

A Starting Score Is Not a Ceiling: When Accuracy Before Reinforcement Learning Predicts Accuracy After It, Why Each Proposed Replacement Sees a Different Failure, and the Axis Test That Decides Which Proxy to Trust

Pranay Mahendrakar

SONYTECH · ORCID 0009-0003-7224-029X

pranaymahendrakar@sonytech.in · mahendrakarpranay@gmail.com · +91 6361723454

Abstract. Reasoning models are usually built in two stages, supervised fine-tuning and then reinforcement learning against a verifier, and the first stage is usually judged by the benchmark accuracy of its checkpoint. An August 2026 study reports that the strategy with by far the lowest accuracy before reinforcement learning finished with the highest accuracy after it, and concludes that pre-RL accuracy is not a reliable indicator of post-RL potential. In this analysis the author reviews and synthesises published results to show that the conclusion is right about the instrument and too broad about its failure. Arithmetic on that study's own tables shows that the whole reversal comes from one candidate whose output format had been disrupted: the four format-stable strategies keep the same order before and after reinforcement learning on the nine-benchmark mean, and pass@32, the replacement proxy proposed elsewhere, ranks the disrupted candidate last as well. Across the wider record, starting accuracy ranks candidates correctly when they differ in how much fresh data they saw, and fails when they differ in repetition, homogeneity, behavioural repertoire or output format, axes along which pass@1 and reinforcement-learning readiness come apart. Each proposed replacement, pass@k at large k, held-out validation loss, behaviour and faithfulness probes, early curve fits, sees one of these failure channels and is blind to at least one other, and "post-RL accuracy" is itself indexed by the reinforcement-learning budget and readout. The paper consolidates 22 published results in a table and a figure, states an axis-first decision procedure for choosing a proxy, and specifies five studies that would settle what remains open.

Keywords: large language models; reinforcement learning with verifiable rewards; supervised fine-tuning; post-training; checkpoint selection; proxy metrics; pass@k; evaluation methodology

SONYTECH ORCID: 0009-0003-7224-029X

1. Introduction

The recipe that produced the open reasoning models of 2025 has two stages. A pretrained model is first fine-tuned on long chains of thought, usually written by a stronger teacher, and is then trained by reinforcement learning against an automatic check of its final answer. DeepSeek-R1 put the pattern on record, reporting that reinforcement learning alone could produce strong reasoning from a base model and that a small supervised "cold start" before reinforcement learning further improved performance and fixed the readability and language-mixing problems of the pure-RL variant (DeepSeek-AI, 2025). A systematic study of long chain-of-thought training reached a similar verdict on the supervised stage: not strictly necessary, but a simplifier of training and a gain in efficiency (Yeo et al., 2025). Most of what followed kept the two stages, and many groups that build such models run them in sequence and tune each one separately (Kang et al., 2025).

Running the stages in sequence creates a selection problem. The supervised stage produces many candidate checkpoints - different data mixtures, different numbers of epochs, different points along one training run - and reinforcement learning is expensive enough that only a few of them can be carried forward. The cheapest way to choose is to evaluate each candidate on the benchmarks

the final model will be judged on and to promote the one that scores highest. The working assumption behind that habit is that a checkpoint that is more accurate before reinforcement learning will be more accurate after it. The assumption is rarely stated, because it seems too obvious to state.

A body of 2025 and 2026 work says that it is false, or at least not reliable. Kang et al. (2025) trained hundreds of models up to 12 billion parameters with supervised fine-tuning and then reinforcement learning, and found that the checkpoint with the best score after fine-tuning was often not the one with the best score after reinforcement learning. Training on the same examples for more epochs raised the post-SFT score while lowering what reinforcement learning could add, and selecting the shortest examples did the same. In the most recent and starkest case, Tang et al. (2026) compared five ways of using reasoning-rich text that has no chain of thought written out - worked solutions, textbook derivations - before a shared reinforcement-learning stage. The strategy they called Mixed SFT, one fine-tuning stage on both kinds of data together, started roughly twenty points below every other strategy and finished above all of them. Their methodological conclusion is direct: pre-RL scores "do not reliably predict post-RLVR performance", and strategies for this kind of data should be judged through the full pipeline.

Other results point the other way. A team that scaled its supervised data from tens of thousands to over two million examples reported that stronger fine-tuned models continued to give better results after large-scale reinforcement learning, although the gap between them narrowed (Liu et al., 2025a). A study that decomposed the final ceiling into the supervised starting point and the headroom reinforcement learning adds found that larger supervised datasets raised both the starting point and the headroom, that harder ones raised the starting point while leaving the headroom similar, and that the right moment to switch from fine-tuning to reinforcement learning was when validation loss had flattened, not earlier (Ding et al., 2025). Kang et al. themselves report correlations between post-SFT and post-RL accuracy that are well above zero. A reader could tally these as two camps: one in which starting accuracy is a reasonable guide, one in which it misleads.

In this analysis the author reviews and synthesises published results to show that the tally is the wrong frame. Whether starting accuracy predicts the outcome of reinforcement learning is not a property of the metric. It is a property of the candidate set: of the axis along which the candidates differ. Pass@1 on a benchmark is a reading of what the policy produces now, under the evaluation's decoding settings and answer format. What reinforcement learning can reach from that policy depends on several further quantities: whether the policy produces answers the verifier can parse, how many distinct problems it can solve at least sometimes, what knowledge it holds that its current outputs do not display, which reasoning behaviours it can be pushed to use more often, and how far its parameters can still move. Some ways of producing candidates move pass@1 and these quantities together. Others trade one against another. Starting accuracy ranks candidates correctly along the first kind of axis and fails along the second, and the published counter-examples are, without exception that this review found, of the second kind.

The same frame explains why the proposed replacements do not agree with each other. Pass@k at large k, held-out validation loss, behaviour probes, faithfulness probes and fits to early reinforcement-learning curves each read one of the hidden quantities. Each therefore repairs the failure of starting accuracy along one axis and inherits it along another. The clearest instance is in the anchor study itself: by arithmetic on its own appendix table, pass@32 before reinforcement learning ranks Mixed SFT last among the five strategies, as average accuracy does. A proxy validated on candidates that differ by repetition, as pass@64 was, did not see a candidate that differs by output format.

The argument proceeds as follows. Section 2 fixes terms, including the point that "post-RL accuracy" depends on a reinforcement-learning budget and on how the final number is read off. Section 3 states the claim at issue and its two readings. Section 4 re-reads the anchor study's tables. Sections 5 and 6 assemble the evidence that starting accuracy predicts and the evidence that it

misleads. Section 7 reconciles them by the axis of variation. Section 8 examines each replacement proxy against the same axes, and Section 9 the dependence of the target on budget and readout. Section 10 consolidates the measured values in a table and a figure. Section 11 states a decision procedure as an algorithm, Section 12 names the studies that would settle the open part, and Sections 13 to 16 give the strongest objection to this paper, what is not known, limitations, and a conclusion.

2. Scope and terms

This section fixes the vocabulary. The terms are stipulated, and the rest of the paper uses them in exactly these senses.

Candidate and candidate set. A candidate is a checkpoint that could be handed to reinforcement learning. A candidate set is the collection being compared in one decision: checkpoints along one supervised run, models fine-tuned on different datasets, or models produced by different methods for using the same data. A predictor is judged on a candidate set, not on a single model.

Axis of variation. The axis is what differs between the members of a candidate set: the number of unique examples, the number of epochs over a fixed set, the difficulty or length of the examples, the presence of particular reasoning behaviours in the demonstrations, the mixture of data types and the output format they impose, or an intermediate training objective. One candidate set may vary along several axes at once, and the published studies differ most in which axis they vary.

Starting accuracy. P_{pre} is the candidate's accuracy on the evaluation benchmarks before reinforcement learning, computed under a stated decoding configuration and answer-extraction rule. Most studies report it as the mean over samples of a binary correctness score, written $\text{avg}@n$ when n samples are drawn per problem; with $n = 1$ it is pass@1.

Coverage. Pass@k is the probability that at least one of k independent samples is correct. It is estimated without bias from $n \geq k$ samples by the formula introduced for code generation (Chen et al., 2021) and used for repeated sampling of reasoning models (Brown et al., 2024). Coverage at large k is a reading of how many distinct problems the policy can solve at all, as opposed to how reliably it solves them.

Post-RL accuracy, gain and ceiling. $P_{\text{post}}(x)$ is accuracy after reinforcement learning with compute or step budget x . The gain is $P_{\text{post}}(x) - P_{\text{pre}}$. The ceiling is the limit of P_{post} as the budget grows, which can only be estimated, usually by fitting a saturating curve to the training trajectory (Khatri et al., 2025; Ding et al., 2025). These three quantities can rank the same candidates differently, and much of the confusion in the literature comes from studies that target different ones.

Readout. The readout is how a single post-RL number is taken from a training run: the score at the end of a fixed

stage, the best score over several checkpoints and several runs, or the asymptote of a fitted curve. Kang et al. (2025), for example, repeated reinforcement learning four or more times per candidate, evaluated four or more checkpoints per run, and reported the best. A different readout can produce a different ranking.

Proxy validity. A proxy for the outcome of reinforcement learning is valid on a candidate set, at a stated budget and readout, to the extent that it ranks the candidates as P_{post} ranks them. Rank correlation is the natural statistic, because the decision is which candidate to promote. Kang et al. (2025) also report the coefficient of determination of a linear predictor fitted on half the candidates and tested on the other half, which measures something stronger: whether the proxy predicts the size of the outcome, not only its order.

Scope. The evidence is drawn from mathematical reasoning with verifiable answers, which is where the controlled studies have been run. The vision-language studies are used for contrast. Agentic and non-verifiable settings are excluded because no study located here measured a starting score against a post-RL outcome in them. Reinforcement learning from a base model without supervised fine-tuning ("zero RL") enters only where it bears on what a starting checkpoint must already contain.

3. The claim at issue and its two readings

The anchor for this paper is Tang et al. (2026). Their main question is whether reinforcement-learning objectives that reward a model for predicting the next piece of a text - "next-chunk reasoning", instantiated as next-token reasoning in the manner of Reinforcement Pre-Training (Dong et al., 2025) and next-sentence reasoning in the manner of RLPT (Li et al., 2025b) - beat supervised fine-tuning as a way to use text that contains reasoning but no explicit trace. They compare five strategies from Qwen3-30B-A3B-Base, all followed by the same stage of group relative policy optimisation (GRPO; Shao et al., 2024) on DAPO-Math-17K: Reasoning SFT on long chain-of-thought data only; Sequential SFT, a stage on the trace-free data followed by a stage on long chain-of-thought data; Mixed SFT, one stage on both; and the two next-chunk objectives applied after Reasoning SFT. The long chain-of-thought data are 152K trajectories, about 1.95B tokens, generated by DeepSeek-V3.2 and filtered for correct answers; the trace-free data are 421K short solutions crawled from the Art of Problem Solving site, about 0.53B tokens. Their headline is that Mixed SFT reaches the highest post-RL accuracy at more than sixty times less training compute than the next-chunk objectives. Their methodological finding, the subject here, is that the strategy ranked last before reinforcement learning was ranked first after it.

The finding admits two readings.

The strong reading. Pre-RL accuracy is not a reliable indicator of post-RL potential in general, so candidates should not be compared by it at all, and the only

trustworthy comparison runs the full pipeline. The anchor's discussion is written in this register, and Kang et al. (2025) frame their study around the same general concern: that an SFT-trained model with the best evaluation performance may not be the best candidate for subsequent reinforcement learning.

The weak reading. Pre-RL accuracy fails as a predictor along some axes of variation and holds along others, so the question for a practitioner is which axis separates the candidates in hand, and which proxy is valid along it.

The strong reading has a cost that the weak one does not. If starting accuracy is unreliable in general, then every comparison of supervised recipes that stops before reinforcement learning is suspect, including most of the data-recipe literature (Guha et al., 2025; Ye et al., 2025; Muennighoff et al., 2025), and the only remedy is a reinforcement-learning run per candidate. If the weak reading holds, most of that literature stands, and the remedy is targeted: a check, before reinforcement learning, of whether the candidates differ along an axis known to break the proxy. The rest of this paper argues that the evidence supports the weak reading, and that the strong reading's most striking example is in fact the cleanest case for it.

4. Re-reading the anchor's own numbers

The anchor reports per-benchmark accuracy before and after reinforcement learning for all five strategies on nine benchmarks (its Table 2), and pass@n for the same cells (its Table 7). Six benchmarks are in-domain competition mathematics (AIME 2024, 2025 and 2026, HMMT 2025 and 2026, IMO-Answer), scored as avg@32; three are out-of-domain (HLE and MMLU-Pro at one sample, GPQA-Diamond at four). The author recomputed means from those tables. Where a number below is a mean, a rank or a correlation, it is the author's arithmetic on the published cells; every input cell is the anchor's.

The reversal is one candidate. On the nine-benchmark mean, the anchor's own text gives Mixed SFT a pre-RL accuracy of 27.5 and a post-RL accuracy of 61.1, while the other four strategies start between 45.9 and 48.8 and gain between 3.3 and 10.3 points. The recomputed means agree. Ordering the other four candidates gives Reasoning SFT followed by next-token reasoning (48.8), then next-sentence reasoning (47.6), then Reasoning SFT alone (46.8), then Sequential SFT (45.9) before reinforcement learning; after it, the same four sit at 58.3, 57.9, 54.7 and 49.2, in the same order. Among the four format-stable candidates, then, starting accuracy ranks the outcome perfectly. With Mixed SFT included, the Spearman correlation between pre-RL and post-RL means is zero; without it, it is one. On the six in-domain benchmarks alone the picture is the same except that the two next-chunk variants swap places, separated by 0.4 points before and 0.6 points after; the anchor reports no repeated runs or variance, so a difference of that size carries no information either way.

This does not weaken the anchor's main finding, which is about Mixed SFT and is large. It changes what the finding is evidence for. The data do not show starting accuracy failing across a population of candidates. They show it failing on one candidate, for a reason the anchor itself identifies.

The reason is the output contract. The anchor's analysis attributes Mixed SFT's low starting score to a "transient format artifact". Long chain-of-thought data carry explicit thinking markers and a standard answer format; the short solutions carry neither. Trained on both at once, the model before reinforcement learning sometimes emits several separate thinking segments, or skips the thinking block and answers directly. Its evaluation extracts the answer from a boxed expression or from the span after the closing thinking tag and compares it by exact match. During reinforcement learning, the anchor reports, the rate of format-compliant responses rises quickly and accuracy rises with it. On this account, the starting score of Mixed SFT measures a product of two things - whether the response follows the contract the evaluation and the verifier expect, and whether its content is right - and only the first was depressed.

Two qualifications follow from the anchor's own description. First, the format failure is not only a scoring failure. A response that skips its reasoning block is a different computation, not the same computation with an unparseable answer, and the anchor says the destabilised responses produce lower-quality solutions. So the low starting score is not purely an artefact of the scorer; part of it is a real, temporary loss of the behaviour the evaluation rewards. Second, the claim that the underlying knowledge was intact before reinforcement learning is inferred, not measured. The anchor's evidence is that knowledge was expressed after reinforcement learning: the final ceiling, and a retention probe of 100 problems from the trace-free corpus on which Mixed SFT followed by reinforcement learning scored 68.63 against 59.19 for Sequential SFT followed by reinforcement learning. No pre-RL measurement under an enforced format is reported. The Sequential SFT side of that comparison - a second supervised stage overwriting what the first taught - fits separate reports that supervised fine-tuning forgets prior capabilities more than on-policy reinforcement learning does (Shenfeld et al., 2025; Chen et al., 2025d), though neither of those studies concerns two consecutive supervised stages. The account is plausible and fits every reported number, but the decomposition of the starting score into format and content has not been carried out, and Section 12 proposes how to do it.

Coverage does not rescue the comparison. Kang et al. (2025) propose $\text{pass}@k$ at large k as a better predictor than $\text{pass}@1$, on the reasoning that reinforcement learning compresses the problems a model can sometimes solve into problems it reliably solves. The anchor's Table 7 lets that proposal be checked on a candidate set it was not validated on. On the six in-

domain benchmarks, where every cell is $\text{pass}@32$, the mean pre-RL coverage of Mixed SFT is 62.5, against 78.0 to 83.4 for the other four. $\text{Pass}@32$ ranks Mixed SFT last, as $\text{avg}@32$ does, and orders the other four as the nine-benchmark mean of $\text{avg}@32$ does. On the six in-domain benchmarks its Spearman correlation with the in-domain post-RL ranking is zero, against -0.1 for in-domain $\text{avg}@32$; on the nine-benchmark means, starting accuracy's correlation is zero. Format disruption that makes a model skip its reasoning lowers its best-of-32 as well as its average. A proxy that reads coverage cannot see past an output contract that every sample violates often enough. (One cell of the anchor's Table 7, Mixed SFT after reinforcement learning on AIME 2026, repeats the $\text{avg}@32$ value 70.42, which is not a multiple of one thirtieth; it is probably a transcription slip in the source. It affects no pre-RL value used here.)

What else the anchor holds fixed. Three features of the design bound how far its result travels. The mixing ratio of the two data types in Mixed SFT is a single fixed value; the anchor lists this as a limitation, naming the ratio's effect on the size of the pre-RL drop and on the post-RL ceiling as unexplored. The reinforcement-learning stage is one configuration, and the anchor's training curves show Mixed SFT trailing the other strategies early in that stage before overtaking them; a stage stopped early would have ranked Mixed SFT lower on the three benchmarks whose curves the anchor plots, and Section 9 returns to this. And the evaluation uses a 64K-token budget at temperature 0.8, settings that plausibly favour format-compliant long reasoning. None of these is a flaw. Each is a variable that the general conclusion silently fixes.

5. The case that starting accuracy predicts the outcome

The evidence that starting accuracy is a usable guide is stronger than the counter-example literature suggests, and some of it comes from the same studies.

Scaling supervised data. AceReason-Nemotron 1.1 scaled its supervised data for a 7B model from 18K math and 18K code samples in its first version to 1.2M math and 1.0M code samples in its seventh, and then ran a staged reinforcement-learning recipe from several of the resulting checkpoints (Liu et al., 2025a). Its authors ask directly whether a stronger fine-tuned model consistently leads to better final performance after large-scale reinforcement learning, and answer yes, conditional on effective reinforcement learning with a well-chosen sampling temperature. The gap between candidates shrinks: between their fifth and seventh supervised versions it fell from 6.6 points to 1.6 points on AIME 2024 over the course of reinforcement learning, and on coding a starting gap of 5.5 points fell to 1.6 by the third stage. But the order held, and when the starting gap was large - their seventh version against DeepSeek-R1-Distill-Qwen-7B - a substantial gap remained on AIME 2025 after reinforcement learning. Their predecessor reported that reinforcement learning added 14.6 and 17.2 points

on AIME 2025 to strong distilled 7B and 14B models (Chen et al., 2025c), so a strong starting point did not exhaust what reinforcement learning could add.

Scale and difficulty, at matched training regime. Ding et al. (2025) fine-tuned Qwen2.5-7B and Llama3.2-3B on supervised datasets ranging from 1K to 889K unique trajectories and followed each with reinforcement learning. On Llama3.2-3B their Table 1 gives, for five datasets, the best score from supervised training alone and the best score within a 200-step reinforcement-learning stage started from the checkpoint with minimum validation loss: 24.2 and 25.0 for S1K, 52.4 and 53.9 for Easy102K, 53.6 and 55.3 for Uniform102K, 55.6 and 56.4 for Hard102K, and 67.7 and 70.4 for SFT889K. The order is identical before and after; the author's rank correlation on these five pairs is one. The gains are small, between 0.8 and 2.7 points, and the ranking is carried almost entirely by the supervised stage. On Qwen2.5-7B they report that larger data raise both the supervised score and the headroom reinforcement learning adds, and that harder trajectories at fixed size raise the supervised score (75.4 against 73.5 for easy and 74.6 for uniform) while leaving the headroom similar. Along the axes of data scale and data difficulty, with SFT trained to minimum validation loss, followed by a 200-step RL stage, starting accuracy predicts the outcome. The Llama3.2-3B readout, though, is not a fitted ceiling: the source reports the maximum score reached within 200 reinforcement-learning steps "instead of the theoretical ceiling", and does so "To prioritize practical relevance". By the logic of Section 9, a budget that short leaves the starting ranking nearly in place almost by definition, so the Llama ordering is weak evidence. The case along these axes rests on the Qwen2.5-7B ceiling estimates and on the four-stage runs of AceReason-Nemotron 1.1.

Switching too early. Ding et al. (2025) also report the dependence on when supervised training stops. On their largest dataset, switching to reinforcement learning after 69.8 exaFLOPs of supervised training gave an estimated ceiling of 80.7, while continuing to the plateau of validation loss at 698.4 exaFLOPs gave 84.0. On two out-of-domain benchmarks, starting reinforcement learning from an early checkpoint produced larger gains during reinforcement learning but lower final scores (92.0 against 94.0 on ARC-C, 51.0 against 57.0 on GPQA-Diamond). Along the axis of supervised training time, up to the plateau, more starting accuracy meant more final accuracy, and the smaller gain from the stronger start did not reverse the order.

Intermediate objectives that raise the start. The next-chunk objectives themselves were proposed with the claim that their gains carry into later training. RLP reports that with identical post-training its gains compound (Hatamizadeh et al., 2025); RLPT reports improvements on Qwen3-4B-Base of 6.6 and 5.3 points on AIME 2024 and 2025 and describes itself as a foundation that enhances later reinforcement learning (Li et al., 2025b); Reinforcement Mid-Training reports up

to an 18.76 percent improvement in mathematics after subsequent post-training (Tian et al., 2025); PretrainZero describes its models as foundations for downstream reinforcement learning (Xing et al., 2025). The anchor's result is that these gains did not beat Mixed SFT. Inside the anchor's own table, though, both next-chunk variants raised starting accuracy over Reasoning SFT and both finished above it, which is again starting accuracy predicting correctly among format-stable candidates.

The counter-example study's own correlations. Kang et al. (2025) are the most-cited evidence against starting accuracy, and their numbers show it is a weak predictor, not a useless one. On Llama3-8B-Instruct, varying sample count and epochs, post-SFT accuracy explains 43 percent of the variance in post-RL accuracy. Rank correlations between post-SFT and post-RL accuracy are 0.75 and 0.78 when the training configuration is varied on one dataset, and 0.69 and 0.70 when the dataset is varied with the configuration fixed. These are the baselines their proposed metrics improve on, and they are substantial. In candidate sets built to probe the proxy's failures, starting accuracy still ordered most pairs correctly.

6. The case that starting accuracy misleads

The counter-examples are real, and they share a structure that Section 7 makes explicit. The general shape is familiar from reward modelling, where Gao et al. (2022) measured how a gold score first rises and then falls as a policy is optimised harder against an imperfect proxy of it. Choosing supervised checkpoints by benchmark accuracy optimises a proxy for the post-RL outcome one stage earlier, and the results below are the cases where the optimisation went past the range in which the proxy tracked the target.

Repetition. Kang et al. (2025) found that training on the same examples for up to eight epochs gave better post-SFT accuracy than two epochs, at four times the compute, but visibly worse accuracy after reinforcement learning. For Mistral-NeMo-12B-Instruct, post-SFT accuracy kept improving up to four epochs while post-RL accuracy saturated after two. For Qwen3-4B-base, post-RL accuracy degraded with more supervised training even though post-SFT accuracy was well above the base model's. With a fixed supervised budget, in most of their experiments, training once on all available examples did worse, after supervised training and after reinforcement learning, than training twice on half of them; there the starting ranking held, so that comparison is not a failure of starting accuracy. The variable that moved was held-out loss: as epochs accumulated, loss on validation examples rose and, in their words, eventually flared up, while accuracy continued to climb.

Homogeneity and length. In the same study, fine-tuning only on the shortest examples raised post-SFT accuracy faster than random or longest examples did, and produced a significantly lower post-RL result. Short examples are closer to what the model already generates and easier to fit; the authors suggest they teach less of

what reinforcement learning later needs. Ding et al. (2025) report the scale analogue: S1K reached a supervised score of 74.1 with 2.3 exaFLOPs, a level Uniform102K and SFT889K reached only with 69.3 and 174.6 exaFLOPs, but S1K then saturated there while the larger datasets kept improving, and its average reinforcement-learning headroom was 2.4 points below that of the 889K dataset. At matched starting accuracy, the candidates had different ceilings, so starting accuracy could not discriminate them.

Diversity collapse. Dang et al. (2025) tracked pass@1 and pass@k through supervised training on mathematical data for Gemma-2-2B and Qwen-2.5-0.5B and found that pass@1 kept improving while pass@k at larger k peaked early and then fell; generations converged toward a single trace per problem. Interpolating the weights of a late checkpoint with an early one, in the manner of WiSE-FT (Wortsman et al., 2021), recovered pass@k while matching or improving pass@1, and reinforcement learning from the interpolated weights reached higher accuracy with less data than reinforcement learning from either the early or the late checkpoint. The late checkpoint had the higher pass@1 and was the worse starting point. The authors also give a stylised argument that standard policy-gradient updates do not restore lost diversity, which, if it generalises, means the collapse is not repaired downstream.

Behaviour, not correctness. Gandhi et al. (2025) found that under identical reinforcement learning on the Countdown game Qwen-2.5-3B reached about 60 percent accuracy and Llama-3.2-3B about 30 percent, and traced the difference to four behaviours - verification, backtracking, subgoal setting and backward chaining - that Qwen exhibited before training and Llama did not. Priming Llama with a small amount of fine-tuning on traces containing these behaviours let it match or exceed Qwen's trajectory, and priming with traces that showed the behaviours but reached incorrect answers worked as well as priming with correct ones. A priming set of wrong answers is not chosen to raise starting accuracy, and the study's result is that the behaviours, not the correctness of the solutions, decided the outcome. Li et al. (2025a) report the supervised counterpart: fine-tuning on long chain-of-thought samples with incorrect answers cost only 3.2 points against fully correct samples, while shuffling or deleting reasoning steps did real damage, so the structure of the trace mattered more than its content.

Injected behaviours that lower the start. Cen et al. (2025) augmented supervised data with exploratory and exploitative behaviours before reinforcement learning. On one arithmetic task their method gave relatively lower pre-RL accuracy, but reinforcement learning then raised it much more than it raised the baselines, and it ended with the best final scores both in distribution and out of it. Kang et al. (2025) cite this result as an instance of the failure they study.

Imitation that locks a policy in. In vision-language models, two studies report that long chain-of-thought fine-tuning can impede later reinforcement learning. Chen et al. (2025a) describe "pseudo reasoning paths" copied from expert models, which leave the fine-tuned model stuck in an imitative style that later training struggles to move, and Chen et al. (2025b) found that two-stage, interleaved and progressive combinations of supervised and reinforcement learning, as well as data mixing and model merging, all failed to give additive benefits. These are not studies of starting accuracy as a predictor, but they show that a supervised stage can help on hard questions while hurting on simple ones, and that what each stage adds need not sum.

Pretraining analogues. The same pattern appears one stage earlier. Springer et al. (2025) report that an instruction-tuned OLMo-1B pretrained on 3T tokens was more than 2 percent worse on several benchmarks than its counterpart pretrained on 2.3T, and attribute this to a rise in the sensitivity of parameters to modification as pretraining extends. Liu et al. (2022) constructed, on simplified datasets, models with the same optimal pretraining loss and different downstream performance, which flatness of the solution tracked where loss did not. Akter et al. (2025) report that high-quality reasoning data in pretraining has latent effects that appear only after fine-tuning, and that naively scaling the fine-tuning data can wash the benefit out. And in mid-training for reinforcement learning, Wang et al. (2025a) found that mid-training Llama on two different mathematical web corpora gave similar base-model improvements, one slightly better than the other, but very different reinforcement-learning outcomes: significant gains from one, marginal gains from the other. In each case a reading at the end of one stage under-resolved or mis-ordered what the next stage would produce.

Format alone. Finally, the starting score of a model can be dominated by format before any question of capability arises. Wang et al. (2025b) report that reinforcement learning on a single example raised Qwen2.5-Math-1.5B from 36.0 to 73.6 percent on MATH500, of which 8.6 points were improvement beyond format correction; by subtraction, a format-only reward accounts for about 29 of the 37.6 points. Chu et al. (2025) report that in their card-game and navigation tasks supervised fine-tuning was needed before reinforcement learning because it stabilised the output format. A starting score taken before the output contract is settled is partly a score of the contract.

7. Reconciling the two sides: the axis decides

The two bodies of evidence are not in tension once each result is filed by the axis its candidates vary along. Five quantities, beyond the current probability of a correct answer, bear on what reinforcement learning can reach from a checkpoint. Each is supported by at least one result above, and each can move independently of pass@1.

Channel 1: output contract. Whether responses follow the format the verifier parses and the reasoning structure the evaluation's decoding budget favours. Mixed SFT (Tang et al., 2026) and the one-example format effect (Wang et al., 2025b) are the clearest cases. Reinforcement learning against a verifier appears to repair format quickly, plausibly because every well-formed rollout is rewarded and the fix is the same for every problem.

Channel 2: coverage. How many distinct problems the policy solves at least sometimes. Group-based policy gradients learn nothing from a group of rollouts that are all wrong, which is why DAPO's dynamic sampling drops prompts whose groups are uniformly right or uniformly wrong (Yu et al., 2025), and why Cen et al. (2025) identify rollout accuracy that is informative for reinforcement learning as a precondition for effective training. Qin et al. (2025) report that GRPO mainly improves execution on problems the model can already solve, a "coverage wall"; Yue et al. (2025) and Wu et al. (2025b) report that, under the settings they study, reinforcement learning sharpens what the base distribution contains more than it extends it, a reading that prolonged training contests (Liu et al., 2025b). In a fully controlled synthetic setting, Zhang et al. (2025) found gains in pass@128 from reinforcement learning only when pretraining had left headroom and the training problems sat at the edge of what the model could do. Repetition and diversity collapse lower coverage while raising pass@1 (Dang et al., 2025; Kang et al., 2025).

Channel 3: latent knowledge. What the model knows that its current outputs do not display. The anchor's retention probe and the latent pretraining effects of Akter et al. (2025) are cases. A candidate can hold more knowledge and show less of it.

Channel 4: behavioural repertoire. Which reasoning moves - verification, backtracking, decomposition - the policy produces often enough for reinforcement learning to reinforce them (Gandhi et al., 2025; Cen et al., 2025). Zhao et al. (2025) found, training models from scratch, that reinforcement learning converges on a dominant output pattern already present from pretraining, and Shao et al. (2025) that what random rewards amplify in one model family is a behaviour of that family. Behaviours can be present in traces that reach wrong answers, and absent from traces that reach right ones. Matsutani et al. (2025), comparing the two stages at the level of whole reasoning paths, report complementary effects: reinforcement learning compresses the set of distinct incorrect trajectories, while supervised fine-tuning expands the set of distinct correct ones - so the repertoire a candidate brings is something the supervised stage builds and the second stage selects from.

Channel 5: plasticity. How far, and in which directions, the parameters can still usefully move. Kang et al. (2025) and Ding et al. (2025) read it through validation loss; Springer et al. (2025) through sensitivity to perturbation. Overfitting can raise pass@1 while using up this

capacity, which is why it shows in loss before it shows in accuracy.

To these must be added a sixth quantity that belongs to the second stage rather than the candidate: the reinforcement-learning recipe and budget. AceReason-Nemotron 1.1 conditions its positive answer on choosing the sampling temperature well (Liu et al., 2025a); Khatri et al. (2025) show that recipes differ in the ceiling they reach, not only in speed; Cui et al. (2025) report an empirical law in which performance is bought with policy entropy and saturates when entropy runs out, which makes the starting entropy of a candidate part of its potential.

With these quantities named, the evidence sorts itself. Along an axis that moves pass@1 together with coverage, behaviour and plasticity - more data, whether more prompts or more responses per prompt, with each blend trained to plateau, harder trajectories at the same size, more training up to the loss plateau - starting accuracy is monotone in what reinforcement learning reaches, and the studies that varied those axes found it predictive (Liu et al., 2025a; Ding et al., 2025). Along an axis that trades pass@1 against one of the hidden quantities - repetition against plasticity and coverage, short examples against behaviour and coverage, behaviour injection against current accuracy, data mixing against output contract - starting accuracy fails, and the studies that varied those axes found it failing (Kang et al., 2025; Dang et al., 2025; Cen et al., 2025; Tang et al., 2026). The anchor's own five candidates divide in exactly this way: four that differ only in which objective or which order of stages produced them, ranked correctly; one that differs in output contract, ranked wrongly.

This reconciliation predicts something that neither side states. A mixed candidate set, containing candidates that differ along both kinds of axis, will show starting accuracy as a moderate predictor whose errors are concentrated on the candidates produced by trading axes. Kang et al.'s dataset-level set, which varies both sample count (a moving-together axis) and epochs (a trading axis), gives a rank correlation of 0.75 to 0.78, and their reported failures are the over-trained candidates. That is what the reconciliation expects. Whether every one of their mis-ranked pairs involves a repetition-heavy candidate cannot be checked from the published figures; it is one of the tests in Section 12.

8. Each replacement sees one channel

If starting accuracy fails along trading axes, the natural response is a proxy that reads the hidden quantity directly. Several have been proposed. Each is valid along the axis it was built for, and the evidence already shows where each one goes blind.

Pass@k at large k. Kang et al. (2025) compute pass@64, with up to 256 repetitions per benchmark, and find it a much better predictor than pass@1: rank correlations of 0.95 and 0.92 against 0.75 and 0.78 when

training configuration varies, and 0.94 and 0.98 against 0.69 and 0.70 when the dataset varies, with corresponding gains in the coefficient of determination. The reasoning is sound for the coverage channel: if reinforcement learning compresses coverage into reliability, coverage bounds the outcome. Hu et al. (2023) made the earlier general case that heavy sampling gives a finer-grained reading of task ability than single-sample accuracy, and Schaeffer et al. (2023) that coarse metrics manufacture apparent discontinuities. But $\text{pass}@k$ is still scored under the output contract, so it is blind to channel 1: on the anchor's in-domain benchmarks $\text{pass}@32$ ranks Mixed SFT last, as Section 4 showed. It also reads diversity without reading what the diversity is made of. Rahman et al. (2026), studying reinforcement learning under weak supervision, found that raw output diversity was uninformative about which models would generalise - Llama showed higher diversity than Qwen and generalised worse - and that diversity became informative only when computed over responses whose reasoning actually supported their answers. And $\text{pass}@k$ is expensive at long sequence lengths, which Kang et al. note as a limitation.

Held-out validation loss. Kang et al. (2025) find that loss on held-out supervised examples rises as epochs accumulate, and that this rise tracks the loss of reinforcement-learning potential; along the configuration axis it raised the coefficient of determination from 0.57 to 0.88 on Llama3-8B and from 0.29 to 0.79 on Mistral-NeMo-12B. Ding et al. (2025) find that the minimum validation loss reached by a dataset correlates with the post-training ceiling at Pearson r of -0.92 on Qwen2.5-7B and -0.96 on Llama3.2-3B, and use the loss plateau to time the switch to reinforcement learning. The two studies disagree on scope. Kang et al. state that validation loss cannot compare different datasets, because each loss mixes generalisation error with a distribution gap between datasets. Ding et al. use it to compare datasets, but their datasets are all R1-style distilled trajectories - four of them drawn from two R1-distilled collections filtered to one trajectory style, the fifth the 1K-example $s1$ set - scored on one shared 199-example validation set, so the distribution gap is small by construction. Read together, validation loss is valid along the plasticity channel when candidates share a distribution and a validation set, and of unknown validity otherwise. Whether it would have caught Mixed SFT is not known: loss on held-out long chain-of-thought examples could fall or rise when short solutions are mixed in, and the anchor reports neither. At the pretraining stage, predicting downstream accuracy from loss has a longer record (Chen et al., 2024) and so has predicting it from a few cheap benchmark scores (Ruan et al., 2024), and Liu et al. (2022) are the standing reminder that identical loss can hide different downstream potential.

Behaviour and faithfulness probes. Gandhi et al. (2025) measure the frequency of four reasoning

behaviours before training; Rahman et al. (2026) measure the share of correct responses whose reasoning supports the answer and find it predicts whether a model will keep improving under weak supervision; Cen et al. (2025) derive two conditions - informative rollout accuracy and strong co-influence between training examples - from the per-step effect of the reinforcement-learning objective. These probes read channel 4 and partly channel 2. They were designed to explain differences between model families or between primed and unprimed models, where a behaviour is present or absent. None of the located studies tests them as a ranking among strong candidates that all exhibit the behaviours, which is the situation of a team choosing between supervised checkpoints of one model.

Early reinforcement-learning curves. The most direct proxy is a short reinforcement-learning run. Khatri et al. (2025), in a study of more than 400,000 GPU-hours, fit sigmoidal compute-performance curves to reinforcement-learning runs, show that stable recipes can be extrapolated from early training, and identify which design choices shift the asymptote and which only change efficiency. They also report that methods which look better at small compute can be worse when extrapolated, that a longer generation budget slows early progress while raising the asymptote, and that small batches can look better early and be overtaken. Ding et al. (2025) apply the same sigmoidal form to the second stage of supervised-then-RL training. A curve fit reads every channel, because it watches reinforcement learning act, but it reads them only after the fit has enough points, and Khatri et al. exclude the earliest low-compute regime from their fits for stability. Kang et al. (2025) note that early stopping in reinforcement learning is generally ineffective because the fastest early improver may not finish highest. The anchor's training curves give a concrete instance: Mixed SFT trails early and overtakes later, so a short probe would have ranked it too low on the benchmarks plotted.

The gain as a target. Some studies score a candidate by how much reinforcement learning adds to it. Cen et al. (2025) report their method's benefit as an increase in the gain from reinforcement learning over the pre-RL model, and the anchor emphasises Mixed SFT's 33.7-point rise. The gain is not the quantity a selection decision should maximise. A candidate with a depressed start - from format disruption or any other repairable defect - has a large gain for that reason alone. Ding et al. (2025) report the opposite configuration: on two out-of-domain benchmarks, starting reinforcement learning from a saturated supervised checkpoint gave marginal gains during reinforcement learning and the highest final scores. In the anchor and in Cen et al. the large gain went with the best final result, so no conclusion there depends on the gain. But a procedure that promoted candidates by gain would promote the early, under-trained checkpoint in Ding et al.'s comparison and lose.

The pattern across the five proxies is the same as for starting accuracy. Each was validated on a candidate set

that varied along one or two axes, and each reads one or two of the hidden quantities. None has been tested across a candidate set that varies along all of them.

9. "Post-RL accuracy" is indexed by budget and readout

The target of every proxy, P_{post} , is not a single quantity. It depends on how much reinforcement learning is run and how the final number is read, and the studies differ on both.

The anchor reads accuracy at the end of one fixed stage. Kang et al. (2025) read the best accuracy over four or more evaluations in each of four or more runs, a readout that rewards a candidate whose runs are variable as well as one whose runs are good. Ding et al. (2025) read the asymptote of a fitted curve (A_{post}), a quantity that no run actually reaches, for Qwen2.5-7B, and the maximum score within 200 reinforcement-learning steps for Llama3.2-3B. AceReason-Nemotron 1.1 reads accuracy at the end of each of four stages of increasing length (Liu et al., 2025a). These readouts can disagree. AceReason's gap between two candidates fell from 6.6 to 1.6 points as stages accumulated, and the anchor's Mixed SFT moved from last to first within one stage. Khatri et al. (2025) show that two recipes can cross: the one ahead at small compute behind at large compute. For a candidate set, the same can happen, and a proxy that is valid for the asymptote can be invalid at the budget the team will actually run, and vice versa.

The practical consequence is that a proxy claim should state its budget and readout, and that the validity of starting accuracy is itself budget-indexed. At a very short reinforcement-learning budget, the starting ranking is close to the final ranking almost by definition, because little has changed. At the asymptote, the starting ranking is overridden by whatever the candidates' ceilings are. Starting accuracy is the right predictor for the first regime and the wrong one for the second, and most published comparisons do not say which regime they are in. Kang et al.'s best-of-many readout and Ding et al.'s fitted asymptote for Qwen2.5-7B both lean toward the ceiling; Ding et al.'s 200-step readout for Llama3.2-3B sits in the short-budget regime; the anchor's fixed stage is somewhere between. The measurement noise in reasoning benchmarks adds to the problem: Hochlehnert et al. (2025) report that results on these benchmarks move with seeds, decoding parameters, prompt format and hardware, and that many reported reinforcement-learning gains shrink under a standardised protocol. A ranking reversal of a point or two between candidates, at one seed, is not evidence of a reversal.

Two further dependencies sit inside the readout. The first is the benchmark. Wu et al. (2025a) report that one widely used model family is exposed to contamination of standard mathematics benchmarks, and Shao et al. (2025) that spurious rewards raise that family's scores while failing on others; Liu et al. (2025c) and Zeng et al. (2025) report that base models differ markedly in what reinforcement learning from scratch elicits. A predictor

validated on one model family and one benchmark suite may be reading the family. The second is the verifier. The answer-extraction rule decides what counts as format failure, so a lenient extractor would raise Mixed SFT's starting score and a strict one would lower it, without changing the model. The anchor's extraction rule is stated; its sensitivity is not reported.

10. The evidence in a table and a figure

Table 1 consolidates the results the argument rests on. Each row is one published result; the axis column is the author's classification of what the source varied; the last column states what the result implies for starting accuracy as a predictor, under the source's own budget and readout.

Read down the "Axis varied" column against the last column, the table shows the pattern of Section 7: the rows where starting accuracy holds vary scale, difficulty, training time up to the plateau, or the objective among format-stable candidates; the rows where it fails vary repetition, behaviour, format or pretraining length. Validation loss and pass@64 improve on starting accuracy in the rows where they were tested. Validation loss was tested where candidates vary repetition, scale or difficulty and share one validation set; pass@64 was tested within one distribution and also across differently curated datasets, which Kang et al. describe as each coming "from a different distribution". None of those candidate sets differed in output format, and the derived pass@32 row is the one case located where coverage was measurable on such an axis.

Figure 1 shows the two situations side by side. In panel (a) the long connector belongs to the one candidate whose output contract changed; neither pre-RL reading places it correctly. In panel (b) the connectors are short and parallel: reinforcement learning, capped at 200 steps, added little and preserved the order that supervised training had set. The figure compares studies with different models, benchmarks and readouts, so its two panels are not on a common scale; what it shows is the shape of each comparison, not a common quantity.

11. A decision procedure

The reconciliation gives a procedure for choosing among candidates before committing reinforcement-learning compute. Its central step is to identify the axis before choosing the proxy, rather than to search for one proxy that works everywhere. Algorithm 1 states it.

Three choices in the procedure need defending. The contract check in step 3 comes before every proxy because format differences break both starting accuracy and coverage, and because they are cheap to detect: a parse rate and an accuracy conditional on well-formed responses cost no more than the evaluation already run. Step 4 keeps starting accuracy as the default for moving-together axes because the evidence in Section 5 supports it there, and because the alternatives cost more without being shown to do better on such axes. Step 7 compares

Table 1. Published results on whether accuracy before reinforcement learning predicts accuracy after it, classified by the axis each study varied. Values are as reported by the source named on the row; entries marked (derived) are the author's arithmetic on the source's tables.

Source	Setting	Axis varied	Before RL	After RL	For starting accuracy
Tang et al. (2026)	Qwen3-30B-A3B, 9 benchmarks	Format (data mixing)	Mixed SFT 27.5, others 45.9-48.8	Mixed SFT 61.1, highest	Fails on one candidate
Tang et al. (2026) (derived)	Same, 4 format-stable strategies	Objective, stage order	Order NTR, NSR, RSFT, Seq	Same order	Holds
Tang et al. (2026) (derived)	Same, pass@32, in-domain	Format	Mixed SFT 62.5, others 78.0-83.4	Post-RL avg@32 67.4, highest	Pass@32 fails too
Kang et al. (2025)	Llama3-8B, sample count and epochs	Repetition	Post-SFT pass@1	R-squared 0.43	Weak
Kang et al. (2025)	Llama3-8B, Mistral-12B, configuration	Repetition	Spearman 0.75, 0.78	Pass@64 0.95, 0.92	Proxy better
Kang et al. (2025)	Same models, dataset choice	Length, homogeneity	Spearman 0.69, 0.70	Pass@64 0.94, 0.98	Proxy better
Kang et al. (2025)	Qwen3-4B-base, epochs	Repetition	Well above base	Flat or lower	Fails
Ding et al. (2025) (derived)	Llama3.2-3B, 5 datasets, 200 RL steps	Scale, difficulty	24.2 to 67.7	25.0 to 70.4, same order	Holds, short budget
Ding et al. (2025)	Qwen2.5-7B, switch time	Training time	69.8 vs 698.4 EF	Ceiling 80.7 vs 84.0	Holds to plateau
Ding et al. (2025)	Qwen2.5-7B, S1K	Scale	74.1 at 2.3 EF	Saturates, lower headroom	Cannot separate
Ding et al. (2025)	Qwen2.5-7B, Llama3.2-3B	Scale, difficulty	Minimum SFT val. loss	$r = -0.92, -0.96$	Loss predicts
Liu et al. (2025a)	Qwen2.5-Math-7B, SFT v5 vs v7	Data scale	AIME24 gap 6.6	Gap 1.6, order kept	Holds, shrinks
Dang et al. (2025)	Gemma-2-2B, Qwen-2.5-0.5B	Epochs (repetition)	Pass@1 up, pass@k down	WiSE-FT start better	Fails
Gandhi et al. (2025)	Llama-3.2-3B, Countdown	Behaviour	Primed on wrong answers	Matches correct priming	Correctness not decisive
Cen et al. (2025)	Arithmetic task	Behaviour	Lower than baselines	Best final score	Fails
Wang et al. (2025a)	Llama mid-training corpora	Data quality	Slight difference	Large difference	Under-resolves
Springer et al. (2025)	OLMo-1B, 3T vs 2.3T tokens	Pretraining length	Longer is better base	Over 2% worse tuned	Fails (analogue)
Wang et al. (2025b) (derived)	Qwen2.5-Math-1.5B, MATH500	Format	36.0	73.6, about 29 from format	Reads format
Rahman et al. (2026)	Several families, weak rewards	Faithfulness	Raw diversity	Uninformative	Diversity misleads
Khatri et al. (2025)	8B dense, 17Bx16 MoE	RL recipe	Early leader	Can finish behind	Budget matters
Akter et al. (2025)	Pretraining reasoning data	Data stage	Latent before SFT	Shown after SFT	Fails (analogue)
Chen et al. (2025b)	Vision-language models	Stage combination	SFT helps hard items	No additive gain	Not additive

fitted asymptotes rather than early accuracy because Khatri et al. (2025) and the anchor both show early leaders being overtaken, and it treats the probe as a last resort because its cost is what the procedure exists to avoid.

The procedure is the author's synthesis, and parts of it are untested. Step 3's conditional accuracy has not been validated as a predictor by any located study; step 6's faithfulness-filtered diversity was validated under weak supervision (Rahman et al., 2026) and not as a ranking among strong candidates; step 7's asymptote comparison was validated for recipes (Khatri et al., 2025) and for supervised-then-RL pipelines at fixed data (Ding et al., 2025) but not for candidate sets that vary in output format. Section 12 names the studies that would test each step.

12. Tests that would settle the open part

Five studies would turn the reconciliation from a reading of the evidence into a measured result. Each is specified against the quantities of Section 2.

T1. Decompose the starting score of a format-disrupted candidate. Take the anchor's Mixed SFT checkpoint, or any checkpoint fine-tuned on a mixture of formats, and evaluate it before reinforcement learning in three ways: under the original prompt and extractor; under a lenient extractor that accepts any final numeric answer; and under a prompt or constrained decoding that enforces a single thinking block. Report parse rate, the share of responses with a reasoning block, and accuracy conditional on well-formed responses. If accuracy conditional on well-formed responses for Mixed SFT is close to or above that of the other strategies, the "transient format artifact" account is confirmed and step 3 of Algorithm 1 is justified; if it is still far below, the low starting score reflects more than format, and the account needs revision.

T2. Sweep the mixing ratio. Repeat the anchor's Mixed SFT with several ratios of trace-free to long chain-of-thought data, from long chain-of-thought only to a large share of trace-free data, and record the starting score, the parse rate, and post-RL accuracy at two budgets. The anchor names this sweep as future work. It would show

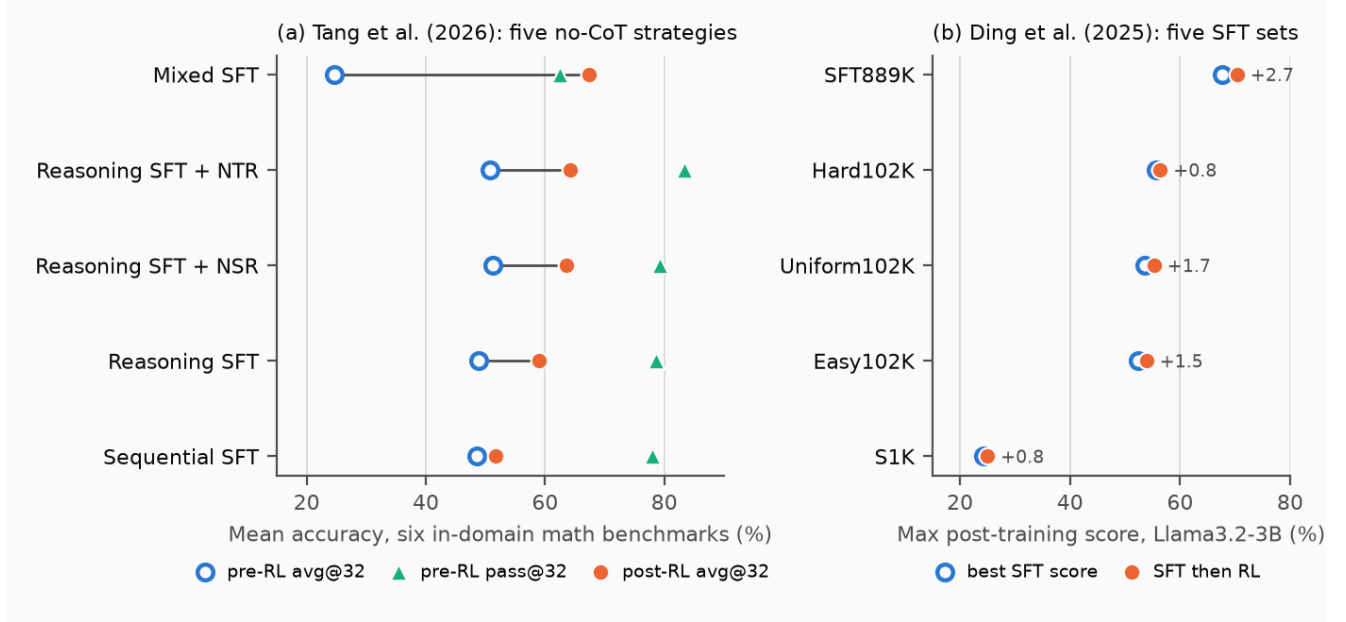


Figure 1. Published accuracies before and after reinforcement learning, re-plotted by the author. (a) Tang et al. (2026): means over the six in-domain math benchmarks of avg@32 before RL (open circles), pass@32 before RL (triangles) and avg@32 after RL (filled circles) for the five strategies, computed from the source's Tables 2 and 7. Mixed SFT is lowest on both pre-RL readings and highest after RL; the other four keep their order apart from a 0.4-point swap of the two next-chunk variants. (b) Ding et al. (2025): for Llama3.2-3B, the best score from supervised training alone and the best score within a 200-step RL stage started from the minimum-validation-loss checkpoint, for five SFT datasets, from the source's Table 1; labels give the difference. Along the scale and difficulty axis the order before RL is the order after it.

Algorithm 1. Axis-first proxy selection for an SFT-then-RL candidate set.

```

Input: candidates C, RL recipe R, budget B, readout Q, eval set E, verifier V.
1 For each c in C, record how it was produced (data, epochs, mixture, objective).
2 Classify the axes along which C varies:
   moving-together: unique-data scale, difficulty at fixed size,
                     training time up to the validation-loss plateau
   trading: repetition, length/homogeneity selection, behaviour priming,
            data-type mixing or any change to output format
3 Contract check: for each c, sample E under the eval decoding settings and
   record parse rate under V, and accuracy on parsed, well-formed responses.
   If parse or well-formed rates differ across C, P_pre confounds format
   with content: do not rank by P_pre or pass@k; go to step 7 for those c.
4 If all variation is moving-together: rank by P_pre (avg@n, n >= 8).
5 If repetition or training time varies within one data distribution:
   add held-out validation loss on one shared validation set; drop
   candidates with both lower P_pre and higher loss.
6 If length, homogeneity or diversity varies: add pass@k at large k,
   estimated without bias from n > k samples; where feasible, compute
   diversity only over responses whose reasoning supports the answer.
7 For unresolved candidates (contract differences, behaviour priming,
   mixed axes): run a short RL probe with recipe R and fit a saturating
   curve; compare fitted asymptotes, not early accuracy.
8 Report the ranking with B and Q stated; repeat RL for the top candidates
   with more than one seed before declaring an order between them.

```

whether the pre-RL drop and the post-RL gain move together - the signature of a repairable format channel - or come apart.

T3. A cross-axis proxy comparison. Build one candidate set from a single base model that varies along every axis of Section 7: unique-data scale, epochs, length selection, behaviour priming including wrong-answer priming, and data-type mixing. Run the same reinforcement-learning recipe on every candidate with at least two seeds. Compute the rank correlation with post-RL accuracy of starting accuracy, pass@64, held-out validation loss on a shared set, a behaviour-frequency

score, faithfulness-filtered diversity, and an early-curve asymptote, both over the whole set and within each axis. The reconciliation predicts that each proxy's correlation is high within the axes its channel covers and low on the others, and that starting accuracy's errors concentrate on candidates produced by trading axes. A single proxy that ranked well across all axes would refute the claim that each sees one channel.

T4. Budget-indexed validity. On the T3 candidate set, read post-RL accuracy at several budgets and under the three readouts in use - end of stage, best of several checkpoints and runs, fitted asymptote - and report each

proxy's rank correlation at each. This would measure how far the validity of starting accuracy decays with budget and would tell a team running a short stage whether the counter-example literature applies to it at all.

T5. Re-analyse the published counter-examples by pair. Kang et al. (2025) trained hundreds of models; the anchor trained five. For every pair of candidates that starting accuracy mis-ranks, classify the pair by the axis that separates them. The reconciliation predicts that nearly all mis-ranked pairs differ along a trading axis. This needs only the per-candidate data behind the published figures, and it is the cheapest test on the list.

13. The strongest case against this paper

The strongest objection is that the reconciliation is unfalsifiable as stated: any failure of starting accuracy can be filed after the fact under a "trading axis", and any success under a "moving-together" one. If so, the classification explains nothing.

The objection has force, and the answer is that the classification is fixed before the outcome, by how the candidates were produced. Repetition, length selection, behaviour priming and data-type mixing are trading axes because each is known, independently of any reinforcement-learning result, to move pass@1 in one direction and coverage, behaviour, plasticity or format in another: repetition raises held-out loss while raising accuracy (Kang et al., 2025), supervised training lowers pass@k while raising pass@1 (Dang et al., 2025), wrong-answer priming adds behaviours without adding correctness (Gandhi et al., 2025), mixing formats breaks the output contract (Tang et al., 2026). A study that varied unique-data scale and found starting accuracy failing, or varied epochs far past the loss plateau and found it holding, would count against the reconciliation. T3 and T5 are designed to look for exactly those cases. The criterion is not applied evenly, however. Supervised training time on fixed data is filed as moving-together for Ding et al. (2025) and AceReason-Nemotron 1.1, whose checkpoints are placed against a validation-loss or accuracy plateau, and as trading for Dang et al. (2025), who report no validation loss; where their checkpoints sit relative to a plateau is not established, so that filing was made after the outcome. A second objection is that the anchor's four format-stable candidates are too few, and too similar, to establish that starting accuracy holds among them. That is correct; four candidates separated by one or two points, with no reported variance, establish only that the anchor's data contain no counter-example once Mixed SFT is set aside. The positive case for moving-together axes rests on the Qwen2.5-7B results of Ding et al. (2025) and on Liu et al. (2025a), not on the anchor.

A third objection is that the paper treats pass@32 in the anchor as a test of a pass@64 proxy estimated with up to 256 repetitions. The difference in k is real, and Kang et al. also used different models and data. But the Mixed SFT gap on pass@32 is 15 to 21 points, and a larger k

would have to close it entirely to change the ranking; nothing in the anchor's tables suggests it would.

14. What is not known

It is not known how much of Mixed SFT's low starting score is format and how much is content; no pre-RL evaluation under an enforced format has been reported. It is not known whether held-out validation loss would have ranked Mixed SFT correctly, because the anchor reports no loss. No located study has compared the proposed proxies on a single candidate set that varies along more than two axes. No located study has measured how the predictive validity of starting accuracy changes with the reinforcement-learning budget. None of the located studies reports seed variance for the post-RL accuracies of its candidates in a way that allows rank reversals of a point or two to be tested. And the controlled comparisons located all score mathematics, code or closely related science benchmarks; none examines agentic or open-ended tasks.

15. Limitations

The first limitation is that the classification of axes in Table 1 is the author's, made from the sources' descriptions of what they varied; a source that varied several things at once has been filed under the one it emphasises. The second is that the recomputed means and rank correlations in Section 4 and Table 1 inherit any transcription errors in the source tables, one of which appears to exist, and that rank correlations over four or five candidates are coarse. The third is that the studies differ in model, data, benchmark suite, verifier and readout, so their numbers are compared by shape, not by magnitude, as the text beside Figure 1 states. The fourth is that several sources are preprints whose results have not been independently replicated, and two of the counter-example literature's most-cited results, those of Kang et al. (2025) and Tang et al. (2026), come from single groups. The fifth is the scope: everything here is about verifiable reasoning, mostly mathematical and in one study also code, and the vision-language and pretraining results enter as analogues, not as evidence about the same pipeline.

16. Conclusion

A score before reinforcement learning measures what a policy does now. What reinforcement learning can reach from that policy depends also on whether its outputs keep the contract the verifier reads, how much it can solve at all, what it knows and does not show, which reasoning moves it can be pushed toward, and how far it can still change. Ways of producing candidates that move these together leave starting accuracy a sound guide, and the studies that varied data scale, difficulty and training time up to the loss plateau found it one. Ways that trade them make it fail, and every published counter-example located here is of that kind - including the anchor's, where the whole reversal comes from one candidate with a disrupted output format, and where the

coverage proxy proposed as a replacement, read at $k = 32$, fails in the same way. The question to ask of a candidate set is therefore not which metric predicts reinforcement learning, but along which axis its members differ, at what budget the outcome will be read, and which proxy has been shown valid along that axis at that budget. Algorithm 1 puts those questions in order, and T1 to T5 would show how far the answers hold.

References

- Akter, S. N., Prabhumoye, S., Nyberg, E., Patwary, M., Shoeybi, M., Choi, Y., et al. (2025). Front-Loading Reasoning: The Synergy between Pretraining and Post-Training Data. arXiv:2510.03264. doi:10.48550/arXiv.2510.03264. [PREPRINT]
- Brown, B., Juravsky, J., Ehrlich, R., Clark, R., Le, Q. V., Re, C., et al. (2024). Large Language Monkeys: Scaling Inference Compute with Repeated Sampling. arXiv:2407.21787. doi:10.48550/arXiv.2407.21787. [PREPRINT]
- Cen, Z., Yao, Y., Han, W., Liu, Z., & Zhao, D. (2025). Behavior Injection: Preparing Language Models for Reinforcement Learning. arXiv:2505.18917. doi:10.48550/arXiv.2505.18917. [PREPRINT]
- Chen, H., Tu, H., Wang, F., Liu, H., Tang, X., Du, X., et al. (2025a). SFT or RL? An Early Investigation into Training R1-Like Reasoning Large Vision-Language Models. arXiv:2504.11468. doi:10.48550/arXiv.2504.11468. [PREPRINT]
- Chen, H., Razin, N., Narasimhan, K., & Chen, D. (2025d). Retaining by Doing: The Role of On-Policy Data in Mitigating Forgetting. arXiv:2510.18874. doi:10.48550/arXiv.2510.18874. [PREPRINT]
- Chen, J., Yu, T., Bai, H., Yao, L., Wu, J., Li, K., et al. (2025b). The Synergy Dilemma of Long-CoT SFT and RL: Investigating Post-Training Techniques for Reasoning VLMs. arXiv:2507.07562. doi:10.48550/arXiv.2507.07562. [PREPRINT]
- Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. d. O., Kaplan, J., et al. (2021). Evaluating Large Language Models Trained on Code. arXiv:2107.03374. doi:10.48550/arXiv.2107.03374. [PREPRINT]
- Chen, Y., Huang, B., Gao, Y., Wang, Z., Yang, J., & Ji, H. (2024). Scaling Laws for Predicting Downstream Performance in LLMs. arXiv:2410.08527. doi:10.48550/arXiv.2410.08527. [PREPRINT]
- Chen, Y., Yang, Z., Liu, Z., Lee, C., Xu, P., Shoeybi, M., et al. (2025c). AceReason-Nemotron: Advancing Math and Code Reasoning through Reinforcement Learning. arXiv:2505.16400. doi:10.48550/arXiv.2505.16400. [PREPRINT]
- Chu, T., Zhai, Y., Yang, J., Tong, S., Xie, S., Schuurmans, D., et al. (2025). SFT Memorizes, RL Generalizes: A Comparative Study of Foundation Model Post-training. arXiv:2501.17161. doi:10.48550/arXiv.2501.17161. [PREPRINT]
- Cui, G., Zhang, Y., Chen, J., Yuan, L., Wang, Z., Zuo, Y., et al. (2025). The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models. arXiv:2505.22617. doi:10.48550/arXiv.2505.22617. [PREPRINT]
- Dang, X., Baek, C., Wen, K., Kolter, Z., & Raghunathan, A. (2025). Weight Ensembling Improves Reasoning in Language Models. arXiv:2504.10478. doi:10.48550/arXiv.2504.10478. [PREPRINT]
- DeepSeek-AI (2025). DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv:2501.12948. doi:10.48550/arXiv.2501.12948. [PREPRINT]
- Ding, B., Chen, Y., Lyv, J., Yuan, J., Zhu, Q., Tian, S., et al. (2025). Rethinking Expert Trajectory Utilization in LLM Post-training for Mathematical Reasoning. arXiv:2512.11470. doi:10.48550/arXiv.2512.11470. [PREPRINT]
- Dong, Q., Dong, L., Tang, Y., Ye, T., Sun, Y., Sui, Z., et al. (2025). Reinforcement Pre-Training. arXiv:2506.08007. doi:10.48550/arXiv.2506.08007. [PREPRINT]
- Gandhi, K., Chakravarthy, A., Singh, A., Lile, N., & Goodman, N. D. (2025). Cognitive Behaviors that Enable Self-Improving Reasoners, or, Four Habits of Highly Effective STaRs. arXiv:2503.01307. doi:10.48550/arXiv.2503.01307. [PREPRINT]
- Gao, L., Schulman, J., & Hilton, J. (2022). Scaling Laws for Reward Model Overoptimization. arXiv:2210.10760. doi:10.48550/arXiv.2210.10760. [PREPRINT]
- Guha, E., Marten, R., Keh, S., Raoof, N., Smyrnis, G., Bansal, H., et al. (2025). OpenThoughts: Data Recipes for Reasoning Models. arXiv:2506.04178. doi:10.48550/arXiv.2506.04178. [PREPRINT]
- Hatamizadeh, A., Akter, S. N., Prabhumoye, S., Kautz, J., Patwary, M., Shoeybi, M., et al. (2025). RLP: Reinforcement as a Pretraining Objective. arXiv:2510.01265. doi:10.48550/arXiv.2510.01265. [PREPRINT]
- Hochlehnert, A., Bhatnagar, H., Udandara, V., Albanie, S., Prabhu, A., & Bethge, M. (2025). A Sober Look at Progress in Language Model Reasoning: Pitfalls and Paths to Reproducibility. arXiv:2504.07086. doi:10.48550/arXiv.2504.07086. [PREPRINT]
- Hu, S., Liu, X., Han, X., Zhang, X., He, C., Zhao, W., et al. (2023). Predicting Emergent Abilities with Infinite Resolution Evaluation. arXiv:2310.03262. doi:10.48550/arXiv.2310.03262. [PREPRINT]
- Kang, F., Kuchnik, M., Padthe, K., Vlastelica, M., Jia, R., Wu, C., et al. (2025). Quagmires in SFT-RL Post-Training: When High SFT Scores Mislead and What to Use Instead. arXiv:2510.01624. doi:10.48550/arXiv.2510.01624. [PREPRINT]
- Khatri, D., Madaan, L., Tiwari, R., Bansal, R., Duvvuri, S. S., Zaheer, M., et al. (2025). The Art of Scaling Reinforcement Learning Compute for LLMs. arXiv:2510.13786. doi:10.48550/arXiv.2510.13786. [PREPRINT]
- Li, D., Cao, S., Griggs, T., Liu, S., Mo, X., Tang, E., et al. (2025a). LLMs Can Easily Learn to Reason from Demonstrations Structure, not content, is what matters!. arXiv:2502.07374. doi:10.48550/arXiv.2502.07374. [PREPRINT]
- Li, S., Li, K., Xu, Z., Huang, G., Yang, E., Li, K., et al. (2025b). Reinforcement Learning on Pre-Training Data. arXiv:2509.19249. doi:10.48550/arXiv.2509.19249. [PREPRINT]
- Liu, H., Xie, S. M., Li, Z., & Ma, T. (2022). Same Pre-training Loss, Better Downstream: Implicit Bias Matters for Language Models. arXiv:2210.14199. doi:10.48550/arXiv.2210.14199. [PREPRINT]
- Liu, M., Diao, S., Lu, X., Hu, J., Dong, X., Choi, Y., et al. (2025b). ProRL: Prolonged Reinforcement Learning Expands Reasoning Boundaries in Large Language Models. arXiv:2505.24864. doi:10.48550/arXiv.2505.24864. [PREPRINT]
- Liu, Z., Chen, C., Li, W., Qi, P., Pang, T., Du, C., et al. (2025c). Understanding R1-Zero-Like Training: A Critical Perspective. arXiv:2503.20783. doi:10.48550/arXiv.2503.20783. [PREPRINT]
- Liu, Z., Yang, Z., Chen, Y., Lee, C., Shoeybi, M., Catanzaro, B., et al. (2025a). AceReason-Nemotron 1.1: Advancing Math and Code Reasoning through SFT and RL Synergy. arXiv:2506.13284. doi:10.48550/arXiv.2506.13284. [PREPRINT]
- Matsutani, K., Takashiro, S., Minegishi, G., Kojima, T., Iwasawa, Y., & Matsuo, Y. (2025). RL Squeezes, SFT Expands: A Comparative Study of Reasoning LLMs. arXiv:2509.21128. doi:10.48550/arXiv.2509.21128. [PREPRINT]
- Muennighoff, N., Yang, Z., Shi, W., Li, X. L., Fei-Fei, L., Hajishirzi, H., et al. (2025). s1: Simple test-time scaling. arXiv:2501.19393. doi:10.48550/arXiv.2501.19393. [PREPRINT]
- Qin, T., Park, C. F., Kwun, M., Walsman, A., Malach, E., Anand, N., et al. (2025). Decomposing Elements of Problem Solving: What "Math" Does RL Teach?. arXiv:2505.22756. doi:10.48550/arXiv.2505.22756. [PREPRINT]
- Rahman, S., Shen, J., Mordvina, A., Palangi, H., Gabriel, S., & Izmailov, P. (2026). When Can LLMs Learn to Reason with Weak Supervision?. arXiv:2604.18574. doi:10.48550/arXiv.2604.18574. [PREPRINT]
- Ruan, Y., Maddison, C. J., & Hashimoto, T. (2024). Observational Scaling Laws and the Predictability of Language Model Performance. arXiv:2405.10938. doi:10.48550/arXiv.2405.10938.

- [PREPRINT]
- Schaeffer, R., Miranda, B., & Koyejo, S. (2023). Are Emergent Abilities of Large Language Models a Mirage?. arXiv:2304.15004. doi:10.48550/arXiv.2304.15004. [PREPRINT]
- Shao, R., Li, S. S., Xin, R., Geng, S., Wang, Y., Oh, S., et al. (2025). Spurious Rewards: Rethinking Training Signals in RLVR. arXiv:2506.10947. doi:10.48550/arXiv.2506.10947. [PREPRINT]
- Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., et al. (2024). DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv:2402.03300. doi:10.48550/arXiv.2402.03300. [PREPRINT]
- Shenfeld, I., Pari, J., & Agrawal, P. (2025). RL's Razor: Why Online Reinforcement Learning Forgets Less. arXiv:2509.04259. doi:10.48550/arXiv.2509.04259. [PREPRINT]
- Springer, J. M., Goyal, S., Wen, K., Kumar, T., Yue, X., Malladi, S., et al. (2025). Overtrained Language Models Are Harder to Fine-Tune. arXiv:2503.19206. doi:10.48550/arXiv.2503.19206. [PREPRINT]
- Tang, Y., Fang, Y., Sun, Y., Liu, J., Wang, Z., Zhao, X., et al. (2026). Is Next-Chunk Reasoning RL Really Better than SFT? Revisiting Training Strategies under no-CoT Data. arXiv:2608.23256. doi:10.48550/arXiv.2608.23256. [PREPRINT]
- Tian, Y., Chen, S., Xu, Z., Wang, Y., Bi, J., Han, P., et al. (2025). Reinforcement Mid-Training. arXiv:2509.24375. doi:10.48550/arXiv.2509.24375. [PREPRINT]
- Wang, Y., Yang, Q., Zeng, Z., Ren, L., Liu, L., Peng, B., et al. (2025b). Reinforcement Learning for Reasoning in Large Language Models with One Training Example. arXiv:2504.20571. doi:10.48550/arXiv.2504.20571. [PREPRINT]
- Wang, Z., Zhou, F., Li, X., & Liu, P. (2025a). OctoThinker: Mid-training Incentivizes Reinforcement Learning Scaling. arXiv:2506.20512. doi:10.48550/arXiv.2506.20512. [PREPRINT]
- Wortsman, M., Ilharco, G., Kim, J. W., Li, M., Kornblith, S., Roelofs, R., et al. (2021). Robust fine-tuning of zero-shot models. arXiv:2109.01903. doi:10.48550/arXiv.2109.01903. [PREPRINT]
- Wu, F., Xuan, W., Lu, X., Liu, M., Dong, Y., Harchaoui, Z., et al. (2025b). The Invisible Leash: Why RLVR May or May Not Escape Its Origin. arXiv:2507.14843. doi:10.48550/arXiv.2507.14843. [PREPRINT]
- Wu, M., Zhang, Z., Dong, Q., Xi, Z., Zhao, J., Jin, S., et al. (2025a). Reasoning or Memorization? Unreliable Results of Reinforcement Learning Due to Data Contamination. arXiv:2507.10532. doi:10.48550/arXiv.2507.10532. [PREPRINT]
- Xing, X., Fan, Z., Lou, J., Li, G., Zhang, J., & Zhang, D. (2025). PretrainZero: Reinforcement Active Pretraining. arXiv:2512.03442. doi:10.48550/arXiv.2512.03442. [PREPRINT]
- Ye, Y., Huang, Z., Xiao, Y., Chern, E., Xia, S., & Liu, P. (2025). LIMO: Less is More for Reasoning. arXiv:2502.03387. doi:10.48550/arXiv.2502.03387. [PREPRINT]
- Yeo, E., Tong, Y., Niu, M., Neubig, G., & Yue, X. (2025). Demystifying Long Chain-of-Thought Reasoning in LLMs. arXiv:2502.03373. doi:10.48550/arXiv.2502.03373. [PREPRINT]
- Yu, Q., Zhang, Z., Zhu, R., Yuan, Y., Zuo, X., Yue, Y., et al. (2025). DAPO: An Open-Source LLM Reinforcement Learning System at Scale. arXiv:2503.14476. doi:10.48550/arXiv.2503.14476. [PREPRINT]
- Yue, Y., Chen, Z., Lu, R., Zhao, A., Wang, Z., Yue, Y., et al. (2025). Does Reinforcement Learning Really Incentivize Reasoning Capacity in LLMs Beyond the Base Model?. arXiv:2504.13837. doi:10.48550/arXiv.2504.13837. [PREPRINT]
- Zeng, W., Huang, Y., Liu, Q., Liu, W., He, K., Ma, Z., et al. (2025). SimpleRL-Zoo: Investigating and Taming Zero Reinforcement Learning for Open Base Models in the Wild. arXiv:2503.18892. doi:10.48550/arXiv.2503.18892. [PREPRINT]
- Zhang, C., Neubig, G., & Yue, X. (2025). On the Interplay of Pre-Training, Mid-Training, and RL on Reasoning Language Models. arXiv:2512.07783. doi:10.48550/arXiv.2512.07783. [PREPRINT]
- Zhao, R., Metereze, A., Kakade, S., Pehlevan, C., Jelassi, S., & Malach, E. (2025). Echo Chamber: RL Post-training Amplifies Behaviors Learned in Pretraining. arXiv:2504.07912. doi:10.48550/arXiv.2504.07912. [PREPRINT]