior at each decoding step. (Bottom row) Illustrates the trajectory of output entropy during RL training for models initialized from varying SFT CKPTs. Models starting from heavier SFT exhibit lower initial entropy and experience a steeper entropy decline during RL. X-axis indicates the no. of RL training steps for a given batchsize.

in domain accuracy. For consistency and fairness across models, each RL experiment is conducted under a fixed number of training epochs, and a constant evaluation dataset.

While determining optimal stopping criteria for RL remains an important direction for future work, the objective of this study is to evaluate the relative final performance outcomes when applying a controlled and equivalent amount of RL training to models initialized from different SFT stages.

5.2 Analysing SFT CKPTs post RL

In Figure 2 and 6, we present the effects of initializing RL training from different SFT CKPTs across both Math and Science benchmarks. For both the 1.5B and 7B models, we illustrate the RL training trajectories when initialized from SFT

CKPTs spanning from early epochs (epoch 1) to later stages (epoch 4 or 5). We observe that models initialized from lightly supervised CKPTs, such as epoch 1, fail to achieve strong accuracy improvements during RL and tend to plateau early. In contrast, CKPTs (marked with solid line) closer to the identified inflection point consistently achieve the highest performance gains under RL training. Furthermore, models trained beyond the inflection point exhibit performance that is largely comparable to models initialized at the inflection point. This suggests that additional SFT beyond this stage provides diminishing returns in RL performance while substantially increasing model rigidity. For clarity, RL training from the base model is not plotted because its substantially lower accuracy compresses

RL CKPT	<i>Olympiad Bench</i>		<i>Test DeepMath</i>		<i>GPQA-Diamond</i>		<i>Test Openscience</i>	
	Pre-RL	Post-RL	Pre-RL	Post-RL	Pre-RL	Post-RL	Pre-RL	Post-RL
DQ-1.5B								
Epoch-1	0.108	0.137 (+27%)	0.126	0.134 (+6%)	0.72	0.76 (+5%)	1.05	1.12 (+7%)
Epoch-2	0.126	0.148 (+17%)	0.144	0.155 (+7%)	0.83	0.90 (+9%)	1.19	1.23 (+3%)
Epoch-3	0.130	0.166 (+28%)	0.159	0.173 (+9%)	0.87	0.97 (+13%)	1.26	1.34 (+7%)
Epoch-4	0.141	0.180 (+28%)	0.166	0.191 (+15%)	0.97	1.05 (+7%)	1.34	1.44 (+8%)
Epoch-5	0.144	0.191 (+32%)	0.173	0.209 (+21%)	0.97	1.16 (+19%)	1.37	1.48 (+8%)
DQ-7B								
Epoch-1	0.173	0.180 (+4%)	0.166	0.199 (+20%)	0.79	0.83 (+5%)	1.41	1.77 (+26%)
Epoch-2	0.177	0.188 (+6%)	0.173	0.217 (+25%)	0.87	0.90 (+4%)	1.62	1.91 (+18%)
Epoch-3	0.199	0.202 (+2%)	0.195	0.227 (+17%)	0.89	0.94 (+6%)	1.73	2.13 (+23%)
Epoch-4	0.245	0.249 (+1%)	0.235	0.260 (+11%)	0.90	0.97 (+8%)	1.77	2.24 (+27%)
Epoch-5	0.264	0.267 (+1%)	0.260	0.274 (+6%)	0.94	1.01 (+8%)	1.88	2.38 (+27%)

Table 1: KL divergence with respect to the base model pre & post RL across SFT CKPTs for DQ-1.5B and DQ-7B models.

the scale of the figures; however, these runs were performed and do not alter the reported trends.

In Figure 3 to further analyze these behaviors, we evaluate the impact of RL training using entropy trends and the gap between the top two token log-probabilities. We observe that models trained with heavier SFT already begin RL with a larger gap between their top two token probabilities, indicating increased output confidence and reduced stochasticity. Moreover, this gap widens more rapidly during RL for heavily SFT-trained models, suggesting accelerated distribution sharpening. A similar pattern emerges in entropy analysis: models with greater SFT begin RL with lower entropy and experience a faster entropy decline as RL progresses. Together, these observations reinforce the hypothesis that excessive SFT leads to reduced exploratory capacity, which persists and amplifies during subsequent RL optimization. Collectively, these results indicate that the inflection point serves as an effective indicator for identifying an optimal balance between insufficient and excessive supervised fine-tuning.

5.3 KL Divergence Pre & Post RL

The table 1 reports the KL divergence with respect to the pretrained model for CKPTs obtained before and after RL fine-tuning across multiple benchmarks. Each row corresponds to a model initialized for different epochs, with all CKPTs subsequently trained using an identical RL procedure and budget. We observe that models subjected to more extensive SFT already exhibit higher divergence prior to RL, and this gap widens further after RL training. Consistent with prior observations that RL seeks to optimize behavior without inducing large distributional shifts relative to the base model (Shenfeld et al., 2025), we observe moderate increases in KL divergence after RL across most settings. However,

models that undergo more extensive SFT begin RL from already shifted distributions, and RL training on these CKPTs causes further distributional shift away from the pretrained model. Consequently, the overall deviation grows with the amount of prior supervised fine-tuning. Notably, the most strongly deviated models typically correspond to the latest SFT epochs (shown in red), indicating that deeper supervised fine-tuning stages predispose the model to larger cumulative shifts once RL is applied.

Although the science benchmarks also consist of deterministic rewards, we observe substantially larger KL divergence compared to mathematical tasks. We hypothesize that the divergence arises from differences in the reasoning trajectories rather than the final outputs. Mathematical solutions exhibit highly structured, symbolic, and procedural patterns, which remain relatively stable under RL optimization. In contrast, science reasoning involves more linguistically diverse and concept-driven explanations, allowing RL to alter discourse structure, evidence weighting, and reasoning style. We claim that alignment modifies reasoning policies more aggressively in linguistically flexible domains than in structurally constrained ones.

Overall, both pre and post-RL the KL values increase monotonically with SFT depth, suggesting that excessive supervised fine-tuning pushes the model into progressively narrower regions of parameter space that RL then reinforces further.

5.4 Diversity on Benchmark

To further understand the impact of the RL-SFT CKPTs we measure output diversity using similarity metrics like Jaccard similarity score and TTR on the model outputs. Figure 4 illustrates the trade-off wherein models initialized from higher SFT CKPTs exhibit significantly greater token-level similarity

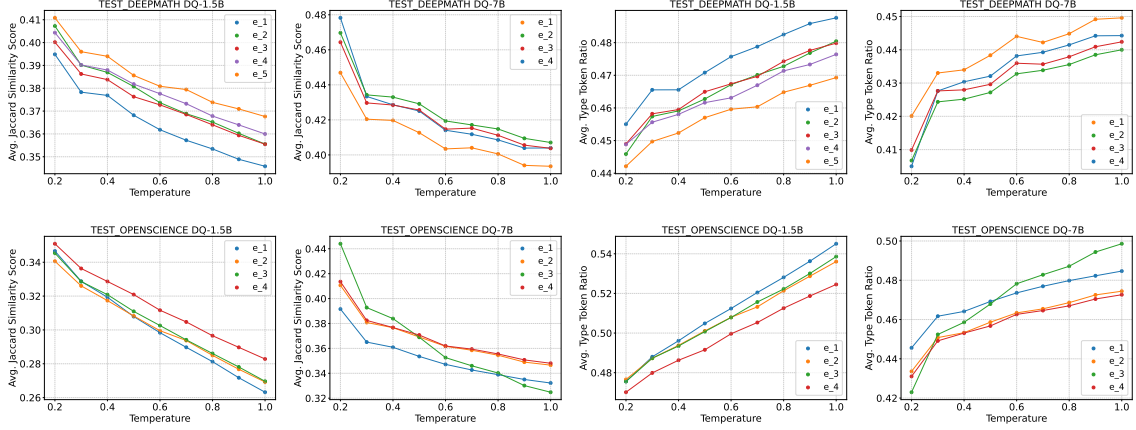


Figure 4: Jaccard Similarity Score and TTR Ratio for DQ-1.5B and DQ-7B SFT + RL trained models.

following a fixed amount of RL training. Specifically, we observe that models starting from heavier SFT consistently achieve higher Jaccard similarity scores across all sampling temperatures. Across all plots in Figure 4 and 7, SFT CKPTs from later epochs (4–5) consistently start with higher Jaccard similarity across all temperatures, whereas earlier CKPTs (1–2) exhibit comparatively lower similarity. As the extent of SFT increases prior to RL, model outputs demonstrate elevated rollout similarity at every temperature, and the rate at which similarity decays with increasing temperature becomes progressively smaller. This suggests that stronger SFT initialization leads to reduced sensitivity to sampling stochasticity during RL.

This hypothesis is further supported by TTR, we observe inverse an trend to the general Jaccard score. In Models initialized from higher SFT CKPTs (Epoch 5) display lower TTR scores after RL training compared to their counter lower SFT CKPTs (Epoch 1), indicating reduced diversity across rollouts. We also see there is much slower increase in TTR especially for the DQ-7B models on both the domains, indicating that the decoding hyperparameters turn much less effective in over SFT finetuned models.

6 Conclusion

In this work, we study the impact of the amount of supervised fine-tuning (SFT) performed prior to reinforcement learning (RL) on model accuracy, generalization, and behavioral diversity. Our findings show that excessive SFT followed by RL leads to increasingly rigid models, characterized by a narrow solution space, reduced output diversity, and degraded generalization relative to the pre-

trained model. Conversely, insufficient SFT results in models that lack the domain-specific knowledge required to benefit effectively from RL, yielding lower overall performance.

These results highlight a critical trade-off between specialization and flexibility: while SFT improves in-domain accuracy, beyond a certain point it amplifies distributional shift and rigidity when coupled with RL. We demonstrate that selecting an appropriate SFT CKPT—where KL divergence from the pretrained model and rigidity remain controlled while accuracy is near-optimal—is essential for effective RL fine-tuning. This balance enables models to retain the generalization and exploratory capacity of the pretrained model while achieving strong domain performance, underscoring the importance of principled CKPT selection in multi-stage training pipelines.

7 Limitations

Due to computational constraints, experiments were conducted with restricted context lengths, domains, and model scales; evaluating the proposed findings on longer contexts, more challenging tasks, and a broader range of model sizes would be necessary to assess generality. Additionally, the analysis focuses on conventional SFT followed by RL, and alternative alignment methods may yield different optimal regimes. The observed rigidity–accuracy trade-off may also depend on dataset characteristics such as size, entropy, and annotation style, as well as on the specific reward function used during RL. Furthermore, the proposed rigidity index captures only certain aspects of output diversity and may not fully represent reasoning flexibility. Finally, the results are derived from a particular pretrained

architecture, and broader validation across models is required to establish robustness. In our future work we will investigate scaling laws for optimal SFT initialization across model size, data entropy, and task complexity.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- Hardy Chen, Haoqin Tu, Fali Wang, Hui Liu, Xianfeng Tang, Xinya Du, Yuyin Zhou, and Cihang Xie. 2025. [Sft or rl? an early investigation into training rl-like reasoning large vision-language models](#). *Preprint*, arXiv:2504.11468.
- Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. 2024. Self-play fine-tuning converts weak language models to strong language models. *arXiv preprint arXiv:2401.01335*.
- Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of machine learning research*, 24(240):1–113.
- Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine, and Yi Ma. 2025. Sft memorizes, rl generalizes: A comparative study of foundation model post-training. *arXiv preprint arXiv:2501.17161*.
- Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.
- Andrew M Dai and Quoc V Le. 2015. Semi-supervised sequence learning. *Advances in neural information processing systems*, 28.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.
- Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don’t stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 8342–8360.
- Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. 2024. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3828–3850.
- Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, et al. 2025. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. *arXiv preprint arXiv:2504.11456*.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.
- Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, DDL Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 10.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.
- Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.
- Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Zhi-Quan Luo, and Ruoyu Sun. 2024. Preserving diversity in supervised fine-tuning of large language models. *arXiv preprint arXiv:2408.16673*.

- NVIDIA Corporation. 2025. Openscience. <https://huggingface.co/datasets/nvidia/OpenScience>. Multi-domain synthetic QA dataset with reasoning traces for training LLMs.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Venkatesh Balavadhani Parthasarathy, Ahtsham Zafar, Aafaq Khan, and Arsalan Shahid. 2024. The ultimate guide to fine-tuning llms from basics to breakthroughs: An exhaustive review of technologies, research, best practices, applied research challenges and opportunities. *arXiv preprint arXiv:2408.13296*.
- Xin Qiu, Yulu Gan, Conor F Hayes, Qiyao Liang, Yinggan Xu, Roberto Dailey, Elliot Meyerson, Babak Hodjat, and Risto Miikkulainen. 2025. Evolution strategies at scale: Llm fine-tuning beyond reinforcement learning. *arXiv preprint arXiv:2509.24372*.
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. [Improving language understanding by generative pre-training](#). *OpenAI Technical Report*.
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. [Language models are unsupervised multitask learners](#).
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.
- David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2024. Gpqa: A graduate-level google-proof q&a benchmark. In *First conference on language modeling*.
- Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.
- Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. 2025. [RL’s razor: Why online reinforcement learning forgets less](#). *Preprint*, arXiv:2509.04259.
- Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021.
- Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7464–7473.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023b. [Llama: Open and efficient foundation language models](#). *Preprint*, arXiv:2302.13971.
- Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.
- Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256.

A Appendix

A.1 SFT & RL Results on MATH-500

We observe a similar relationship between rigidity and accuracy slopes on the MATH-500 benchmark in figure 5. For DQ-1.5B, the inflection point occurs slightly later than for DQ-7B.

In figure 6, we also see the similar pattern as other MATH benchmarks for post RL accuracy on different SFT CKPTs.

A.2 Diversity Ablation on Olympiad Bench and MATH-500

In Fig. 7, we observe similar diversity patterns on MATH-500 and Olympiad Bench for both models. Although minor irregularities exist—for instance, the TTR at epoch 3 slightly exceeding that of epoch 2 on Olympiad Bench—the overall trend remains consistent across checkpoints.

A.3 Variance Across Multiple RL Runs on Test DeepMath

We tested for variance across multiple runs to establish the reliability of the observed training dynamics. We conducted two independent RL runs with different random seeds for the DQ-1.5B model on the Math domain and report the mean and standard deviation of peak accuracy and final entropy across runs on the TEST DEEPMATH dataset. The results in table 2 demonstrate that our findings are consistent across seeds notably, the checkpoint at E5 shows near-zero variance in both accuracy (± 0.01) and entropy (± 0.006), strongly supporting the robustness of the inflection point as a stable initialization target for RL.

B Licenses

Here, we discussed about the licenses of the artifacts we used in our work.

For models we used DeepSeek-R1-Qwen-Distill-1.5B and 7B models (Guo et al., 2025), which are originally licensed under MIT License.

For data we used DeepMath-103k (He et al., 2025) and nvidia/OpenScience (NVIDIA Corporation, 2025) data which are license under MIT License and Creative Commons Attribution 4.0 License respectively.

All of the licenses above enable us to perform research experiments.

C Others

When constructing this paper, we used gpt-4 (Achiam et al., 2023) for several paraphrasing.

SFT CKPT	Post-RL Acc (Run1)	Post-RL Acc (Run2)	Post-RL Acc (Mean)	Post-RL Ent (Run1)	Post-RL Ent (Run2)	Post-RL Ent (Mean)
E3	0.704	0.725	0.714 ± 0.011	0.356	0.345	0.350 ± 0.006
E4	0.748	0.767	0.758 ± 0.010	0.326	0.334	0.330 ± 0.004
E5	0.802	0.798	0.800 ± 0.002	0.267	0.268	0.267 ± 0.000
E6	0.784	0.806	0.795 ± 0.011	0.234	0.242	0.238 ± 0.004

Table 2: Variance across two independent RL runs (different random seeds) for DQ-1.5B on the Math domain, evaluated on TEST DEEPMATH. We report mean and standard deviation across runs for post-RL peak accuracy and final entropy.

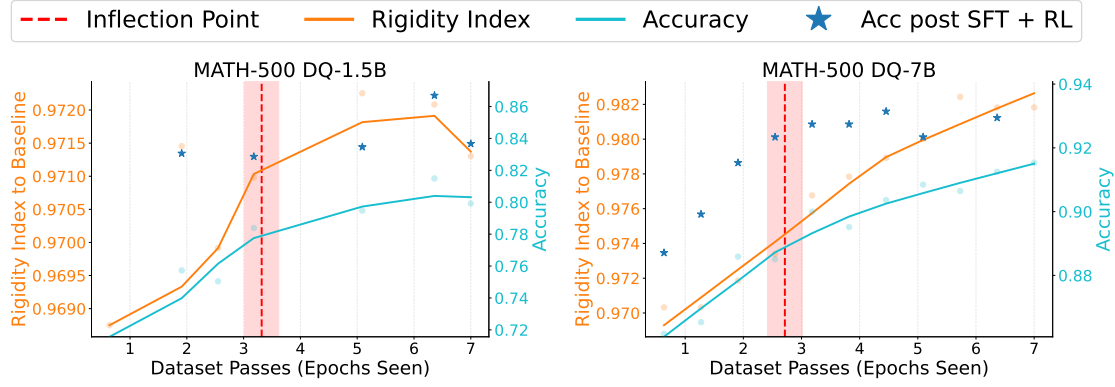


Figure 5: Orange markers and curves represent the Rigidity Index, while light blue markers and curves denote model accuracy for DQ-1.5B (Left) and DQ-7B (Right) during supervised fine-tuning on Math-500 Benchmark. The red vertical line indicates the identified inflection point, where rigidity growth begins to exceed accuracy improvement. Blue asterisks denote the maximum accuracy achieved after subsequent RL training when initialized from the corresponding SFT CKPTs.

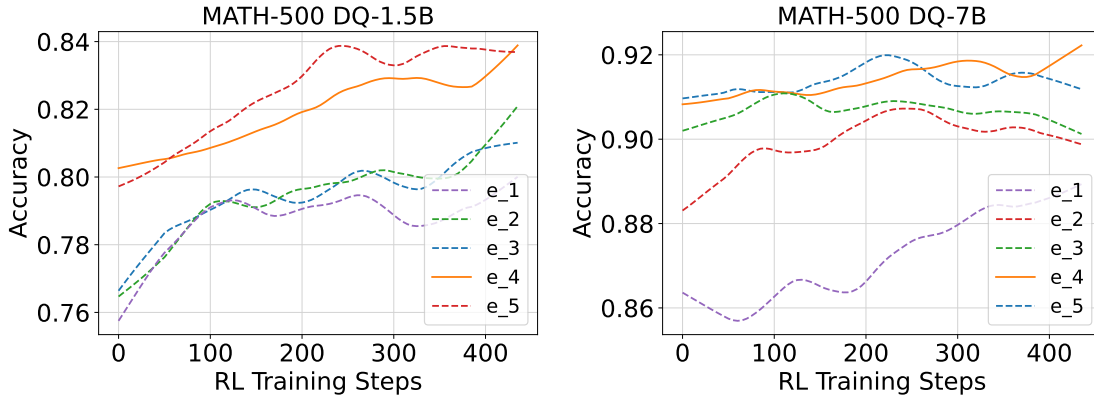


Figure 6: RL performance on different SFT CKPTs of DQ-1.5B (left) and DQ-7B (right) across Math-500 benchmark. SFT CKPT yielding the highest post-RL performance is highlighted with a solid line

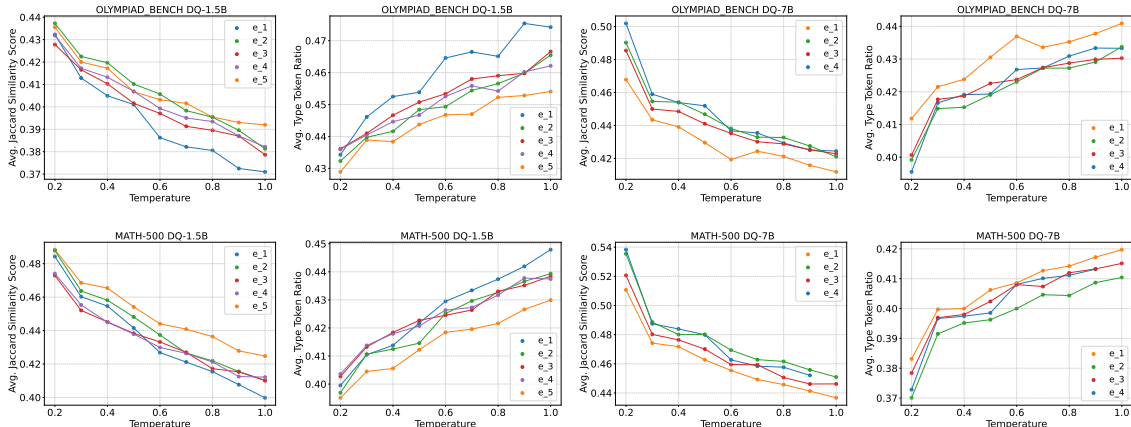


Figure 7: Jaccard Similarity Score and TTR Ratio for DQ-1.5B and DQ-7B SFT + RL trained models.