Warm Up Before You Train: Unlocking General Reasoning in Resource-Constrained Settings

Safal Shrestha **Minwu Kim** **Aadim Nepal** **Anubhav Shrestha** **Keith Ross**[*]

Department of Computer Science, New York University Abu Dhabi

Abstract

Designing effective reasoning-capable LLMs typically requires training using Reinforcement Learning with Verifiable Rewards (RLVR) or distillation with carefully curated Long Chain of Thoughts (CoT), both of which depend heavily on extensive training data. This creates a major challenge when the amount of quality training data is scarce. We propose a sample-efficient, two-stage training strategy to develop reasoning LLMs under limited supervision. In the first stage, we "warm up" the model by distilling Long CoTs from a toy domain, namely, Knights & Knaves (K&K) logic puzzles to acquire general reasoning skills. In the second stage, we apply RLVR to the warmed-up model using a limited set of target-domain examples. Our experiments demonstrate that this two-phase approach offers several benefits: (i) the warmup phase alone facilitates generalized reasoning, leading to performance improvements across a range of tasks, including MATH, HumanEval$^+$, and MMLU-Pro; (ii) When both the base model and the warmed-up model are RLVR trained on the same small dataset (≤ 100 examples), the warmed-up model consistently outperforms the base model; (iii) Warming up before RLVR training allows a model to maintain cross-domain generalizability even after training on a specific domain; (iv) Introducing warmup in the pipeline improves not only accuracy but also overall sample efficiency during RLVR training. The results in this paper highlight the promise of warmup for building robust reasoning LLMs in data-scarce environments.[1][2]

1 Introduction

Reasoning-capable large language models (LLMs) have driven a major shift in artificial intelligence, particularly in tasks requiring multi-step, cognitively complex problem solving (Guo et al., 2025; Team et al., 2025; OpenAI, 2024). These models generate long CoTs, capturing reasoning behaviors such as self-reflection, self-correction, and hypothesis testing (Guo et al., 2025; Team et al., 2025; Qin et al., 2024; Huang et al., 2024; Gao et al., 2024). The CoTs mirror human-like problem-solving and have shown a notable improvement in accuracy on various benchmarks (Qin et al., 2024; Huang et al., 2024; Gao et al., 2024; Xiang et al., 2025).

Most prior approaches to training reasoning-capable models rely on either Reinforcement Learning with Verifiable Rewards (RLVR) or distillation with curated long-CoT demonstrations in specific domains such as math and coding (Guo et al., 2025; Team et al., 2025; Team, 2025; Xu et al., 2025b; Labs, 2025). While effective, these methods require substantial effort to collect and curate domain-specific data, raising the need for more effective and efficient training strategies. Thus, in the LLM reasoning space, there is a need for meta-learning: learning generalizable reasoning strategies that a model can rapidly adapt to multiple downstream tasks (Thrun and Pratt, 1998; Finn et al., 2017; Naik and Mammone, 1992). Following this motivation, we ask:

Can we train models that acquire general reasoning strategies and rapidly adapt them to new domains with minimal supervision?

In this work, we present a new framework for training reasoning models in resource-constrained settings, that is, when only a small amount of domain-specific training data is available. As shown in Figure 1, the framework has two training phases. In the first phase, we warm up the model on a simplified logic game environment—Knights & Knaves (Smullyan, 1986; Xie et al., 2024)—to induce generalizable reasoning behaviors. To accomplish this, we generate long CoTs for Knights & Knaves questions, using QwQ-32B, a reasoning model (Qwen Team, 2025). Then, without fol-

[*] Corresponding author: keithwross@nyu.edu

[1] Our code is available here.

[2] Our model weights and dataset are available here.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 14380–14401
November 4-9, 2025 ©2025 Association for Computational Linguistics

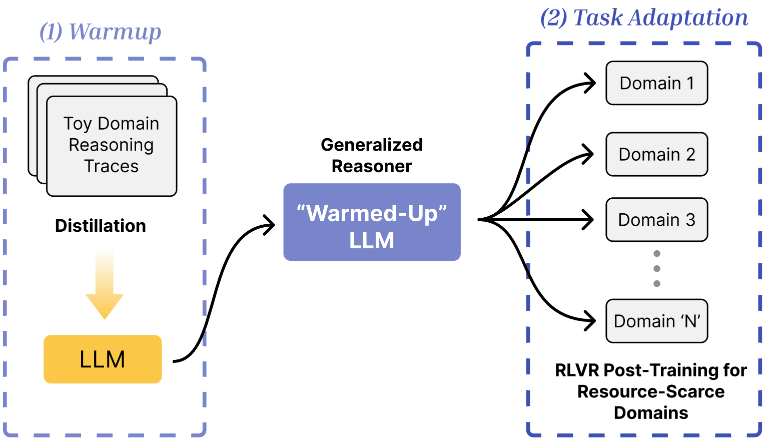


Figure 1: Our two stage methodology involving a (1) *warmup* phase and a (2) *target-task adaptation* phase for training reasoning models in resource-constrained settings

lowing any specialized data filtering scheme, we distill the responses into the base model to create a "warmed-up model". In the second phase, we adapt the warmed-up model to downstream tasks using only a small number of task-specific examples. This setup mimics the meta-learning paradigm: first learn abstract reasoning strategies that can generalize and quickly specialize to new tasks. The main contributions of the paper are:

Warmup alone creates a strong generalizable reasoning model. In Section 2, we show that the initial warmup phase alone yields a strong reasoning model. We warm up various models and consistently observe that it leads to significant improvement across benchmarks from different domains. For example, Qwen2.5-3B model with warmup gains $+10.2\%$ on MATH, $+15.3\%$ on HumanEval$^+$, and $+9.0\%$ on MMLU-Pro. Strikingly, warming up the Qwen2.5-14B model increases its MATH accuracy to 77.4%, closely approaching the 80.2% achieved by full RLVR training with domain-specific data in a previous work (Zeng et al., 2025). This suggests that the warmup phase activates generalizable reasoning behaviors in the model. Additionally, we find that our K&K distillation often outperforms distillation with s1K dataset, a well-curated dataset of Long CoTs from diverse domains.

Warmup turns the model into an effective meta-learner In Section 3.1, we show that the warmed-up model adapts more effectively when trained with RLVR using limited data. When both the base model and the warmed-up model are RLVR-trained on the same small dataset (≤ 100 examples), the warmed-up model consistently outperforms the base model. For example, with the Qwen2.5-3B model, the warmed-up model achieves absolute gains of $+6.7\%$ on MATH and $+5.0\%$ on HumanEval$^+$ compared to the base model RLVR-trained under the same conditions. Remarkably, on MATH, our two-stage approach on just 100 random examples matches the performance of a model trained directly with RLVR on the full training set of 7,500 examples. In addition to stronger final performance, the warmed-up model also converges faster, requiring fewer training steps to reach its peak. These findings demonstrate that the warmup phase turns the model into an effective meta-learner, enabling both efficient and effective RLVR training.

Warming up allows a model to maintain cross-domain generalizability even after subsequent RLVR training on a specific domain. RLVR training can sometimes lead models to adopt concise, domain-specific reasoning patterns that optimize for the target task (Fatemi et al., 2025). While such specialization can yield strong in-domain performance, we show in Section 3.3 that it often comes at a cost of degrading cross-domain reasoning abilities. However, by introducing the warmup phase into the pipeline, we can retain broader generalization capabilities even after task-specific RLVR training. To further assess generalization, we extend our evaluation beyond the commonly studied areas of math and coding to include subsets of the MMLU-Pro benchmark (in particular, physics & history), offering a more comprehensive view of how RLVR affects performance in varied domains (Guo et al., 2025; Team et al., 2025; Wei et al., 2025; Hu et al., 2025; Zeng et al., 2025).

With regards to performance, sample efficiency, and generalization, our two-stage approach consistently outperforms baseline models, underscoring the effectiveness of simple warmup with domain-agnostic reasoning traces. These results highlight the promise of our method as a practical strategy for training robust reasoning models in resource-constrained settings.

2 Warmup phase

The warmup stage is based on the idea that reasoning behaviors—such as self-reflection and self-correction—are broadly applicable across domains. Therefore, this stage is designed to expose the model to such reasoning behaviors such that it can later apply them across various domains. While in past work such a warmup phase has mainly relied on domain-specific data (Luong et al., 2024; Guo et al., 2025), we instead train the LLM on reasoning traces from the simple Knights & Knaves (K&K) logic game (Smullyan, 1986; Xie et al., 2024, 2025). In this section, we ask: *Can warming up a base model with domain-agnostic reasoning patterns lead to substantial improvements on domain-specific tasks?*

2.1 Knights & Knaves logic game

Knights & Knaves Puzzle

Question

A very special island is inhabited only by knights and knaves. Knights always tell the truth, and knaves always lie. You meet four inhabitants: Luke, Liam, Matthew, and Ella. Luke says: "Ella is a knave." Liam says: "Liam is a knight if and only if Luke is a knave." Matthew says: "Liam is a knave if and only if Ella is a knight." Ella says: "Matthew is not a knight." *Who is a knight and who is a knave?*

Answer

Luke is a knave; Liam is a knight; Matthew is a knave; Ella is a knight

Table 1: An example Knights & Knaves logic puzzle.

In K&K, the goal is to determine which characters are knights (who always tell the truth) and which are knaves (who always lie), based on their statements (Example shown in Table 1). We choose K&K for two reasons. First, solving these problems requires extensive reasoning, which highlights generalizable reasoning strategies in the teacher model's responses. Second, K&K relies solely on basic boolean logic and is independent of domain-specific knowledge—such as mathematical theorems or specialized software libraries—allowing the model to focus purely on reasoning behaviors rather than memorization of domain-specific content.

2.2 Experiment Details

We mainly conduct our experiments on the Qwen2.5-3B base model, chosen for its compact size and strong baseline abilities (Yang et al., 2024a). For warmup, we use QwQ-32B (Qwen Team, 2025), a strong reasoning model, to generate long CoTs on K&K questions. One example of such long CoT is shown in Table 3. Since our goal is to collect responses reflecting extensive reasoning behaviors rather than improving accuracy on K&K, we do not apply rejection sampling that filters out wrong responses, similar to the approach taken in some prior work (Gandhi et al., 2025; Li et al., 2025a).

After generating the K&K reasoning traces, we perform Supervised Fine-Tuning (SFT) on the base model with the traces to warmup our model. All details on SFT training with K&K is available in Appendix A.1.

As evaluating LLMs can be noisy (Hochlehnert et al., 2025), we sample multiple times for each question with careful answer extraction mechanisms, as discussed in Appendix A.3. We deal with three diverse datasets to assess the performance of our two-stage methodology across tasks with fundamentally different requirements:

1. **MATH:** A benchmark comprising 12,500 competition-level math problems (7,500 in train and 5,000 in test) designed to assess mathematical reasoning in LLMs (Hendrycks et al., 2021). For evaluation, we use the MATH500 subset of the test set (Lightman et al., 2023).

2. **HumanEval^{+}:** An extended benchmark of 164 hand-crafted programming problems, each with function signatures, docstrings, and unit tests, aimed at evaluating the code generation capabilities of LLMs (Chen et al., 2021;

Liu et al., 2023). In this section, we use all 164 questions for evaluation. When training with RLVR (Section 3), we randomly sample 50 questions and use the remaining examples for evaluation.

3. **MMLU-Pro:** An enhanced version of the Massive Multitask Language Understanding (MMLU) benchmark, featuring challenging questions across 14 categories like physics, law, history, and so on to test real-world text understanding and complex problem solving (Hendrycks et al., 2020; Wang et al., 2024). For MMLU-Pro evaluation, we use the entire dataset except the examples we set aside for RLVR training (Section 3).

2.3 Results for Warmup Only

Table 2 shows the performance of warmup for the three domains for four base models: the Qwen2.5-3B base model as discussed previously; the Qwen2.5-1.5B-Math, a math-specialized smaller model (Yang et al., 2024b); DeepSeek-Math-7B-Base, a relatively weaker but larger model (Shao et al., 2024), and Qwen2.5-14B, an even larger model from the Qwen family (Yang et al., 2024a). For comparison against our non-curated K&K game dataset, we also distilled the base model with s1K, a dataset comprising of methodically collected, high-quality reasoning samples from various domains, including mathematics, physics, geography, logic, law, and so on (Muennighoff et al., 2025). From Table 2, we observe:

1. For the the Qwen2.5-3B base model, although the warmup process does not convey any mathematics or domain-specific knowledge, *warmup leads to marked improvements over the base model across the three domain-specific benchmarks*. This suggests that the warmup phase, using reasoning traces from the K&K domain alone, can effectively activate general reasoning capabilities in the model.

2. In similar vain to the 3B model, we see strong improvements for the other base models as well. In case of the 1.5B model, we see significant improvements on the MATH dataset, possibly because of the math-specialized nature of this model. Similarly, for the 14B Qwen model, we find significant improvements across all three benchmarks. Notably, the model achieves 77.4% on MATH after warmup alone, a result that approaches the 80.2% reported in prior work using full-scale RL training with domain-specific data (Zeng et al., 2025). In case of the DeepSeek family model as well, we see notable improvements across the benchmarks, showcasing the effectiveness of domain-agnostic warmup.

3. As shown in the "s1K" rows of Table 2 for the models, *distilling with CoTs for simple logic games (i.e., warming up) generally matches or even performs better than distilling with domain-knowledge-rich s1K data.* Also, Qwen2.5-14B experiences notable degradation after s1K training. Thus, we explore a different set of hyperparameters for optimal training but still find minimal gains (see Appendix A.4.2).

Some recent studies have showcased that distillation with reasoning-intensive Long-CoT (like s1K data) on smaller models might struggle to learn effectively, showcasing minimal gains or even degradation (Xu et al., 2025a; Li et al., 2025b). Our findings show that distillation in a simplified, domain-agnostic setting with non-curated data like K&K may actually benefit smaller LLMs by avoiding the complexity of domain-specific reasoning and leading to general performance gains. We confirm the validity of the warmup methodology by distilling on models from other model families as well (see Appendix A.4.1). Furthermore, to assess robustness to traces from a different teacher model, we do an ablation study by using DeepSeek-R1 traces for distillation and continue to observe strong performance gains (see Appendix A.5).

2.4 Understanding the Effect of Reasoning Behaviors

The reasoning traces generated by the reasoning teacher model (QwQ) using the K&K games data have general reasoning behaviors, like self-reflection and verification, along with task-specific logic like performing boolean operations and handling states. To test whether the performance gains stem from the reasoning behaviors rather than from the domain-specific logic of Knights & Knaves (K&K), we run a controlled experiment.

In addition to the warmup experiment just described in Section 2.3, we perform another distilla-

Model	**MATH (%)**	**HumanEval$^+$ (%)**	**MMLU-Pro (%)**
Qwen2.5-3B	43.8 ± 0.8	32.5 ± 1.2	29.2 ± 0.3
K&K	**54.0 ± 1.4** ↑	**47.8 ± 3.7** ↑	**38.2 ± 0.4** ↑
s1K	53.9 ± 1.3 ↑	39.5 ± 4.0 ↑	28.1 ± 0.1 ↓
Qwen2.5-Math-1.5B	41.2 ± 1.6	**8.8 ± 2.0**	18.3 ± 0.2
K&K	**65.2 ± 1.0** ↑	5.5 ± 0.8 ↓	**27.5 ± 0.2** ↑
s1K	53.7 ± 1.1 ↑	7.0 ± 0.7 ↓	22.4 ± 0.5 ↑
DeepSeek-Math-7B-Base	20.6 ± 2.3	5.3 ± 1.2	21.1 ± 0.5
K&K	28.7 ± 2.1 ↑	**13.8 ± 4.9** ↑	23.2 ± 0.3 ↑
s1K	**32.4 ± 2.5** ↑	5.9 ± 3.3 ↑	**26.3 ± 0.6** ↑
Qwen2.5-14B	55.6 ± 3.23	64.3 ± 3.0	52.7 ± 0.4
K&K	**77.4 ± 1.9** ↑	**67.3 ± 2.3** ↑	**62.7 ± 0.2** ↑
s1K	51.5 ± 0.7 ↓	24.3 ± 1.9 ↓	34.9 ± 0.7 ↓

Table 2: Performance of the warmed-up model distilled with the K&K game data and s1K data. Best performance highlighted in bold. We share results of warmup on other model families in Appendix A.4.1.

tion experiment. We again choose the Qwen2.5-3B model as our base model but now perform distillation with short CoTs which don't have explicit reasoning behaviors. To collect the short CoTs, we use the non-reasoning Qwen2.5-32B as the teacher model. In this setting, the model quickly overfits to the K&K task, exhibiting significantly reduced generalization abilities. For instance, accuracy on MATH drops to just 11%. In contrast, our "warmed-up" model improves to 54% as shown in Table 2. This shows that the model trained on the K&K games *with* reasoning supervision successfully abstracts transferable reasoning patterns, leading to substantial performance gains across diverse tasks.

3 Target-Domain Adaptation

While simple reasoning behaviors may generalize across domains, as demonstrated in the previous section, many downstream tasks require domain-specific reasoning, e.g., decomposing complex numerical problems in math or generating unit tests in coding (Team, 2025). Such skills need to be polished and are unlikely to emerge from training on a toy domain such as Knights & Knaves alone. Therefore, we introduce a target adaptation phase, where the warmed-up model is RLVR-trained using a few samples ($\leq$ 100) from the resource-constrained target domain.

We first ask *can an initial warmup phase with domain-agnostic reasoning traces give a significant boost to the performance of RLVR training?* If the answer is yes, then in the spirit of meta-learning, we can simply warm up the base model once and then adapt it to many different low-resource domains with RLVR.

3.1 Adaptation for math and coding

We first conduct experiments across the MATH and HumanEval$^+$ datasets discussed in Section 2.2. To create the resource-constrained dataset, we randomly sample 100 questions from the MATH train set and 50 samples (of 164 available) from the HumanEval$^+$. We use the remainder of the dataset for evaluation in case of HumanEval$^+$, and we use the MATH500 dataset for evaluation (Lightman et al., 2023) for MATH. We perform two RLVR runs per dataset, one on the base model and one on our warmed-up (that is, K&K distilled) model. For our experiments, we employ the unbiased GRPO algorithm (Dr. GRPO)(Liu et al., 2025). Due to the high cost of RL training, we perform experiments only on the Qwen2.5-3B base model.

MATH Using the training set of only 100 examples, we perform RLVR fine-tuning on both the base model and the warmed-up model, keeping all hyperparameters constant. As shown in first column of Figure 2, the base model after RLVR training achieves an absolute improvement of +14.0% over the base model after 250 training steps. In comparison, the warmed-up model achieves a significantly larger absolute gain of +20.7% after just 100 steps.

Since we have a large training set (7,500 exam-

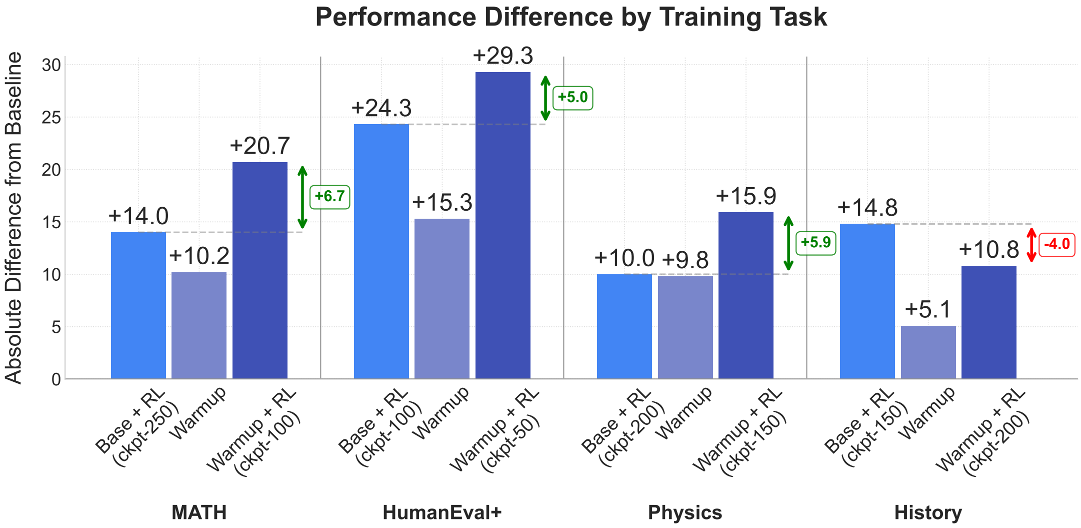


Figure 2: Absolute percentage increase relative to the base model for MATH, HumanEval+, and MMLU-Pro subsets (Physics & History) for various models: RLVR on base model, warmup only, and RLVR on warmed-up model

ples) for MATH, in order to assess performance at scale, we also conduct RLVR training on the base model with the full MATH training set until the model's performance plateaus. Interestingly, the final performance of the base model trained on the full dataset is comparable to that of the warmed-up model trained on just 100 examples: the former reaches a score of 63.2% on MATH500, while the latter achieves 64.5%. While large-scale training with well-curated datasets remains valuable and may ultimately lead to higher ceilings, these results highlight that a warmed-up model can achieve strong performance with fewer examples—demonstrating greater performance and sample efficiency in low-resource settings. Detailed results are shown in Table 9 and 10 in the Appendix.

HumanEval+ For this experiment, we use the RLVR training set of 50 examples. As shown in Figure 2, the base model achieves an absolute improvement of +24.3% after 100 RLVR training steps. In comparison, the warmed-up model alone has good performance, showing an absolute gain of +15.3% without any domain-specific fine-tuning, and reaching an absolute improvement of +29.3% after just 50 RLVR training steps. These results highlight the effectiveness of our two-stage approach in the coding domain as well. Detailed results are shown in Table 11 and 12 in the Appendix.

3.2 Adaptation for MMLU-Pro

MATH and HumanEval+ primarily consist of problems that demand intensive reasoning (Hendrycks et al., 2021; Liu et al., 2023). To assess broader applicability, we expand our study to the MMLU-Pro benchmark. MMLU-Pro presents a broader mix, combining both challenging reasoning tasks and factual recall questions. For instance, MMLU-Pro includes questions from the physics domain, which require a mix of analytical problem-solving skills and familiarity with certain domain-specific concepts. It also includes the history domain, which has questions that rely more heavily on factual world knowledge. We perform RLVR runs on these two specific subsets (physics and history) to assess how the model performs in diverse reasoning and knowledge demands.

Physics From the 1,299 available examples in the physics subset, we randomly sample 100 instances for training and use the remaining for evaluation. As shown in Figure 2, RLVR training improves the base model's absolute performance by +10.0%, reaching peak performance after 200 steps. In comparison, the warmed-up model already exhibits strong performance out of the box with an absolute +9.8% gain, and further improves to +15.9% after just 150 training steps, demonstrating both faster convergence and higher final performance. We find in this case where we have a mix of both reasoning-intensive and knowledge-intensive questions, RLVR training still leads to significant gains. Detailed results are shown in Table 13 and 14 in the Appendix.

History The history subset of MMLU-Pro contains 381 examples, from which we select 100 at random for training and reserve the rest for evaluation. For this domain, RLVR training applied directly to the base model achieves a notable absolute improvement of $+14.8\%$, outperforming the warmed-up model, which reaches a lower gain of $+10.8\%$ after RLVR training. Past work has shown that for knowledge-intensive tasks, even direct prompting without step by step reasoning outperforms Chain of Thought prompting (Wei et al., 2022; Wang et al., 2024). Thus, for knowledge-intensive tasks, having a short CoT might be the optimal policy. As shown in Table 17, the base model after RLVR training learns to produce very short responses on history questions, hence giving better performance. In comparison, since the warmed-up model has been primed for reasoning through the warmup phase, it struggles to perform on par with the base model on a non-reasoning task like history. However, as we discuss in Section 3.3, this domain-specific improvement with RLVR directly applied to the base model may come at the cost of broader generalization. Detailed results for the history runs are shown in Table 15 and 16 in the Appendix.

While previous work has found that SFT using short CoT can limit the benefits of subsequent RL (Zeng et al., 2025), in this section, we showed that RL training a model warmed up with long CoT (reasoning traces), although from a simplified domain (K&K), can significantly improve generalization and sample efficiency. This is especially beneficial for training in resource-constrained settings with limited samples. Details on all hyperparameters related to our experiments, training curves, and results along with standard deviations are available in Appendix A.6.

3.3 Cross-Domain Generalization

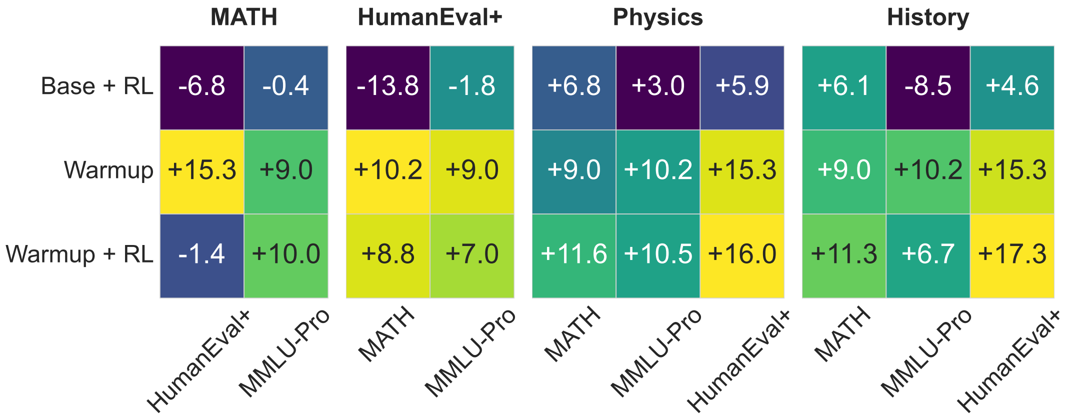
While recent research has shown that large scale RLVR-trained models can generalize to out-of-domain tasks (Zeng et al., 2025; Hu et al., 2025; Xie et al., 2024), we study generalization across multiple distinctly different domains when the sample size is low. We test for generalization across domains with fundamentally different structures, such as math, code generation, and language understanding as discussed in Section 2.2.

In this section, we compare generalizability across the two RLVR runs discussed in Section 3; one with the base model and the other with the warmed-up model. For a model RLVR-trained on one domain (such as physics), we test generalization by evaluating on other domains (such as mathematics and coding). We ask the question, *how does the generalization performance of the two runs compare? Does warming up the base model before RLVR training help to maintain or even improve generalization to other domains?*

Loss in Generalizability on Base Model As shown in Figure 3, we observe that although RLVR with domain-specific questions typically increases the performance of the model when asked to solve problems in the same domain, it tends to lose its initial base-model reasoning capability when asked to solve problems in different domains. This effect is especially pronounced in case of training on reasoning-intensive domains like MATH and HumanEval^{+}. For instance, RLVR training the base model on HumanEval^{+} leads to a -13.8% drop on MATH, while RLVR training on the History subset results in a -8.5% drop on MMLU-Pro. This degradation suggests that RLVR training may push the model to internalize narrow, domain-specific heuristics, thereby weakening its broader reasoning abilities. This interpretation is further supported by our observation (Appendix A.6) that RLVR fine-tuning often leads to more concise completions—consistent with prior findings (Fatemi et al., 2025; Kim et al., 2025)—potentially reflecting a contraction in reasoning depth.

Improved Generalizability with Warmup Figure 3 also shows the generalization results when we begin the pipeline with the warmed-up model. While the forgetting effect is occasionally observed in the warmed-up model (e.g., a -1.4% decrease on HumanEval^{+} relative to the base model after RL training on MATH), *beginning with the warmed-up model rather than with the base model consistently yields stronger cross-domain generalization.*

Figure 4 shows the change in average completion length relative to the base model after RLVR training on different datasets and evaluating on all others. Although both models have short completion lengths after training(A.6), the warmed-up model has a strictly higher completion length relative to the base model's original completions. The consistently longer completions may help preserve the model's general reasoning abilities across multiple



	MATH		HumanEval+		Physics			History		
	HumanEval+	MMLU-Pro	MATH	MMLU-Pro	MATH	MMLU-Pro	HumanEval+	MATH	MMLU-Pro	HumanEval+
Base + RL	-6.8	-0.4	-13.8	-1.8	+6.8	+3.0	+5.9	+6.1	-8.5	+4.6
Warmup	+15.3	+9.0	+10.2	+9.0	+9.0	+10.2	+15.3	+9.0	+10.2	+15.3
Warmup + RL	-1.4	+10.0	+8.8	+7.0	+11.6	+10.5	+16.0	+11.3	+6.7	+17.3

Figure 3: Generalization results for base+RLVR, warmup-only, and warmup+RLVR models. Top labels indicate RLVR training domain; bottom labels indicate evaluation domain. Warmup uses no domain-specific data.

domains.

Interestingly, we find that training on a domain like Physics, which blends both multi-step reasoning and factual recall, better maintains cross-domain performance for both the base and warmed-up models. This observation suggests that task diversity during RLVR training may play a critical role in preserving generalization.

4 Discussion & Related Work

4.1 Meta-Learning & Generalization

Previous studies have adopted two main techniques for developing reasoning LLMs: distillation with well-curated long CoTs, and large scale RLVR training. Both techniques demand large amounts of high-quality data (Guo et al., 2025; Hu et al., 2025; Muennighoff et al., 2025). These methods typically focus on task-specific optimization and do not directly address how to equip models with the ability to adapt rapidly to new domains under low-resource scenarios. Our work, inspired by the meta-learning paradigm, aims to fill this gap (Thrun and Pratt, 1998; Naik and Mammone, 1992; Finn et al., 2017).

While some studies have shown impressive improvements in math domains with RLVR training on tiny curated datasets (Wang et al., 2025; Fatemi et al., 2025), our approach differs in both scope and methodology. We prioritize sample efficiency and cross-domain generalization across diverse domains, with fundamentally different requirements (math, coding, and language understanding), without relying on selective filtering during either the warmup or downstream adaptation phases.

4.2 SFT Warmup for Reasoning

Previous work has shown that reasoning behaviors such as self-verification, backtracking, and self-reflection play a critical role in effective reasoning for large language models (LLMs) (Gandhi et al., 2025; OpenAI, 2024; Guo et al., 2025). Although a base model might exhibit some reasoning behaviors already, performing distillation with long CoTs from tasks requiring extensive reasoning, like Olympiad level math problems or complex coding tasks, can enable these models to become effective reasoners (Muennighoff et al., 2025; Zhao et al., 2025; Liu et al., 2025; Team, 2025; Labs, 2025). However, rather than relying on meticulously curated domain-specific data, we explore the effectiveness of distilling reasoning patterns in the absence of such meticulously curated, domain-specific Long CoTs. We find that warmup with simple domain-agnostic reasoning may also serve as a strong activation mechanism that helps the model to learn to piece together reasoning behaviors for application in multiple downstream domains.

Some previous studies have focused on learning generalizable behaviors in the form of formal logic and evaluating their transferability across various domains (Morishita et al., 2023, 2024). Our motivation is similar, but we instead analyze the transfer of general reasoning skills, such as self-verification and self-reflection, that have been adopted in recent reasoning LLMs (Guo et al., 2025; OpenAI, 2024). Moreover, we not only study the generalizability of these reasoning skills but also investigate how they improve sample efficiency and downstream performance after RlVR training.

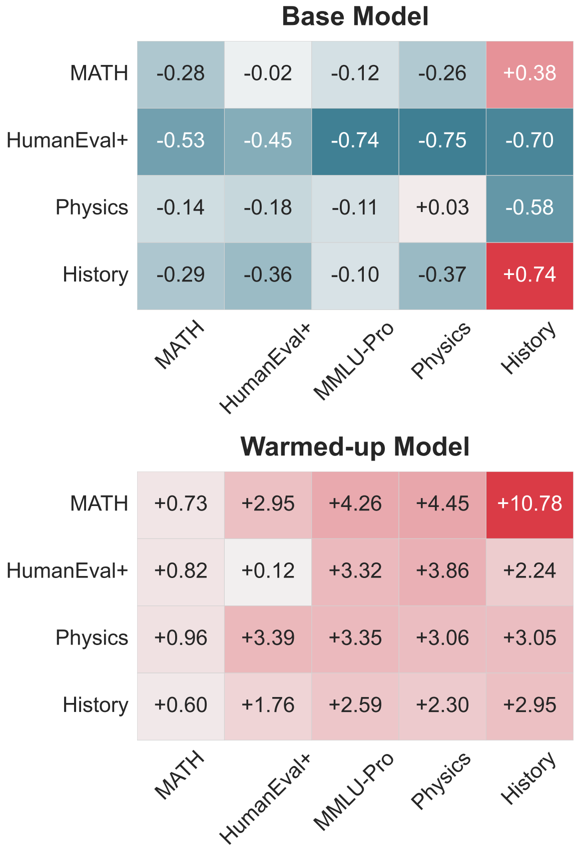


Figure 4: Relative change in completion length (vs. base model) after training on each dataset (y-axis) and evaluating on others (x-axis). Top: base model; Bottom: warmed-up model.

5 Conclusion

We present a two-stage training framework that combines a lightweight warmup phase with reinforcement learning using verifiable rewards (RLVR) to develop reasoning-capable LLMs under limited supervision. By distilling general reasoning from a toy domain and fine-tuning on a small set of target examples, our approach consistently outperforms direct RLVR on base models across math, code, and language understanding tasks. Warmup enhances sample efficiency, boosts performance, and preserves cross-domain generalization—highlighting its promise as a practical strategy for training robust LLMs in low-resource settings.

Limitations

Our findings demonstrate that reasoning distilled from a simple, abstract domain like Knights & Knaves (K&K) can generalize effectively to tasks in math, code generation, and general language understanding—underscoring the promise of domain-agnostic reasoning as a transferable prior. However, it remains an open question how well such reasoning transfers to more complex tasks involving richer dynamics, interaction, or domain-specific constraints, like multi-agent environments.

While we selected K&K for its simplicity and interpretability, we do not claim that it is the only source of transferable reasoning. Other synthetic environments may offer equally or more effective reasoning priors tailored to downstream needs.

We encourage future work to explore the design of such environments as playgrounds for models to learn generalizable reasoning behaviors and apply them to specific domains. Additionally, limited by our compute, we were able to perform distillation only till 14B models and RLVR train on a 3B model; evaluating this methodology on larger models could offer further insights into the scalability and generalization potential of warmup-based reasoning.

Acknowledgements

This work is submitted in part by the NYU Abu Dhabi Center for Artificial Intelligence and Robotics, funded by Tamkeen under the Research Institute Award CG010. Some experiments were carried out on the High Performance Computing resources at New York University Abu Dhabi.

References

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and 1 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Mehdi Fatemi, Banafsheh Rafiee, Mingjie Tang, and Kartik Talamadupula. 2025. Concise reasoning via reinforcement learning. *arXiv preprint arXiv:2504.05185*.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR.

Kanishk Gandhi, Ayush Chakravarthy, Anikait Singh, Nathan Lile, and Noah D Goodman. 2025. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective stars. *arXiv preprint arXiv:2503.01307*.

Peizhong Gao, Ao Xie, Shaoguang Mao, Wenshan Wu, Yan Xia, Haipeng Mi, and Furu Wei. 2024. Meta reasoning for large language models. *arXiv preprint arXiv:2406.11698*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

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*.

Andreas Hochlehnert, Hardik Bhatnagar, Vishaal Udandarao, Samuel Albanie, Ameya Prabhu, and Matthias Bethge. 2025. A sober look at progress in language model reasoning: Pitfalls and paths to reproducibility. *arXiv preprint arXiv:2504.07086*.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. 2025. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. *arXiv preprint arXiv:2503.24290*.

Zhen Huang, Haoyang Zou, Xuefeng Li, Yixiu Liu, Yuxiang Zheng, Ethan Chern, Shijie Xia, Yiwei Qin, Weizhe Yuan, and Pengfei Liu. 2024. O1 replication journey–part 2: Surpassing o1-preview through simple distillation, big progress or bitter lesson? *arXiv preprint arXiv:2411.16489*.

Minwu Kim, Anubhav Shrestha, Safal Shrestha, Aadim Nepal, and Keith Ross. 2025. Reinforcement learning vs. distillation: Understanding accuracy and capability in llm reasoning. *arXiv preprint arXiv:2505.14216*.

Bespoke Labs. 2025. Bespoke-stratos: The unreasonable effectiveness of reasoning distillation. Accessed: 2025-01-22.

Dacheng Li, Shiyi Cao, Tyler Griggs, Shu Liu, Xiangxi Mo, Eric Tang, Sumanth Hegde, Kourosh Hakhamaneshi, Shishir G Patil, Matei Zaharia, and 1 others. 2025a. Llms can easily learn to reason from demonstrations structure, not content, is what matters! *arXiv preprint arXiv:2502.07374*.

Yuetai Li, Xiang Yue, Zhangchen Xu, Fengqing Jiang, Luyao Niu, Bill Yuchen Lin, Bhaskar Ramasubramanian, and Radha Poovendran. 2025b. Small models struggle to learn from strong reasoners. *arXiv preprint arXiv:2502.12143*.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2023. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in Neural Information Processing Systems*, 36:21558–21572.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*.

Trung Quoc Luong, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. 2024. Reft: Reasoning with reinforced fine-tuning. *arXiv preprint arXiv:2401.08967*, 3.

Terufumi Morishita, Gaku Morio, Atsuki Yamaguchi, and Yasuhiro Sogawa. 2023. Learning deductive reasoning from synthetic corpus based on formal logic. In *International Conference on Machine Learning*, pages 25254–25274. PMLR.

Terufumi Morishita, Gaku Morio, Atsuki Yamaguchi, and Yasuhiro Sogawa. 2024. Enhancing reasoning capabilities of llms via principled synthetic logic corpus. *Advances in Neural Information Processing Systems*, 37:73572–73604.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. 2025. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*.

Devang K Naik and Richard J Mammone. 1992. Meta-neural networks that learn by learning. In *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*, volume 1, pages 437–442. IEEE.

OpenAI. 2024. Openai o1 hub. Accessed: 2025-05-11.

Yiwei Qin, Xuefeng Li, Haoyang Zou, Yixiu Liu, Shijie Xia, Zhen Huang, Yixin Ye, Weizhe Yuan, Hector Liu, Yuanzhi Li, and 1 others. 2024. O1 replication journey: A strategic progress report–part 1. *arXiv preprint arXiv:2410.18982*.

Qwen Team. 2025. QwQ-32B: Embracing the Power of Reinforcement Learning. Accessed: 2025-05-11.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory optimizations toward training trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16. IEEE.

Rulin Shao, Shuyue Stella Li, Rui Xin, Scott Geng, Yiping Wang, Sewoong Oh, Simon Shaolei Du, Nathan Lambert, Sewon Min, Ranjay Krishna, and 1 others. 2025. Spurious rewards: Rethinking training signals in rlvr. *arXiv preprint arXiv:2506.10947*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Raymond Smullyan. 1986. *What is the name of this book?* Touchstone Books Guildford, UK.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, and 1 others. 2025. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*.

NovaSky Team. 2025. Sky-t1: Train your own o1 preview model within $450. https://novasky-ai.github.io/posts/sky-t1. Accessed: 2025-01-09.

Sebastian Thrun and Lorien Pratt. 1998. Learning to learn: Introduction and overview. In *Learning to learn*, pages 3–17. Springer.

Yiping Wang, Qing Yang, Zhiyuan Zeng, Liliang Ren, Lucas Liu, Baolin Peng, Hao Cheng, Xuehai He, Kuan Wang, Jianfeng Gao, and 1 others. 2025. Reinforcement learning for reasoning in large language models with one training example. *arXiv preprint arXiv:2504.20571*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, and 1 others. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Yuxiang Wei, Olivier Duchenne, Jade Copet, Quentin Carbonneaux, Lingming Zhang, Daniel Fried, Gabriel Synnaeve, Rishabh Singh, and Sida I Wang. 2025. Swe-rl: Advancing llm reasoning via reinforcement learning on open software evolution. *arXiv preprint arXiv:2502.18449*.

Violet Xiang, Charlie Snell, Kanishk Gandhi, Alon Albalak, Anikait Singh, Chase Blagden, Duy Phung, Rafael Rafailov, Nathan Lile, Dakota Mahan, and 1 others. 2025. Towards system 2 reasoning in llms: Learning how to think with meta chain-of-though. *arXiv preprint arXiv:2501.04682*.

Chulin Xie, Yangsibo Huang, Chiyuan Zhang, Da Yu, Xinyun Chen, Bill Yuchen Lin, Bo Li, Badih Ghazi, and Ravi Kumar. 2024. On memorization of large language models in logical reasoning. *arXiv preprint arXiv:2410.23123*.

Tian Xie, Zitian Gao, Qingnan Ren, Haoming Luo, Yuqian Hong, Bryan Dai, Joey Zhou, Kai Qiu, Zhirong Wu, and Chong Luo. 2025. Logic-rl: Unleashing llm reasoning with rule-based reinforcement learning. *arXiv preprint arXiv:2502.14768*.

Haoran Xu, Baolin Peng, Hany Awadalla, Dongdong Chen, Yen-Chun Chen, Mei Gao, Young Jin Kim, Yunsheng Li, Liliang Ren, Yelong Shen, and 1 others. 2025a. Phi-4-mini-reasoning: Exploring the limits of small reasoning language models in math. *arXiv preprint arXiv:2504.21233*.

Haotian Xu, Xing Wu, Weinong Wang, Zhongzhi Li, Da Zheng, Boyuan Chen, Yi Hu, Shijia Kang, Jiaming Ji, Yingying Zhang, and 1 others. 2025b. Redstar: Does scaling long-cot data unlock better slow-reasoning systems? *arXiv preprint arXiv:2501.11284*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024a. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, and 1 others. 2024b. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*.

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. 2025. Simplerl-zoo: Investigating and taming zero reinforcement learning for open base models in the wild. *arXiv preprint arXiv:2503.18892*.

Rosie Zhao, Alexandru Meterez, Sham Kakade, Cengiz Pehlevan, Samy Jelassi, and Eran Malach. 2025. Echo chamber: Rl post-training amplifies behaviors learned in pretraining. *arXiv preprint arXiv:2504.07912*.

A Appendix

A.1 The Knights & Knaves (K&K) Game

We use Knights & Knaves questions from a previous work (Xie et al., 2024). The questions deal with 3 to 7 characters. Here, the number of characters determines the complexity of the problem. The dataset has a total of $5,000$ examples. We retain 90% of the dataset $(4,500)$ for training and keep the rest for evaluation (500). After distillation on the Qwen 3B model, performance on K&K improves from $19.4\% \pm 1.6$ to $81\% \pm 1.5$ on the test set.

In Table 1, we show an example of the K&K game. In Table 3, we show an example of a reasoning trace, generated by the reasoning model QwQ-32B, on one of such puzzles.

Prompt

A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. The reasoning process and answer are enclosed within <think> </think> and <answer> </answer> tags, respectively. i.e., <think> reasoning process here </think> <answer> answer here </answer>. User: Please solve this logical reasoning problem. After thinking, when you finally reach a conclusion, clearly state the identity of each character within <answer> </answer> tags i.e., <answer> (1) Zoey is a knight (2) ... </answer>. Here is my question,
A very special island is inhabited only by knights and knaves. Knights always tell the truth, and knaves always lie. You meet 3 inhabitants: Riley, Emily, and Lucas. In Riley's words: "Lucas is not a knave". Emily said that Lucas is a knight and Riley is a knight. In a statement by Lucas: "Emily is a knight or Emily is a knave". So who is a knight and who is a knave?
Assistant: <think>

Reasoning from QwQ-32B

Okay, so I need to figure out who among Riley, Emily, and Lucas are knights or knaves. Let me start by recalling that knights always tell the truth and knaves always lie.
First, let's note down each person's statements:
- Riley says: "Lucas is not a knave." - Emily says: "Lucas is a knight and Riley is a knight." - Lucas says: "Emily is a knight or Emily is a knave."

Hmm, Lucas's statement is a bit tricky. Let's break it down. The statement "Emily is a knight or Emily is a knave" is actually a tautology because every inhabitant is either a knight or a knave. So regardless of whether Emily is a knight or knave, that statement is always true. Wait, but if Lucas is a knave, he must lie. But his statement is a true statement, so if he's a knave, he can't tell the truth. That means Lucas cannot be a knave, because his statement is always true. Therefore, Lucas must be a knight? Because if he were a knave, he would have to lie, but his statement is inherently true, so he can't lie about it. So Lucas must be a knight.
Wait, let me think again. Let's analyze Lucas's statement more carefully. The statement "Emily is a knight or Emily is a knave" is a logical OR. Since every inhabitant is either a knight or a knave, this OR is always true. So the statement is a tautology. Therefore, regardless of Emily's actual status, the statement is true. So if Lucas is a knight, he tells the truth, so that's fine. If he were a knave, he would have to lie. But since his statement is true, a knave cannot say it. Therefore, Lucas must be a knight. That's solid.
So Lucas is a knight. Therefore, Riley's statement that "Lucas is not a knave" is true. Since Riley's statement is true, Riley must be a knight as well. Wait, but let's check Emily's statement. Emily says that Lucas is a knight and Riley is a knight. Since we just concluded both are knights, then Emily's statement is true. Therefore Emily must also be a knight. But that can't be right because then all three would be knights, but let me check again.
Wait, hold on. Let me go step by step.
First conclusion: Lucas is a knight (from his own statement being a tautology). Therefore, Riley's statement "Lucas is not a knave" is true, so Riley is a knight. Then Emily's statement is "Lucas is a knight and Riley is a knight." Since both parts are true (Lucas and Riley are knights), Emily's statement is true, so Emily must be a knight. That would mean all three are knights. But is there any contradiction here?
Wait, but let me see. If all three are knights, then all their statements are true, which they are. So that seems okay. But maybe I missed something? Let me try another approach.
Alternative approach: Suppose Lucas is a knave. Then his statement "Emily is a knight or a knave" would be a lie. But since the statement is a tautology (always true), lying about it would require the statement to be false. But since it's always true, Lucas can't be a knave. Therefore, Lucas must be a knight. So that part holds.

Therefore, Lucas is a knight. Then Riley's statement is true, so Riley is a knight. Then Emily's statement is "Lucas is a knight and Riley is a knight." Since both are true, Emily's statement is true, so Emily is a knight. So all three are knights. That seems possible, but maybe the problem expects some of them to be knaves. Let me check again for errors.

Wait, perhaps I made an error in analyzing Emily's statement. Let me parse Emily's statement again. She says "Lucas is a knight and Riley is a knight." If both are true, then Emily is a knight. If either is false, then her statement is false, so she would be a knave.

But according to our previous conclusion, both are true, so she must be a knight. So that's consistent. Therefore all three are knights. But the problem says "inhabited only by knights and knaves," so having all knights is allowed. Hmm, but maybe I missed something else?

Alternatively, perhaps the problem has a trick. Let me consider if there's another possibility. Suppose Lucas is a knight (as before). Then Riley is a knight. Emily's statement is true, so she's a knight. So all three are knights. That works.

Alternatively, is there a scenario where someone is a knave?

Suppose, for contradiction, that Emily is a knave. Then her statement "Lucas is a knight and Riley is a knight" would be a lie. The negation of "A and B" is "not A or not B". So either Lucas is a knave, or Riley is a knave, or both. But we already established that Lucas must be a knight (since his own statement is a tautology). Therefore, if Emily is a knave, then Riley must be a knave. But Riley's statement "Lucas is not a knave" would then be a lie, meaning Lucas is a knave. But that contradicts our earlier conclusion that Lucas must be a knight. Therefore, Emily can't be a knave. So Emily must be a knight.

Therefore, all three are knights. Wait, but that seems a bit too straightforward. Let me check again.

Alternatively, perhaps the problem is designed so that one is a knave. Let me see:

Wait, Lucas's statement: "Emily is a knight or a knave" — since every inhabitant is one or the other, that's always true, so Lucas must be a knight. So that part is solid. Then Riley's statement is true, so Riley is a knight. Emily's statement is true, so she's a knight. So all three are knights. So the answer would be all three are knights.

Hmm, but maybe the problem is expecting a different answer? Let me see the original problem again:

Problem states: Riley says Lucas is not a knave. Emily says "Lucas is a knight and Riley is a knight." Lucas says "Emily is a knight or Emily is a knave."

Wait, the problem might have a trick in Emily's statement. Let me parse Emily's exact words again: "Lucas is a knight and Riley is a knight." So if Emily is a knave, then her statement must be false, so either Lucas is not a knight (so knave) or Riley is not a knight (so knave). But since we know Lucas is a knight, that would require Riley to be a knave. But Riley's statement is "Lucas is not a knave," which would be true (since Lucas is a knight), so Riley would be a knight, contradicting the assumption that Riley is a knave. Therefore Emily can't be a knave. Therefore, all three must be knights.

Alternatively, is there any other possibility? Let me try assuming that Emily is a knave again. If Emily is a knave, then her statement is false. Since the statement is "Lucas is a knight AND Riley is a knight," the negation is "Lucas is a knave OR Riley is a knave." But since Lucas must be a knight (from his own statement), then the only way for Emily's statement to be false is if Riley is a knave. But then Riley's statement "Lucas is not a knave" would be true, so Riley would be a knight, which contradicts the assumption that Riley is a knave. Therefore Emily can't be a knave.

Therefore the only possible solution is that all three are knights.

Wait, but the problem says "a very special island inhabited only by knights and knaves." There's no rule that there must be at least one of each, so all three being knights is acceptable.

Hmm, so the conclusion is that all three are knights. But maybe I made a mistake in the analysis. Let me try another angle.

Suppose Lucas is a knight (as established). Therefore, his statement is true, so the rest follows. Then Riley's statement is true, so Riley is a knight. Emily's statement is true, so Emily is a knight. So yes, that's consistent.

Alternatively, let's suppose that Lucas is a knave. Then his statement "Emily is a knight or a knave" is a lie. But since the statement is a tautology (always true), lying about it is impossible. So Lucas cannot be a knave. Therefore, Lucas must be a knight.

Thus, the only possible conclusion is all three are knights.

But maybe the problem is designed so that someone is a knave? Let me think again.

Wait, perhaps there's a mistake in considering that "Emily is a knight or a knave" is a tautology. Wait, actually, in logic, "A or not A" is a tautology, but here Emily is either a knight or a knave, so "Emily is a knight or Emily is a knave" is indeed a tautology. Therefore, Lucas's statement is always true, so he must be a knight.

Therefore, I think the correct answer is that all three are knights. So Riley: knight, Emily: knight, Lucas: knight.

Wait, but let me check once more. Let me tabulate possible scenarios:

Case 1: All knights. Then all statements are true. That works.

Case 2: Suppose Lucas is a knight, Riley a knight, Emily a knave. Then Emily's statement would be false, but since both parts are true (Lucas and Riley are knights), her statement is true, so she can't be a knave. So this case invalid.

Case 3: Suppose Emily is a knave, then her statement is false, so either Lucas is knave or Riley is knave. But Lucas must be a knight, so Riley must be knave. But then Riley's statement "Lucas is not a knave" is true, making Riley a knight, contradiction. So Emily can't be a knave.

Case 4: Suppose Riley is a knave. Then his statement "Lucas is not a knave" is false, so Lucas is a knave. But Lucas's statement would then be a lie, but his statement is a tautology (true), so Lucas can't be a knave. Contradiction. So Riley must be a knight.

Thus, all three must be knights. Therefore the answer is all three are knights. </think>
<answer> (1) Riley is a knight (2) Emily is a knight (3) Lucas is a knight </answer>

Table 3: Example of a reasoning trace for the Knights & Knaves logic puzzle generated with QwQ-32B teacher model. As seen here, the response is quite verbose with complex reasoning behaviors like self-reflection, hypothesis testing, and so on.

A.2 Codebase

We use the TRL package[3] with the DeepSpeed ZeRO Stage 3 (Rajbhandari et al., 2020) configuration for Supervised Fine-tuning and Reinforcement Learning with GRPO (Shao et al., 2024).

The complete codebase for this work is available at `https://anonymous.4open.science/r/warmup-before-you-train-0EEF/`.

A.3 Evaluation

We keep the temperature 0.7 and top-p 0.95 for both training and evaluation. Given the sensitivity during evaluation, we sample 4 times for each question and report the average accuracy along with the standard deviation.

We use the `math_verify` package for math evaluation. For HumanEval$^+$, we base our evaluation on OpenAI's codebase[4]. Finally, for MMLU-Pro, we adapt our evaluation from MMLU-Pro's official repository[5]. We use the `vllm` package for running inference for evaluation on H100 GPUs.[6] All evaluation code is made available in the shared repository.

A.4 Distillation Experiment Details

For all distillation experiments, we used the hyperparameters as shown in Table 4.

Hyperparameter	Value
Optimizer	AdamW
Weight Decay	$1e-4$
Warmup Steps	25
Max Sequence Length	32,768
Batch Size (per device)	1
Gradient Accumulation Steps	2
Number of Epochs	3
Learning Rate Scheduler	constant
bf16	True
GPUs	6 H100s

Table 4: Key hyperparameters used in supervised fine-tuning

For learning rate, we choose $1e-6$ for distillation with K&K, and choose $2e-5$ for distillation with s1K data as they showed the best results in empirical observations.

A.4.1 Warmup with K&K on Other Models

Recent studies have shown that Qwen model families show "unsual" behaviors, bringing into question whether patterns observed in Qwen models are generally applicable (Shao et al., 2025). Thus, to further confirm the validity of our approach, we experiment with warmup on other model families as well as shown in Table 5. We find that our warmup with K&K data brings strong improvement in non-Qwen model families as well.

A.4.2 Overfitting

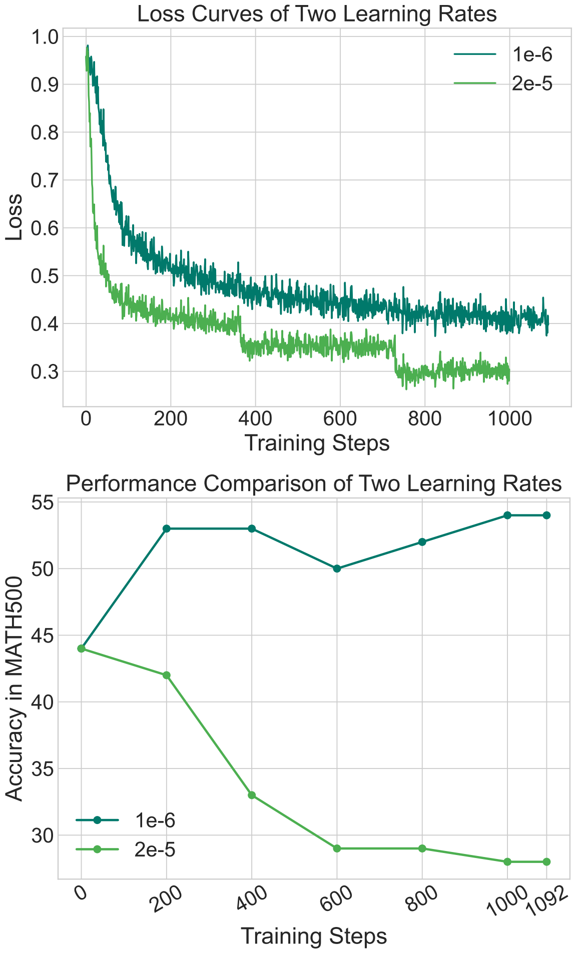


Figure 5: Results of Qwen2.5-3B K&K distillation. Loss curve shown on **top** & performance on MATH500 shown on **bottom**. Choosing a higher learning rate, $2e-5$ has a chance of overfitting to the K&K domain rather than learning generalizable reasoning behaviors.

We found that choosing a low learning rate was the best choice for distillation with K&K data. As shown in Figure 5, a lower learning rate leads to a sharp improvement in the performance of the base model in the MATH500 dataset. Since the data is specific to the K&K domain and might have lower

[3] `https://huggingface.co/docs/trl/en/index`
[4] `https://github.com/openai/human-eval`
[5] `https://github.com/TIGER-AI-Lab/MMLU-Pro`
[6] `https://docs.vllm.ai/en/latest/`

Model	**MATH (%)**	**HumanEval⁺ (%)**	**MMLU-Pro (%)**
Olmo2-1B	4.6 ± 0.4	0.0	11.7 ± 0
K&K	**5.5 ± 0.5** ↑	0.0	**11.0 ± 0.3** ↓
Olmo2-7B	14.3 ± 0.7	3.9 ± 0.5	20.0 ± 0.5
K&K	**21.4 ± 1.1** ↑	**4.8 ± 1.1** ↑	**25.1 ± 0.1** ↑
Llama-3.2-3B	6.8 ± 0.3	0.9 ± 0	16.2 ± 0.6
K&K	**8.5 ± 1.5** ↑	**7.5 ± 1.5** ↑	**18.3 ± 0.3** ↑
Llama-3.1-8B	11.8 ± 0.3	13.1 ± 0.6	23 ± 0.2
K&K	**25.2 ± 0** ↑	**17.1 ± 1.9** ↑	**33.9 ± 0.7** ↑

Table 5: Performance of non-Qwen model families distilled with the K&K game data. Best performance highlighted in bold.

token diversity, it might be possible that a higher learning rate causes the model to overfit quickly to the K&K task. In comparison, running distillation with s1K data with the learning rate of $2e-5$, keeping the rest of the hyperparameters constant (Table 4), did not lead to drastic overfitting.

To explore another alternative for hyperparameters, we followed the hyperparameter setting from the s1 paper (Muennighoff et al., 2025), choosing a learning rate of $1e-5$ followed by a cosine decay schedule in contrast to our constant schedule, and training for 5 epochs instead of 3 (Table 4). Following these hyperparameters, we distilled on the Qwen2.5-14B model with s1K data. We found that the model still underperformed our K&K distillation. This new setting led to a performance of 48.3 ± 1.6 on MATH500, 31.4 ± 6.0 on HumanEval⁺, and 44.2 ± 0.5 on MMLU-Pro. With both old and new hyperparameter settings, distillation with s1K data on the 14B model still leads to a significant performance degradation relative to the base model. A deeper study on why this degradation happens in the 14B model might be necessary. We stick to our hyperparameter settings as we found that to give the best output overall for both distillations, with K&K and s1K, empirically.

A.4.3 Warmup with 1,000 K&K Samples

As mentioned in Appendix A.1, we use 4,500 K&K reasoning traces for distillation. This is a larger number than s1K data which composes of high-quality 1,000 data points. Since the questions for the logic game, K&K, can be generated arbitrarily, we end up using a high number of samples. Still, we look at the performance of distillation with just 1,000 examples (matching s1K data size) on the Qwen2.5-3B model. We still find strong performance gains throughout the three benchmarks: 51.8 ± 1.4 on MATH, 35.1 ± 1.2 on HumanEval⁺, and 32.9 ± 0.3 on MMLU-Pro. Although these results are slightly lower than the ones reported in Table 2, they are still substantial.

A.5 Distillation with R1 reasoning traces

To ensure that the observed performance gains were not merely due to potential synergies between the teacher and student models from the Qwen family, we evaluated the effect of using reasoning traces from a teacher model of a different origin. In particular, we considered the possibility that the specific language patterns or token distributions in QwQ's responses might be especially well-suited to eliciting reasoning behavior in Qwen models. To test this, we employed DeepSeek-R1-Distill-Qwen-32B—a model distilled on reasoning traces generated by DeepSeek R1 (Guo et al., 2025)—to produce K&K traces for warmup, following the procedure described in Section 2. These traces were then distilled into the Qwen2.5-3B model.

This setup still resulted in consistent improvements across all benchmarks: accuracy on MATH rose to 51.2 ± 1.6, HumanEval⁺ to 53.5 ± 4.3, and MMLU-Pro to 34.6 ± 0.5. These results suggest that the warmup benefits are not limited to teacher-student alignment within the same model family.

A.6 RLVR Experiment Details

We use the R1 prompt (Table 6) across our training and evaluations (Guo et al., 2025). We modify the prompt slightly based on the task for better instruction following.

For RLVR, we use the hyperparameters shared in Table 7.

R1 prompt
A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in mind and then provides the user with the answer. The reasoning process and answer are enclosed within `<think>` `</think>` and `<answer>` `</answer>` tags, respectively, i.e., `<think>` reasoning process here `</think>` `<answer>` final answer inside `\boxed{}` tag `</answer>`. User: You must put your answer inside `<answer>` `</answer>` tags, i.e., `<answer>` answer here `</answer>`. And your final answer will be extracted automatically by the `\boxed{}` tag. `{prompt}` Assistant: `<think>` `</answer>`

Table 6: Prompt used across training and evaluations

Hyperparameter	Value
optimizer	AdamW
learning_rate	1e-6
beta	0
adam_beta1	0.9
adam_beta2	0.99
weight_decay	0.1
warmup_steps	25
Learning Rate Scheduler	constant
bf16	True
bf16_full_eval	True
Batch Size (per device)	4
Gradient Accumulation Steps	5
Gradient Checkpoint	True
num_generations	10
max_prompt_length	256
max_completion_length	8192
max_grad_norm	0.1
use_vllm	True
vllm_max_model_len	8192
temperature	0.7
top-p	0.95
GPUs	6 H100s

Table 7: RLVR training hyperparameters

Training curves of reward and completion length for MATH (Figure 6), HumanEval$^+$ (Figure 7), Physics (Figure 8), and History (Figure 9) are shown below.

Task	Base Model	Warmed-up Model
MATH	2:00	4:00
History	3:30	3:30
Physics	6:30	11:00
HumanEval$^+$	3:30	4:00

Table 8: RLVR Training Times (hh:mm) for Base vs. Warmed-up Models Across Tasks

Accuracy and completion lengths (in tokens) for MATH (Table 9 and Table 10), HumanEval$^+$ (Table 11 and Table 12), Physics (Table 13 and Table 14), and History (Table 15 and Table 16) are shown below.

A.7 Compute Costs

The computational requirements varies across models. The smallest model 1.5B takes around 20 minutes for K&K distillation with the largest 14B model taking around 1 hour 10 minutes. Since the s1K data only has 1K examples, it takes around 5 minutes for the 1.5B model and around 25 minutes for the 14B model. We use 6 NVIDIA H100 SXM GPUs (80GB VRAM) from runpod[7] coming at a cost of $2.99 per hour per GPU.

We use the same setup of 6 H100 GPUs for RLVR training. For a specific subject, we train both the base and warmed-up model for the same amount of training steps. Duration of training is shown in Table 8.

A.8 History Samples

After RLVR training the base model on history questions, it learns direct answering as it seems to be the most efficient strategy (Wei et al., 2022; Wang et al., 2024). However, since the warmed-up model is trained on reasoning, it struggles to learn the most efficient strategy as it is biased towards generating more tokens. A sample of both of these cases is shown in Table 17.

[7] https://www.runpod.io/pricing

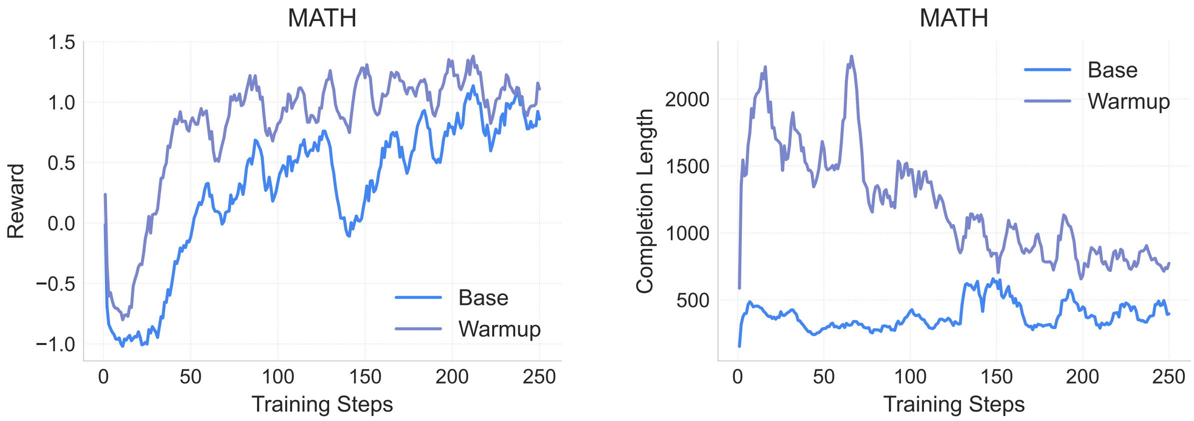


Figure 6: Training curves for Reward **(left)** and Completion Length **(right)** for MATH training, smoothed with a moving average (window size 10)

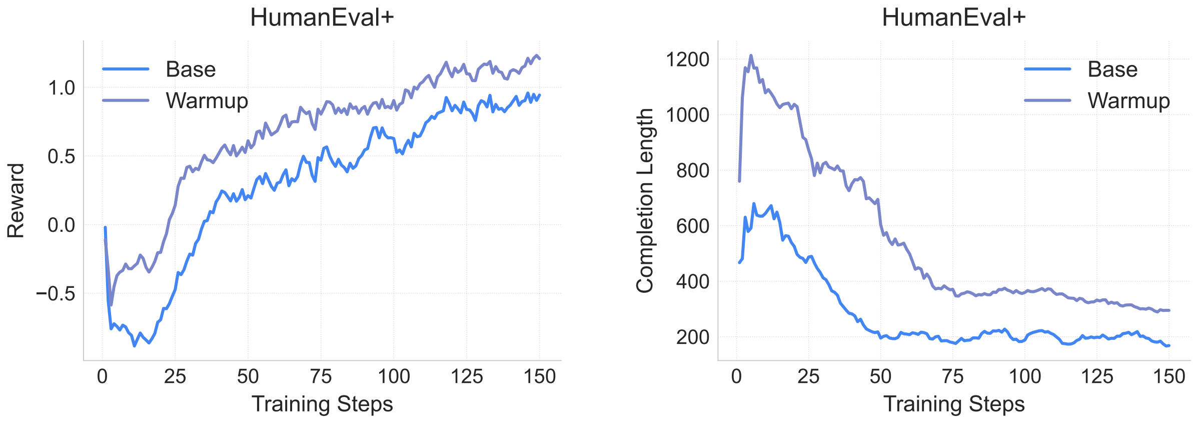


Figure 7: Training curves for Reward **(left)** and Completion Length **(right)** for HumanEval$^+$ training, smoothed with a moving average (window size 10)

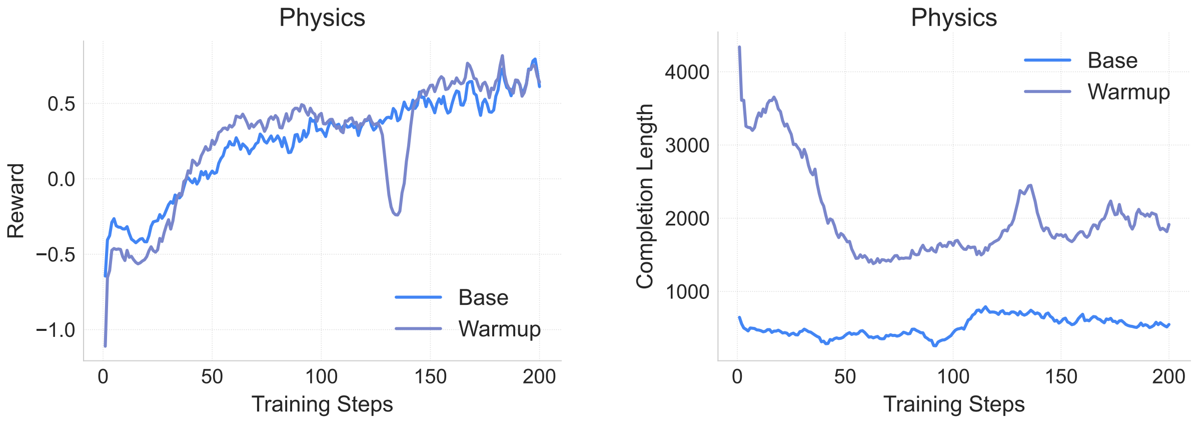


Figure 8: Training curves for Reward **(left)** and Completion Length **(right)** for Physics training, smoothed with a moving average (window size 10)

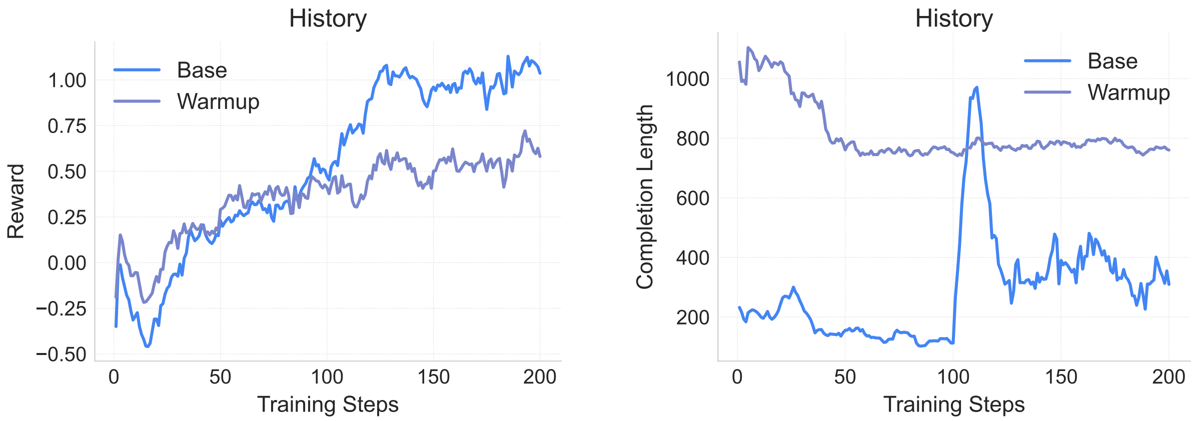


Figure 9: Training curves for Reward **(left)** and Completion Length **(right)** for History training, smoothed with a moving average (window size 10)

Model	**MATH (%)**	**HumanEval⁺ (%)**	**MMLU-Pro (%)**
Base Model	43.8 ± 0.8	32.5 ± 1.2	29.2 ± 0.3
+ RL	57.8 ± 1.9	25.7 ± 4.4	28.8 ± 0.2
Warmup	54.0 ± 1.4	47.8 ± 3.7	38.2 ± 0.4
+ RL	64.5 ± 0.3	31.1 ± 4.8	39.2 ± 0.3

Table 9: Performance comparison of training with MATH on Base vs Warmup Model

Model	**MATH**	**HumanEval⁺**	**MMLU-Pro**
Base Model	696.9 ± 84.8	312.6 ± 67.1	309.6 ± 13.6
+ RL	503.9 ± 19.8	307.5 ± 38.5	272.2 ± 5.8
Warmup	1623.5 ± 19.7	1205.9 ± 94.0	2203.7 ± 24.5
+ RL	1205.8 ± 33.4	1235.3 ± 31.6	1629.6 ± 7.1

Table 10: Completion Length after training with MATH on Base vs Warmup Model

Model	**HumanEval⁺ (%)**	**MATH (%)**	**MMLU-Pro (%)**
Base Model	32.5 ± 1.2	43.8 ± 0.8	29.2 ± 0.3
+ RL	56.8 ± 3.8	30.0 ± 0.9	27.4 ± 0.9
Warmup	47.8 ± 3.7	54.0 ± 1.4	38.2 ± 0.4
+ RL	61.8 ± 3.9	52.6 ± 1.5	36.2 ± 0.3

Table 11: Performance comparison of training with HumanEval⁺ on Base vs Warmup Model

Model	**HumanEval⁺**	**MATH**	**MMLU-Pro**
Base Model	312.6 ± 67.1	696.9 ± 84.8	309.6 ± 13.6
+ RL	172.4 ± 2.0	328.0 ± 21.3	81.3 ± 0.8
Warmup	1205.9 ± 94.0	1623.5 ± 19.7	2203.7 ± 24.5
+ RL	351.0 ± 11.6	1269.2 ± 29.6	1336.4 ± 12.8

Table 12: Completion Length after training with HumanEval⁺ on Base vs Warmup Model

Model	Physics (%)	MMLU-Pro (%)	MATH (%)	HumanEval$^+$ (%)
Base Model	24.4 ± 2.6	29.2 ± 0.3	43.8 ± 0.8	32.5 ± 1.2
+ RL	33.6 ± 1.1	36.0 ± 0.3	46.8 ± 0.9	38.4 ± 3.0
Warmup	34.4 ± 1.2	38.2 ± 0.4	54.0 ± 1.4	47.8 ± 3.7
+ RL	41.6 ± 1.6	40.8 ± 0.3	54.3 ± 2.1	48.5 ± 6.1

Table 13: Performance comparison of training with Physics on Base vs Warmup Model

Model	Physics	MMLU-Pro	MATH	HumanEval$^+$
Base Model	448.1 ± 37.8	309.6 ± 13.6	696.9 ± 84.8	312.6 ± 67.1
+ RL	459.9 ± 30.5	274.5 ± 7.7	601.1 ± 46.4	257.2 ± 42.2
Warmup	3607.8 ± 30.5	2203.7 ± 24.5	1623.5 ± 19.7	1205.9 ± 94.0
+ RL	1821.3 ± 30.5	1348.1 ± 7.6	1365.5 ± 45.3	1371.2 ± 38.3

Table 14: Completion Length after training with Physics on Base vs Warmup Model

Model	History (%)	MMLU-Pro (%)	MATH (%)	HumanEval$^+$ (%)
Base Model	24.9 ± 4.8	29.2 ± 0.3	43.8 ± 0.8	32.5 ± 1.2
+ RL	39.7 ± 1.9	35.3 ± 0.2	35.3 ± 1.0	37.1 ± 3.2
Warmup	30.0 ± 3.3	38.2 ± 0.4	54.0 ± 1.4	47.8 ± 3.7
+ RL	35.7 ± 1.6	40.5 ± 0.5	50.5 ± 1.8	49.8 ± 3.9

Table 15: Performance comparison of training with History on Base vs Warmup Model

Model	History	MMLU-Pro	MATH	HumanEval$^+$
Base Model	207.2 ± 44.1	309.6 ± 13.6	696.9 ± 84.8	312.6 ± 67.1
+ RL	360.6 ± 59.3	277.9 ± 10.5	498.0 ± 55.8	199.5 ± 39.9
Warmup	1083.1 ± 129.0	2203.7 ± 24.5	1623.5 ± 19.7	1205.9 ± 94.0
+ RL	818.2 ± 5.7	1112.1 ± 7.9	1117.1 ± 48.3	862.8 ± 42.6

Table 16: Completion Length after training with History on Base vs Warmup Model

Question The ancient city of Cahokia contained more than ________ and a population of around ________.
Options:

A. ’40 residential complexes; 40,000’

B. ’50 religious monuments; 5,000’

C. ’120 earthen mounds; 10,000’

D. ’70 public squares; 7,000’

E. ’80 ceremonial mounds; 50,000’

F. ’200 pyramidal mounds; 200,000’

G. ’60 burial mounds; 2,000’

H. ’30 agricultural terraces; 30,000’

I. ’150 defensive walls; 15,000 ’

J. ’100 trading posts; 1,000’

Correct Answer (C)

Sample Response after RLVR Training with History questions on base model
The ancient city of Cahokia was home to more than 120 earthen mounds. The population of Cahokia was around 10,000. </think><answer> C </answer>

Correct.

Sample Response after RLVR Training with History questions on warmed-up model
The question is about the ancient city of Cahokia, which was a significant Mississippian culture settlement in what is now the United States. The goal is to determine the correct answer from the given options regarding the number of residential complexes, religious monuments, earthen mounds, public squares, ceremonial mounds, pyramidal mounds, burial mounds, agricultural terraces, defensive walls, and trading posts. However, the options provided might be a mix of different types of features, and I need to find the pair that corresponds to the most accurate count for the city’s size.

First, let me recall that Cahokia was a major urban center in the 11th to 13th centuries AD. The city was notable for its large population and impressive architecture. The question asks for the numbers related to the city’s residential complexes, religious monuments, earthen mounds, etc., but the options don’t list the actual counts of these features. Therefore, I need to look up the historical data related to Cahokia. From historical records, the estimated population of Cahokia is around 15,000-20,000 people. However, the question’s options are about specific counts (like the number of residential complexes, religious monuments, etc.), not the population.

So, perhaps the options are meant to be about the architectural or ceremonial features that are associated with the city. Let me check each option one by one: A. 40 residential complexes; 40,000 - The population of Cahokia is much smaller, around 15,000, so this seems too high. B. 50 religious monuments; 5,000 - This also seems too high for a city with a population of 15,000. A. 40 residential complexes; 40,000 - The population of Cahokia is much smaller, around 15,000, so this seems too high. B. 50 religious monuments; 5,000 - This also seems too high for a city with a population of 15,000. C. 120 earthen mounds; 10,000 - Again, the population is around 15,000, so this is too high. D. 70 public squares; 7,000 - This is also too high. E. 80 ceremonial mounds; 50,000 - Again, the population is much smaller. F. 200 pyramidal mounds; 200,000 - This is a very high number for a city with around 15,000 inhabitants. G. 60 burial mounds; 2,000 - The population is around 15,000, so this could be possible but let's see other options. H. 30 agricultural terraces; 30,000 - The population is around 15,000, so this is also high. I. 150 defensive walls; 15,000 - The population is around 15,000, so this is possible. J. 100 trading posts; 1,000 - The population is around 15,000, so this is possible. Now, looking at the options, the ones that could be closer to the actual counts of architectural features in Cahokia would be I and J. However, the options don't list the actual numbers of each feature but rather their counts.

Since I and J are about defensive walls and trading posts, which are both architectural features, but not necessarily the most accurate count, perhaps I is more reasonable because defensive walls could be a significant part of the city's infrastructure. However, the question might expect a combination of residential complexes, religious monuments, etc., which are not provided. Alternatively, perhaps the question is testing knowledge of the approximate number of ceremonial mounds or public squares in Cahokia. Since the options are about different features, and the population is the only real constraint, I need to find the option that aligns with the city's size. Cahokia had a large number of earthen mounds, which were used for ceremonial purposes. The number of earthen mounds in Cahokia is estimated to be around 120. Let's see if this can be paired with a possible residential complex count. The user's question says ""The ancient city of Cahokia contained more than ________ residential complexes and a population of around ________."" So, the first part is the number of residential complexes, and the second part is the population. Given that the population is around 15,000, the number of residential complexes would likely be a significant number but not too high. Let me consider option G: 60 burial mounds. This is a lower number but not necessarily impossible, but the question suggests that the number of residential complexes might be higher. Since the options are about different features, perhaps I is the best choice here because it's about a tangible architectural feature that could be part of the city's layout, such as defensive walls. Therefore, the best choice among the given options is I: 150 defensive walls; 15,000. This is because the question asks for a combination of ""more than ________ residential complexes"" and ""a population of around ________,"" and the defensive walls could be a significant defensive structure in the city, contributing to the city's layout and thus the ""more than"" part.

The population is around 15,000, which aligns with the actual historical estimates for Cahokia. </think> <answer> \boxed{H} </answer>

Incorrect.

Table 17: Base model and warmed-up model responses on a History question after RLVR training on History. While the base model gives a very short response without using any Chain of Thought behavior at all, the warmed-up model outputs a long, complex reasoning chain that overcomplicates things. This shows that direct answering without any Chain of Thought seems to be the more favorable strategy for History questions.