Incentivizing Strong Reasoning from Weak Supervision

Anonymous ACL submission

Abstract

Large language models (LLMs) have demonstrated impressive performance on reasoning-intensive tasks, but enhancing their reasoning abilities typically relies on either reinforcement learning (RL) with verifiable signals or supervised fine-tuning (SFT) with high-quality long chain-of-thought (CoT) demonstrations, both of which are expensive. In this paper, we study a novel problem of incentivizing the reasoning capacity of LLMs without expensive high-quality demonstrations and reinforcement learning. We investigate whether the reasoning capabilities of LLMs can be effectively incentivized via supervision from significantly weaker models. We further analyze when and why such weak supervision succeeds in eliciting reasoning abilities in stronger models. Our findings show that supervision from significantly weaker reasoners can substantially improve student reasoning performance, recovering close to 94% of the gains of expensive RL at a fraction of the cost. Experiments across diverse benchmarks and model architectures demonstrate that weak reasoners can effectively incentivize reasoning in stronger student models, consistently improving performance across a wide range of reasoning tasks. Our results suggest that this simple weak-to-strong paradigm is a promising and generalizable alternative to costly methods for incentivizing strong reasoning capabilities at inference-time in LLMs. Code is publicly available at this link.

1 Introduction

Large language models (LLMs) have demonstrated strong performance across a variety of reasoning-intensive tasks, such as mathematical problem solving (Guo et al., 2025; Yang et al., 2024b), symbolic computation (Fang et al., 2024), and code generation (Jiang et al., 2024), often achieving results competitive with or even surpassing human-level capabilities (Xiao et al., 2025a, 2024b, 2025b, 2024a). A key technique for enabling such reasoning abilities is to scale up inference compute via long chain-of-thought (CoT), that encourages models to explicitly generate intermediate reasoning steps before arriving at a final answer (Guo et al., 2025; Team et al., 2025; Wei et al., 2022). This approach has proven effective in improving answer accuracy and enabling the decomposition of complex problems into more manageable subproblems.

A mainstream way to incentivize the reasoning ability of LLMs is *reinforcement learning* (RL) with verifiable reward signals, which shows strongest empirical gains but at the price of substantial compute and data engineering (Guo et al., 2025; Team et al., 2025). In addition to the high computational cost of thousands of GPU-hours per run, RL is effective only when the base model can discover correct trajectories during roll-outs, yet many open-source models fail to meet this assumption. Recent studies further indicate that while RL with verifiable reward can boost sampling efficiency, it may simultaneously limit the exploration capacity, resulting in a narrower reasoning capability boundary compared to base models (Shao et al., 2024; Yue et al., 2025). A more compute-friendly alternative strategy is *supervised fine-tuning* (SFT) on high-quality chain-of-thought (CoT) data distilled from a teacher model (Yeo et al., 2025; Ye et al., 2025; Muennighoff et al., 2025). However, collecting high-quality CoT data remains challenging in specialized domains, where human evaluation is time-consuming or costly. Obtaining reasoning demonstrations from frontier (i.e., extremely large) teacher models can also be prohibitively expensive, especially at scale. Moreover, the assumption of high-quality supervision from stronger models may not hold for superhuman models (Burns et al.). This persistent trade-off remains unsolved and calls for a fresh perspective for incentivizing reasoning.

As an alternative, in this work, we ask a fundamental yet unexplored question: *can we incentivize*

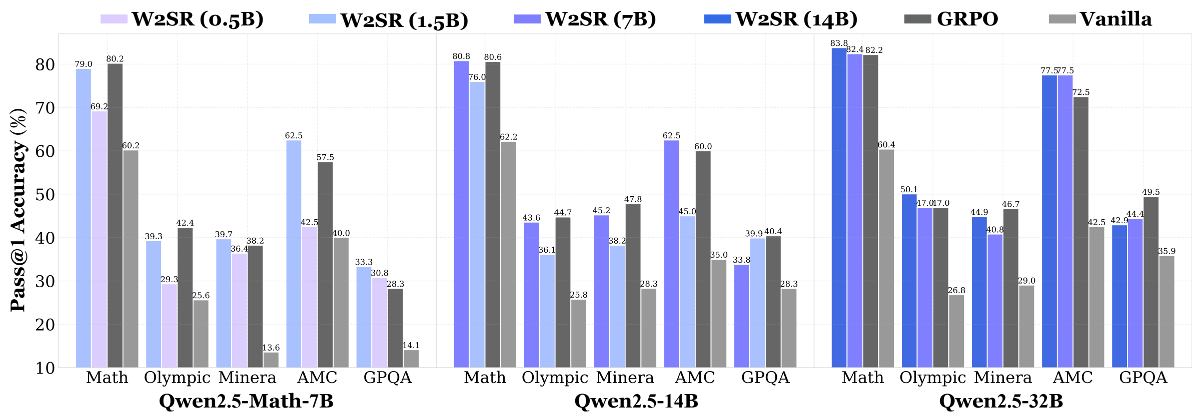

Figure 1: Benchmark performance of W2SR across student scales (Qwen2.5-Math-7B, Qwen2.5-14B, and Qwen2.5-32B). Each student model is trained with weak teachers: 7B uses 0.5B/1.5B teachers, 14B uses 1.5B/7B teachers, and 32B uses 7B/14B teachers. We compare W2SR to GRPO and vanilla models on five reasoning benchmarks. Weak supervision consistently yields strong reasoning, rivaling RL with far lower cost.

the reasoning capacity of LLMs without expensive strong teacher models and reinforcement learning?

To address this question, we investigate whether the reasoning capacities of LLMs can be effectively incentivized through supervision from significantly weaker models. Specifically, we propose a lightweight and scalable Weak-to-Strong Reasoning paradigm (W2SR), where a strong student model is trained on CoT trajectories generated by much weaker teacher models. Our hypothesis is that, even if a weak reasoner underperforms relative to the student or is considerably smaller in size, it can still provide imperfect yet informative reasoning traces that help elicit the student's reasoning abilities.

To investigate our hypothesis, we conduct extensive empirical studies using supervised fine-tuning (SFT) on strong student models with long CoTs distilled from weak teachers. As shown in Figure 1, our analysis across multiple reasoning benchmarks reveals that weak supervision can incentivize strong reasoning abilities. For example, distilling reasoning traces from a 0.5B–1.5B teacher enables a 7B–32B student to recover up to 94.34% of the reasoning gains from reinforcement learning (e.g., +18.8 Pass@1 on MATH), outperforming both the teacher and, in some cases, high-cost RL baselines. We further investigate when weak supervision is most effective. Results show that teacher reasoning ability, such as producing structurally well-formed CoTs, is more important than model size or final accuracy. Surprisingly, even imperfect or partially incorrect traces can incentivize reasoning in the student, while increasing teacher strength beyond a moderate level yields diminishing returns.

Our findings advocate a lightweight and scalable paradigm for incentivizing reasoning LLMs. Instead of collecting high-quality CoTs form expensive strong teachers or running costly RL, we distill *coarse yet structured* CoTs from much weaker supervision, which is sufficient to unlock the student's latent reasoning circuits. To sum up, our contributions are three-fold: (i) We *formulate and validate* weak-to-strong reasoning distillation, demonstrating that strong reasoning abilities can be incentivized from weak supervision to $4\times$ smaller and less accurate. (ii) Through Extensive ablations and analysis, we reveal *the surprisingly key to effective supervision*: the teacher's reasoning ability rather than the model size or performance. (iii) We demonstrate *practical benefits*: The new paradigm achieves drastic cost reduction compared to both SFT and RL, while outperforming the teacher model and even high-cost RL baselines in some cases, providing a privacy-friendly path for domain experts to refine frontier models using lightweight local teachers. Together, these contributions position weak-to-strong distillation as a promising paradigm for the lightweight and widely accessible reasoning LLMs.

2 Related Work

2.1 Large Reasoning Models

Large language models (LLMs) have demonstrated profound capabilities in many domains. A recent line of research aims to enhance reasoning capabilities through inference-time scaling, as demonstrated by OpenAI's o1(Jaech et al., 2024), DeepSeek's R1(Guo et al., 2025), and Kimi K1.5 (Team et al., 2025). These methods typically prompt models to generate ultra-long reasoning traces to solve complex problems. At the training stage, reasoning abilities are generally improved using two main strategies: (1) reinforcement learning with verifiable rewards (RLVR) (Guo et al., 2025; Lambert et al., 2024), and (2) supervised fine-tuning (SFT) on human-curated or model-distilled data(Guo et al., 2025; Ye et al., 2025; Muennighoff et al., 2025). While both approaches have shown promise, each comes with notable limitations. In particular, RL-based fine-tuning is significantly more computationally intensive, often requiring thousands of GPU-hours per run. A more compute-efficient alternative is to apply SFT on high-quality chain-of-thought (CoT) data distilled from a stronger teacher model (Ye et al., 2025; Muennighoff et al., 2025; Guo et al., 2025). However, collecting such data remains challenging, especially in specialized domains where human annotation is costly and time-consuming. Moreover, extracting reasoning traces from frontier (extremely strong) teacher models incurs substantial computational overhead, especially when scaling to large datasets, and may be infeasible if the model is already highly capable. *In contrast to prior work, we shift focus to a novel and underexplored direction: understanding and leveraging the reasoning patterns of weaker models within a weak-to-strong generalization paradigm. Our work investigates whether and how weak supervision can incentivize strong reasoning capabilities.*

2.2 Weak to Strong Generalization

Weak-to-strong generalization refers to scenarios where stronger models surpass their weaker supervisors after fine-tuning (Burns et al.). This paradigm has emerged as a promising framework for addressing the challenges of AI alignment, particularly in the context of superalignment (Leike et al., 2018), where future AI systems may exceed human capabilities, rendering human supervision inherently weak or insufficient. The weak-to-strong approach leverages weaker models to guide the training of stronger models, with the potential to unlock advanced capabilities while maintaining alignment with human values. This framework has been extensively explored through algorithmic innovations (Liu and Alahi, 2024; Guo and Yang, 2024a), empirical studies (Tao and Li, 2024; Yang et al., 2024c; Ye et al., 2024), and theoretical analyses (Lang et al., 2024). (Yang et al., 2024c) studies reasoning tasks within the weak-to-strong learning framework. However, their approach still relies on supervised fine-tuning using a selectively curated high-quality dataset. Moreover, prior work has largely focused on relatively simple tasks without requiring long CoT, leaving open the question of whether and how strong reasoning with long CoT can be effectively induced through weak supervision. In this paper, we present the first study that fine-tunes LLMs using long CoT trajectories generated by significantly weaker reasoners, and demonstrate that such weak supervision can substantially enhance general reasoning abilities. *Our approach is orthogonal to existing strategies, providing a compute-efficient and scalable alternative for incentivizing strong reasoning capabilities with inference-time scaling.*

Due to space constraints, we provide additional related works and discussions in Appendix F.

3 Incentivizing Strong Reasoning Capacity from Weak Supervision

3.1 Weak-to-Strong Reasoning Paradigm

Let $\mathbf{x} = \{\mathbf{x}_i\}_{i=1}^N$ denote a set of N input questions, where each question $\mathbf{x}_i = (x^1, x^2, \ldots, x^T)$ is represented as a sequence of T input tokens. The corresponding ground-truth answers are given by $\mathbf{a} = \{\mathbf{a}_i\}_{i=1}^N$. LLM policy π_θ generates output sequences $\mathbf{y} = \{\mathbf{y}_i\}_{i=1}^N$. For each output sequence $\mathbf{y}_i$, the model defines a conditional probability distribution over tokens $\pi_\theta(y^t \mid \mathbf{x}, \mathbf{y}^{<t})$, where y^t is the token at position t, and $\mathbf{y}^{<t}$ denotes the preceding tokens. From each $\mathbf{y}_i$, we extract a substring $\hat{\mathbf{a}}_i \subseteq \mathbf{y}_i$ as the predicted answer for question i. We define the CoT as a subsequence of $\mathbf{y}_i$ that contains intermediate reasoning steps (i.e., reasoning trajectories) leading to the final answer $\hat{\mathbf{a}}_i$.

In this work, we propose a simple weak-to-strong reasoning (`W2SR`) paradigm. For each input question $\mathbf{x}_i$, a weak reasoner (the *teacher*) generates a long CoT trajectory $\mathbf{y}_i$, which we use as weak supervision to fine-tune a stronger student model π_{θ_s} that initially lacks explicit incentivization for reasoning. The student is trained via simple SFT to imitate the teacher's reasoning trajectories, despite their potential imperfections. The goal of the `W2SR` paradigm is to incentivize stronger reasoning capabilities in the student through weak yet structured supervision provided by the teacher.

Note that we define a *weak reasoner* as a teacher model with *explicitly incentivized but limited reasoning abilities*, typically due to smaller model size, lower answer accuracy, or both. Conversely, a *strong student* is a model with greater capacity or better performance relative to its teacher. We refer to teacher models with explicitly incentivized reasoning capabilities as `-Reasoner` or `(R)`, and those without such capabilities as `-Non-Reasoner`. To evaluate the effectiveness of our `W2SR` paradigm, we investigate three *weak-to-strong* variants based on the correctness of the teacher's final answer:

- `W2SR`: Uses all CoT trajectories from the weak teacher, regardless of answer correctness:

$$\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}, \text{ where } \mathbf{y}_i = \pi_{\theta_t}(\mathbf{x}_i). \quad (1)$$

- `W2SR-P`: Uses only CoT trajectories that yield the correct final answer from the weak teacher:

$$\mathcal{D}_p = \{(\mathbf{x}_i, \mathbf{y}_i) \mid \hat{\mathbf{a}}_i = \mathbf{a}_i\}. \quad (2)$$

- `W2SR-N`: Uses only trajectories with incorrect final answers from the weak teacher:

$$\mathcal{D}_n = \{(\mathbf{x}_i, \mathbf{y}_i) \mid \hat{\mathbf{a}}_i \neq \mathbf{a}_i\}. \quad (3)$$

The *student* model π_{θ_s} is fine-tuned on the above CoT data via simple SFT, which updates its parameters θ_s by minimizing the following negative log-likelihood (NLL) loss.

$$\mathcal{L}_{\text{SFT}}(\theta_s) = -\sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}} \sum_{t=1}^{|\mathbf{y}|} \log \pi_{\theta_s}(y^t \mid \mathbf{x}, \mathbf{y}^{<t}) \quad (4)$$

3.2 Evaluation via Reasoning Gap Recovered

To evaluate the effectiveness of the `W2SR` paradigm, we introduce a metric called Reasoning Gap Recovered (RGR). RGR measures how much of the reasoning improvement achieved through RL can be recovered by supervising a strong student with a weak reasoner. It quantifies the extent to which weak-to-strong training closes the performance gap relative to a student whose reasoning capabilities are incentivized via computationally intensive RL. Specifically, we define RGR based on:

- `Weak Reasoner`: The weak teacher model with reasoning capability incentivized via RL.
- `Weak-to-Strong Reasoner` (`W2SR`): Stronger student model trained via SFT on noisy or imperfect CoT traces produced by weak reasoner.
- `Strong Reasoner`: The same strong student trained with RL, serving as performance ceiling.

The following RGR metric quantifies the performance gap of the Pass@1 score between a weak reasoner and two strong reasoners that are trained with weak supervision and reinforcement learning (RL), respectively. Higher RGR indicates that weak supervision from the weak reasoner effectively recovers or even exceeds the performance of the RL-trained strong reasoner. In contrast, lower RGR suggests that weak-to-strong training provides only limited improvement over the weak reasoner.

$$\texttt{RGR} = \frac{\texttt{Weak-to-Strong Reasoner} - \texttt{Weak Reasoner}}{\texttt{Strong Reasoner} - \texttt{Weak Reasoner}}. \quad (5)$$

Note that RGR can exceed 1 when the student trained under weak supervision outperforms its RL-trained counterpart, demonstrating that structured but imperfect traces from weaker models can sometimes elicit superior reasoning. Compared to raw accuracy, RGR normalizes performance gains relative to the RL upper bound, offering a faithful measure of reasoning-specific transfer.

3.3 Experimental Setups

Full experimental details are provided in Appendix B, all prompts are included in Appendix G.

Dataset We use the MATH dataset (Hendrycks et al.) with difficulty levels 3–5 as the training data, following (Zeng et al., 2025). This dataset encompasses seven math topics such as advanced calculus, geometry, and linear algebra.

Models The student models include three parameter sizes from the Qwen-2.5 family (Yang et al., 2024a): `Qwen2.5-Math-7B`, `Qwen2.5-14B`, and `Qwen2.5-32B`. The teacher models are from `Qwen-2.5` (0.5B, 1.5B, 7B, 14B, 32B), for each we evaluate both the base version and its reinforcement learning fine-tuned counterpart. All reinforcement learning fine-tuned models are sourced from SimpleRL-Zoo (Zeng et al., 2025), where GRPO (Shao et al., 2024) is the training algorithm.

Training Our knowledge distillation framework involves two stages: (1) data distillation with teacher models using vLLM (Kwon et al., 2023) (greedy decoding, temperature 0, top-p 1, max length 4096 tokens, one sample per input); and (2) supervised fine-tuning (SFT) of student models using LLaMA-Factory (Zheng et al., 2024) with batch size 128, learning rate 10^{-5}, for 5 epochs.

Evaluation We evaluate on five reasoning benchmarks, including three standard math datasets,

Table 1: Weak-to-Strong Reasoning Performance Comparison. Evaluation of three student models (7B–32B) trained with supervision from four weaker reasoner teachers (0.5B–14B) across five reasoning benchmarks. Reports Metrics of Pass@1 (%) and Reasoning Gap Recovered (RGR %) for (1) base teacher, (2) base student, as well as student trained with (3) GRPO, (4) LIMO and (5) our Weak-to-Strong Reasoning (`W2SR`, `W2SR-P`). "-" indicates RGR is inapplicable. Best performance is marked with **boldface**, second best of our method is marked with <u>underline</u>. (R) denotes the teacher possesses reasoning capabilities.

Datasets (→)	**Metrics (→)**		**Math**		**Olympic**		**Minera**		**AMC**		**GPQA**		**Average**	
Student (↓)	**Teacher / Method (↓)**		Pass@1	RGR	Pass@1	RGR	Pass@1	RGR	Pass@1	RGR	Pass@1	RGR	Pass@1	RGR
	Qwen2.5-0.5B (R)		32.20	–	9.78	–	9.93	–	22.50	–	31.82		21.25	–
	Qwen2.5-1.5B (R)		59.00	–	20.74	–	21.32	–	27.50	–	25.76	–	30.86	–
Qwen2.5 Math-7B	Student only	`Vanilla`	60.20	–	25.63	–	13.60	–	40.00	–	14.14	–	30.71	–
	Student only	`GRPO`	**80.20**	–	**42.37**	–	38.24	–	57.50	–	28.28	–	49.32	–
	LIMO	`SFT`	70.40	–	34.07	–	30.88	–	47.50	–	17.17	–	40.00	–
	Qwen2.5-0.5B (R)	`W2SR`	52.00	41.25	16.30	20.01	26.84	59.73	35.00	35.71	28.79	85.59	31.79	48.46
	Qwen2.5-0.5B (R)	`W2SR-P`	69.20	77.08	29.33	59.99	36.40	93.50	42.50	57.14	30.81	28.53	41.65	63.25
	Qwen2.5-1.5B (R)	`W2SR`	71.20	57.55	34.81	65.05	37.50	95.63	52.50	83.33	31.82	240.48	45.57	108.41
	Qwen2.5-1.5B (R)	`W2SR-P`	<u>79.00</u>	94.34	<u>39.26</u>	85.62	**39.71**	108.69	**62.50**	116.67	**33.33**	300.40	**50.76**	141.14
	Qwen2.5-1.5B (R)		59.00	–	20.74	–	21.32	–	27.50	–	25.76	–	30.86	–
	Qwen2.5-7B (R)		77.80	–	41.78	–	38.97	–	65.00	–	28.79	–	50.47	–
Qwen2.5 14B	Student only	`Vanilla`	62.20	–	25.78	–	28.31	–	35.00	–	28.28	–	35.91	–
	Student only	`GRPO`	80.60	–	**44.74**	–	**47.79**	–	60.00	–	**40.40**	–	**54.71**	–
	LIMO	`SFT`	75.60	–	<u>43.85</u>	–	31.25	–	52.50	–	26.26	–	45.89	–
	Qwen2.5-1.5B (R)	`W2SR`	70.20	51.85	32.89	50.63	32.72	43.07	47.50	61.54	29.80	27.60	42.62	46.94
	Qwen2.5-1.5B (R)	`W2SR-P`	76.00	78.70	36.15	64.21	38.24	63.92	45.00	53.85	<u>39.90</u>	96.58	47.06	71.45
	Qwen2.5-7B (R)	`W2SR`	80.00	78.57	42.07	9.80	41.54	29.14	57.50	150.00	28.28	-4.39	49.88	52.62
	Qwen2.5-7B (R)	`W2SR-P`	**80.80**	107.14	43.56	60.14	<u>45.22</u>	70.86	**62.50**	50.00	33.84	43.50	<u>53.18</u>	66.33
	Qwen2.5-1.5B (R)		59.00	–	20.74	–	21.32	–	27.50	–	25.76	–	30.86	–
	Qwen2.5-7B (R)		77.80	–	41.78	–	38.97	–	65.00	–	28.79	–	50.47	–
	Qwen2.5-14B (R)		80.60	–	44.74	–	47.79	–	60.00	–	40.40	–	54.71	–
Qwen2.5 32B	Student only	`Vanilla`	60.40	–	26.81	–	29.04	–	42.50	–	35.86	–	38.92	–
	Student only	`GRPO`	82.20	–	46.96	–	**46.69**	–	72.50	–	**49.49**	–	59.57	–
	LIMO	`SFT`	75.40	–	47.11	–	30.51	–	62.50	–	32.32	–	49.57	–
	Qwen2.5-1.5B (R)	`W2SR`	70.20	48.28	31.85	42.37	37.13	62.32	52.50	55.56	35.35	40.41	45.41	49.79
	Qwen2.5-1.5B (R)	`W2SR-P`	76.00	73.28	37.19	62.74	40.07	73.91	62.50	77.78	38.89	55.33	50.93	68.61
	Qwen2.5-7B (R)	`W2SR`	81.60	86.36	47.41	108.69	42.28	42.88	75.00	133.33	35.86	34.15	56.43	81.08
	Qwen2.5-7B (R)	`W2SR-P`	82.40	104.55	46.96	100.00	40.81	23.83	**77.50**	166.67	<u>44.44</u>	75.60	58.42	94.13
	Qwen2.5-14B (R)	`W2SR`	83.60	187.50	48.00	146.85	<u>45.59</u>	200.00	72.50	100.00	43.94	38.94	58.73	134.66
	Qwen2.5-14B (R)	`W2SR-P`	**83.80**	200.00	**50.07**	240.09	44.85	267.27	**77.50**	140.00	42.93	27.83	**59.83**	175.04

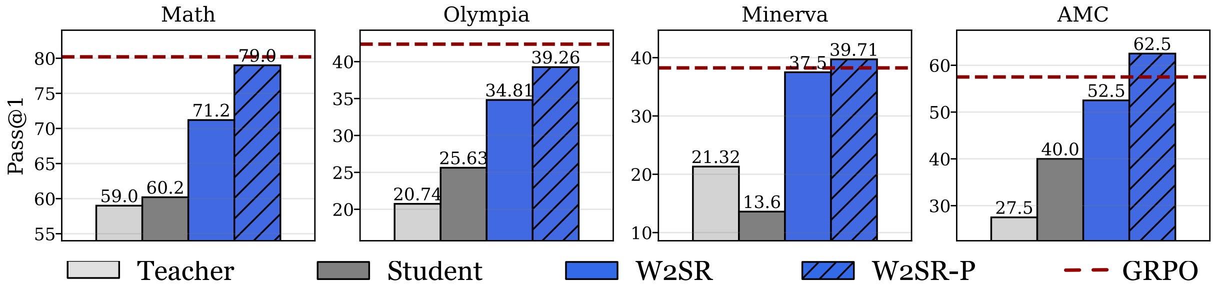

Figure 2: Using `W2SR` with `Qwen2.5-Math-7B` as the strong student and `Qwen2.5-1.5B-Reasoner` as the weak teacher, the four bars represent: (1) the weak teacher's standalone performance, (2) the strong student's standalone performance, (3) the student's performance after `W2SR`, and (4) the student's performance after `W2SR-P`. More results are provided in Appendix C.

MATH500 (Hendrycks et al.), OlympiaBench (He et al., 2024), MinervaMath (Lewkowycz et al., 2022a), plus the competition-level AMC2023 (Art of Problem Solving Foundation, 2023) and the non-mathematical GPQA (Rein et al., 2024). Evaluation uses the codebase from (Li et al., 2025), with sampling temperature 0.6, top-p 0.95, 1 sample per input, and max generation length 32,768 tokens.

4 Experiments

The remainder of paper analyzes weak-to-strong reasoning paradigm from three key perspectives: **(RQ1)** Can weak supervision incentivize reasoning in stronger models? **(RQ2)** What aspects of teacher supervision are most critical for incentivizing reasoning? **(RQ3)** What practical benefits and broader impacts arise from weak-to-strong reasoning?

4.1 RQ1: Can Weak Supervision Incentivize Reasoning in Stronger Models?

In this section, we examine whether high-capacity student models can improve their reasoning capabilities when trained on reasoning trajectories produced by significantly smaller and less capable teacher models. We also assess how effectively such weak-to-strong supervision can narrow the performance gap relative to models fine-tuned with RL). As illustrated in Table 1 and Figure 2, we evaluate three student models ranging from 7B to 32B parameters, each trained using supervision from multiple weaker teachers, whose sizes range from 0.5B to 14B parameters.

Based on the results in Table 1, we evaluate the effectiveness of weak-to-strong reasoning for each strong student model with respect to its corresponding weak teacher(s), where the teacher possesses reasoning capabilities (denoted as "R"). For each teacher-student pair, we report both Pass@1 and Relative Gain in Reasoning (RGR) metrics under both `W2SR` and `W2SR-P`. To ensure a comprehensive evaluation, we additionally assess model performance on LIMO (Ye et al., 2025), a benchmark constructed from high-quality CoT data.

As shown in Figure 2, we analyze the training of the `Qwen2.5-Math-7B` student with the `Qwen2.5-1.5B-Reasoner` as the teacher, comparing five settings: (1) the standalone weak teacher, (2) the standalone student, (3) the student trained via `W2SR`, (4) the student trained via `W2SR-P`, and (5) the student trained via RL. These comparisons reveal a striking effect, where combining a weak teacher with a strong student results in performance far exceeding that of either model alone. For instance, on MATH and AMC, the original student achieves Pass@1 of 60.20% and 40.00%, while the teacher scores 59.00% and 27.50%, respectively, both considerably lower. However, after `W2SR-P` training, despite the teacher being 4.7× smaller and performing 1.99% and 31.25% relatively worse than the student, it enables the student to reach Pass@1 of 79.00% and 62.50%, corresponding to relative improvements of 31.22% and 56.25%, reflected in RGR of 94.34% and 116.67%. Notably, on AMC, this improvement even surpasses the performance ceiling established by RL training.

Our findings reveal that teachers with structured reasoning abilities, regardless of model scale or final-answer accuracy, can effectively elicit and enhance reasoning capabilities in more powerful student models. This knowledge transfer enables students to not only surpass their teachers' performance but also exceed results from direct reinforcement learning on the student models themselves, aligning with the insight shown in Takeaway 1.

Table 2: Pass@1 Performance for three variants of Weak-to-Strong Reasoning: (1) All: Unfiltered reasoning trajectories (`W2SR`), (2) Correct Only: Reasoning trajectories filtered by correct answers (`W2SR-P`), and (3) Incorrect Only: Reasoning trajectories filtered by incorrect answers (`W2SR-N`).

Student	Teacher		Math	Olympic	Minera	AMC	GPQA	Average
Qwen2.5-Math-7B	Student only	Vanilla	60.20	25.63	13.60	40.00	14.14	30.71
	Qwen2.5-1.5B (R)	`W2SR` (✓ + ✗)	71.20	34.81	37.50	52.50	31.82	45.57
		`W2SR-P` (✓)	79.00	39.26	39.71	62.50	33.33	50.76
		`W2SR-N` (✗)	70.20	32.44	37.87	42.50	35.86	43.77
Qwen2.5 14B	Student only	Vanilla	62.20	25.78	28.31	35.00	28.28	35.91
	Qwen2.5-7B (R)	`W2SR` (✓ + ✗)	78.80	42.96	40.07	60.00	28.28	50.02
		`W2SR-P` (✓)	80.80	43.56	45.22	62.50	33.84	53.18
		`W2SR-N` (✗)	78.20	41.04	40.81	55.00	36.36	50.28
Qwen2.5 32B	Student only	Vanilla	60.40	26.81	29.04	42.50	35.86	38.92
	Qwen2.5-14B (R)	`W2SR` (✓ + ✗)	83.60	48.00	45.59	72.50	43.94	58.73
		`W2SR-P` (✓)	83.80	50.07	44.85	77.50	42.93	59.83
		`W2SR-N` (✗)	83.20	47.41	46.69	62.50	39.90	55.94

Takeaway 1: Weak Yet Incentivizable

Weak Supervision Can Incentivize Reasoning in Stronger Models: Reasoning trajectories from weaker teachers can effectively incentivize reasoning in stronger students and reach comparable/superior performance compared to expensive RL methods.

4.2 RQ2: Key Aspects of Teacher Supervision for Incentivizing Reasoning?

We ask what makes weak supervision effective for transferring reasoning to stronger students. Specifically: (1) Which attributes, reasoning ability, model scale, or task accuracy, are most critical for supervision? (2) Must supervision trajectories yield correct answers, or can incorrect ones still be useful?

4.2.1 What Makes a Good Teacher for Incentivizing Reasoning?

This section provides empirical evidence that the capability of a teacher model's explicit inference-time scaling plays a more critical role than parameter scaling or overall task accuracy.

(1) Reasoning Ability vs. Model Scale. As shown in the upper part of Figure 3, across four benchmarks, students trained with Reasoner teachers consistently outperform those trained with Non-Reasoner teachers, regardless of the teacher's parameter count. Notably, increasing the size of Non-Reasoner teachers from 1.5B to 32B yields no meaningful improvement in student performance. In most cases, students trained with the smallest Reasoner teacher (1.5B) surpass those trained with even the largest Non-Reasoner teacher (32B), despite the latter being 21× larger in model size. These results indicate that scaling model size alone is insufficient for effective supervision, what mat-

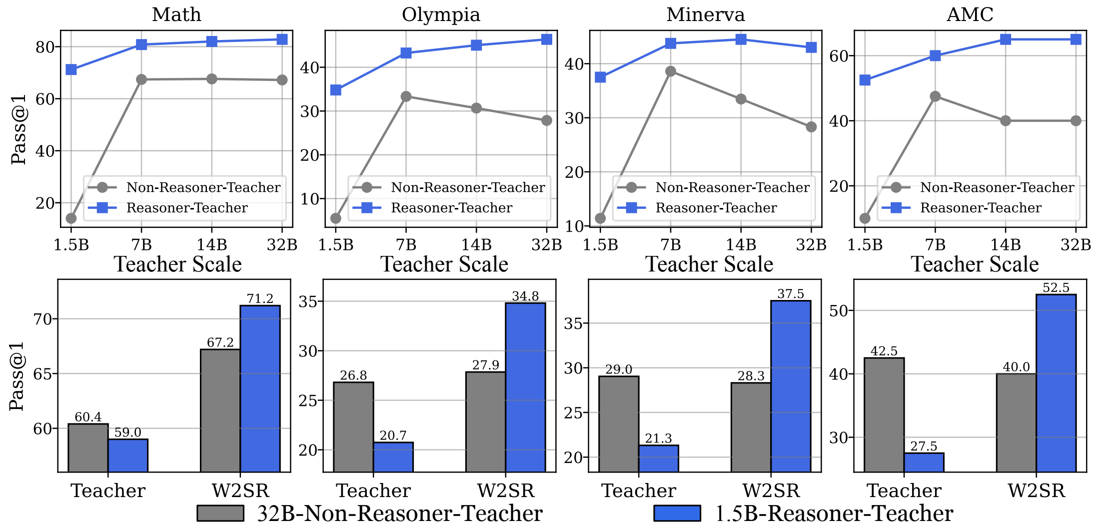

Figure 3: Comparison of Weak-to-Strong Reasoning (`W2SR`) between Reasoner and Non-Reasoner Teachers using `Qwen2.5-Math-7B` as the student. **Upper**: The x-axis represents teacher model scale, with two lines indicating the `W2SR` student performance trained with Reasoner vs. Non-Reasoner teachers of corresponding scales. **Lower**: The first two bars show initial performance of `Qwen2.5-1.5B-Reasoner` and `Qwen2.5-32B-Non-Reasoner` teacher, while the last two bars show the resulting `W2SR` student performance. More results are in Appendix C.

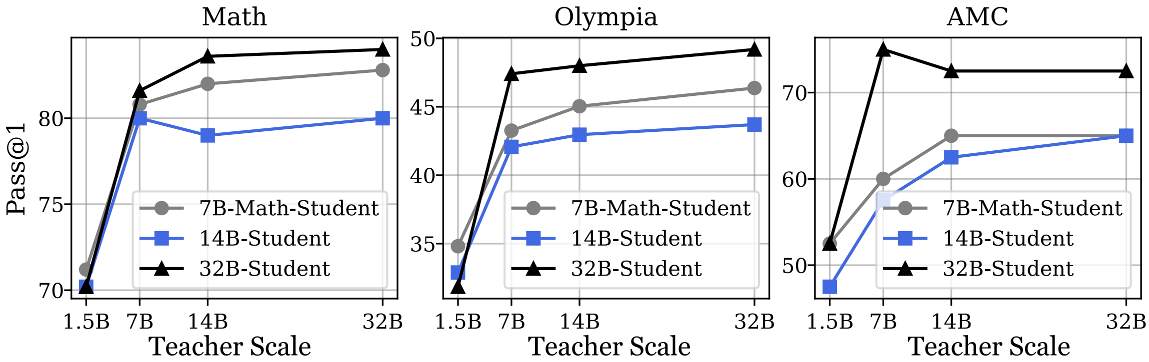

Figure 4: Diminishing marginal returns from increasing reasoner-teacher scale on student performance. The x-axis represents model sizes/scale of the reasoner teacher, increasing from 1.5B, 7B, 14B to 32. The y-axis shows the Pass@1 performance.

ters more is the presence of explicit reasoning trajectories in the teacher's outputs.

(2) Reasoning Ability vs. Task Performance. The lower part of Figure 3 highlights a direct comparison between two extreme cases: the small reasoner teacher (`Qwen2.5-1.5B-Reasoner`) and the large Non-Reasoner teacher (`Qwen2.5-32B-Non-Reasoner`). While the 1.5B Reasoner teacher performs worse in isolation, e.g., on AMC it scores 27.5%, compared to 32B's 42.5%, it still leads to significantly better student models across all four benchmarks. For example, the student trained with the 1.5B Reasoner teacher achieves a Math Pass@1 of 71.2%, compared to 67.2% for the one trained with the 32B Non-Reasoner as teacher. The effect is even more pronounced on Minerva and AMC, where using the 32B Non-Reasoner teacher actually results in student performance worse than the teacher by 2.41% and 5.88% relatively, the 1.5B Reasoner improves it by 76.06% and 90.91%.

These findings underscore that reasoning supervision, enabled by inference-time scaling through explicit CoT traces, is fundamentally more effective than simply increasing model size. Overall, our results affirm that inference scaling is more critical than parameter scaling in fostering reasoning capabilities in student models, echoing the insight summarized in Takeaway 2.

Takeaway 2: Reasoning > Size + Acc

Reasoning Capability Matters More Than Model Size or Accuracy: A teacher's explicit reasoning capability plays a more critical role than its parameter scale or overall task accuracy for effectively incentivizing student reasoning ability.

4.2.2 Is Correctness a Necessary Condition for Useful Reasoning Supervision?

We investigate whether effective reasoning supervision requires correct final answers or if reasoning traces with incorrect outcomes can still provide useful learning signals in weak-to-strong training.

As shown in Table 2, student models trained with `W2SR-N` consistently outperform vanilla students (trained without teacher supervision) across all benchmarks and model scales. For example, the `Qwen2.5-Math-7B` student distilled with incorrect traces from the `Qwen2.5-1.5B-Reasoner` teacher improves its average Pass@1 from 30.71% to 43.77%, even surpassing `W2SR`. Similarly, the `Qwen2.5-14B` student distilled from the `Qwen2.5-7B-Reasoner` teacher achieves Pass@1 scores of 50.02% with All, 53.53% with Correct Only, and still a strong 50.28% with Incorrect Only. These results demonstrate that intermediate reasoning steps can remain pedagogically valuable even when the final answers are incorrect.

Our findings support Takeaway 3, showing that the correctness of final answer alone is an insufficient criterion for high-quality supervision. Instead, structurally sound reasoning traces, even when yielding incorrect final answers, can effectively activate reasoning capabilities in student models.

> **Takeaway 3: Wrong Yet Helpful**
>
> **Incorrect Answers Can Still Teach Correct Reasoning**: Incorrect reasoning trajectories can still effectively incentivize the reasoning capability of the student model, showing that correctness is not essential for effective supervision.

4.3 RQ3: What Practical Benefits Arise from Weak-to-Strong Reasoning?

This section investigates the benefits of `W2SR` through the perspectives of effectiveness, efficiency and further analysis of cognitive behaviors and inference-time scaling (See Appendices C.1 and H)

4.3.1 Effectiveness: Beyond Strong Supervision and Base Model Constraints

Existing methods like SFT and RL face limitations: strong supervision becomes scarce as models improve, and RL mainly amplifies existing abilities without fostering novel reasoning (Yue et al., 2025). `W2SR` uses weak supervision to elicit reasoning beyond base model priors, enabling knowledge acquisition without relying on rare expert labels and overcoming SFT and RL constraints.

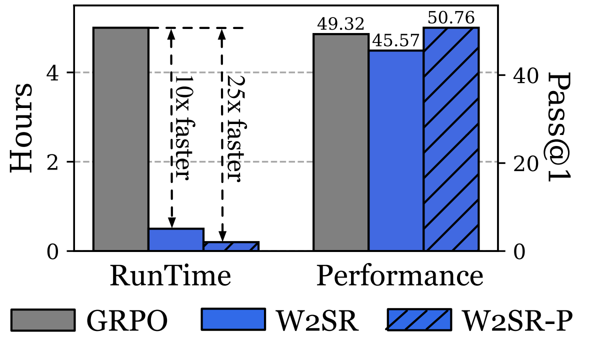

Figure 5: Comparison of efficiency (left bars) and performance (right bars) among GRPO, `W2SR`, and `W2SR-P`.

4.3.2 Efficiency: Weaker Teachers are Sufficient for Incentivization

Compared to costly, unstable RL and expert-dependent SFT, `W2SR` offers a more efficient alternative with strong performance. Scaling teachers yields diminishing returns, e.g., on Math with a `Qwen2.5-32B` student, increasing teacher size from 7B to 32B adds just 0.4%. In Figure 5, `W2SR-P` outperforms GRPO by 2.92% while reducing training time 25×. These results (Takeaway 4) highlight that weak teachers can effectively incentivize reasoning with better efficiency–performance trade-offs.

> **Takeaway 4: Weak Is Sufficient**
>
> **Weaker Teachers Offer Better Efficiency–Performance Trade-off**: Scaling teacher strength provides diminishing returns; notably, weak teachers often suffice for effectively incentivizing reasoning, significantly reducing computational cost.

5 Conclusion

We propose `W2SR`, a simple yet effective approach to enhance LLM reasoning by training with weak, structured reasoning trajectories instead of costly reinforcement learning or expert demonstrations. Despite relying on weaker supervision, models trained with `W2SR` outperform their teachers and rival RL-based methods with significantly less computation. This paradigm opens several directions for future research. We hope this work encourages further exploration into scalable and cost-efficient approaches for reasoning incentivization in large language models, paving the way toward scalable oversight that enables strong reasoning capabilities to be widely attainable.

References

Art of Problem Solving Foundation. 2023. Amc23 — 2023 american mathematics competitions test set. https://github.com/QwenLM/Qwen2.5-Math/tree/main/evaluation/data/amc23. 40 problems drawn from the 2023 AMC 12 contests.

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, and et al. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. In *Forty-first International Conference on Machine Learning*.

Ethan Chern, Haoyang Zou, Xuefeng Li, Jiewen Hu, Kehua Feng, Junlong Li, and Pengfei Liu. 2023. Generative ai for math: Abel. https://github.com/GAIR-NLP/abel.

Meng Fang, Shilong Deng, Yudi Zhang, Zijing Shi, Ling Chen, Mykola Pechenizkiy, and Jun Wang. 2024. Large language models are neurosymbolic reasoners. In *Proceedings of the AAAI conference on artificial intelligence*, pages 17985–17993.

Kaituo Feng, Changsheng Li, Xiaolu Zhang, Jun Zhou, Ye Yuan, and Guoren Wang. 2024. Keypoint-based progressive chain-of-thought distillation for llms. *arXiv preprint arXiv:2405.16064*.

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*.

Yue Guo and Yi Yang. 2024a. Improving weak-to-strong generalization with reliability-aware alignment. *arXiv preprint arXiv:2406.19032*.

Yue Guo and Yi Yang. 2024b. Improving weak-to-strong generalization with reliability-aware alignment. *arXiv preprint arXiv:2406.19032*.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. 2024. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, and 1 others. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Juyong Jiang, Fan Wang, Jiasi Shen, Sungju Kim, and Sunghun Kim. 2024. A survey on large language models for code generation. *arXiv preprint arXiv:2406.00515*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, and et al. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, and 1 others. 2024. T\" ulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.

Hunter Lang, David Sontag, and Aravindan Vijayaraghavan. 2024. Theoretical analysis of weak-to-strong generalization. *Advances in neural information processing systems*, 37:46837–46880.

Jan Leike, David Krueger, Tom Everitt, Miljan Martic, Vishal Maini, and Shane Legg. 2018. Scalable agent alignment via reward modeling: a research direction. *arXiv preprint arXiv:1811.07871*.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. 2022a. Solving quantitative reasoning problems with language models. *arXiv preprint arXiv:2206.14858*.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, and 1 others. 2022b. Solving quantitative reasoning problems with language models. pages 3843–3857.

Liunian Harold Li, Jack Hessel, Youngjae Yu, Xiang Ren, Kai-Wei Chang, and Yejin Choi. 2023. Symbolic chain-of-thought distillation: Small models can also "think" step-by-step. *arXiv preprint arXiv:2306.14050*.

Xuefeng Li, Haoyang Zou, and Pengfei Liu. 2025. Limr: Less is more for rl scaling. *arXiv preprint arXiv:2502.11886*.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Yuejiang Liu and Alexandre Alahi. 2024. Co-supervised learning: Improving weak-to-strong generalization with hierarchical mixture of experts. *arXiv preprint arXiv:2402.15505*.

Yijia Luo, Yulin Song, Xingyao Zhang, Jiaheng Liu, Weixun Wang, GengRu Chen, Wenbo Su, and Bo Zheng. 2025. Deconstructing long chain-of-thought: A structured reasoning optimization framework for long cot distillation. *arXiv preprint arXiv:2503.16385*.

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*.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2024. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, and et al. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Zhenyi Shen, Hanqi Yan, Linhai Zhang, Zhanghao Hu, Yali Du, and Yulan He. 2025. Codi: Compressing chain-of-thought into continuous space via self-distillation. *arXiv preprint arXiv:2502.21074*.

Leitian Tao and Yixuan Li. 2024. Your weak llm is secretly a strong teacher for alignment. *arXiv preprint arXiv:2409.08813*.

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*.

Jason Wei, Xuezhi Wang, Schuurmans, and et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Teng Xiao, Mingxiao Li, Yige Yuan, Huaisheng Zhu, Chao Cui, and Vasant G Honavar. 2024a. How to leverage demonstration data in alignment for large language model? a self-imitation learning perspective. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Teng Xiao, Yige Yuan, Zhengyu Chen, Mingxiao Li, Shangsong Liang, and et al. 2025a. SimPER: A minimalist approach to preference alignment without hyperparameters. In *International Conference on Learning Representations*.

Teng Xiao, Yige Yuan, Mingxiao Li, Zhengyu Chen, and Vasant G Honavar. 2025b. On a connection between imitation learning and RLHF. In *International Conference on Learning Representations*.

Teng Xiao, Yige Yuan, Huaisheng Zhu, Mingxiao Li, and Vasant G Honavar. 2024b. Cal-dpo: Calibrated direct preference optimization for language model alignment. In *Advances in Neural Information Processing Systems*.

An Yang, Baosong Yang, Beichen Zhang, and et al. 2024a. Qwen2.5 technical report.

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*.

Yuqing Yang, Yan Ma, and Pengfei Liu. 2024c. Weak-to-strong reasoning. *arXiv preprint arXiv:2407.13647*.

Ruimeng Ye, Yang Xiao, and Bo Hui. 2024. Weak-to-strong generalization beyond accuracy: a pilot study in safety, toxicity, and legal reasoning. *arXiv preprint arXiv:2410.12621*.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. 2025. Limo: Less is more for reasoning. *arXiv preprint arXiv:2502.03387*.

Edward Yeo, Yuxuan Tong, Morry Niu, and et al. 2025. Demystifying long chain-of-thought reasoning in llms. *arXiv preprint arXiv:2502.03373*.

Yige Yuan, Teng Xiao, Li Yunfan, Xu Bingbing, Shuchang Tao, Yunqi Qiu, Huawei Shen, and Xueqi Cheng. Inference-time alignment in continuous space. In *ICLR 2025 Workshop on Bidirectional Human-AI Alignment*.

Yige Yuan, Bingbing Xu, Hexiang Tan, Fei Sun, Teng Xiao, Wei Li, Huawei Shen, and Xueqi Cheng. 2024. Fact-level confidence calibration and self-correction. *arXiv preprint arXiv:2411.13343*.

Yang Yue, Zhiqi Chen, Rui Lu, Zhao, and et al. 2025. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *arXiv preprint arXiv:2504.13837*.

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*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Zhanke Zhou, Rong Tao, Jianing Zhu, Yiwen Luo, Zengmao Wang, and Bo Han. 2024. Can language models perform robust reasoning in chain-of-thought prompting with noisy rationales? In *Advances in Neural Information Processing Systems*.

A Appendix Summary

- **Experimental Details** (Appendix B):
  - Datasets (Appendix B.1)
  - Evaluation metrics (Appendix B.2)
  - Models (Appendix B.3)
  - Implementation (Appendix B.4)
  - Compute resources (Appendix B.5)
- **Additional Experiments** (Appendix C):
  - Inference Scaling (Appendix C.1)
  - Additional Results for Takeaway 1 (Appendix C.2)
  - Additional Results for Takeaway 2 (Appendix C.3)
  - Additional Results for Takeaway 4 (Appendix C.4)
- **Limitations and Broader Impacts** (Appendix D)
- **More Related Works** (Appendix F).
- **Insights and Future directions** (Appendix E)
- **Prompt Details** (Appendix G).
- **Case Studies** (Appendix H).

B Experimental Details

B.1 The Details of Datasets

MATH (Lightman et al., 2023) contains 500 high-school math competition problems, which are of higher difficulty and complexity, requiring more in-depth mathematical reasoning ability.

OlympiadBench (He et al., 2024) is a bilingual, multimodal scientific benchmark at the Olympiad level, comprising 8,476 problems from advanced mathematics and physics competitions, including the Chinese college entrance examination. Each problem is accompanied by expert-level annotations that provide step-by-step reasoning. The standard test benchmark contains 675 problems.

Minerva (Lewkowycz et al., 2022b) is a dataset of undergraduate-level questions in science and mathematics from MIT's OpenCourseWare.

AMC is a 2023 middle school math competition covering arithmetic, algebra, geometry, and more. It consists of 40 questions.

GPQA (Rein et al., 2024) is a challenging dataset of 448 multiple-choice questions written by domain experts in biology, physics, and chemistry. We use GPQA Diamond, the highest quality subset consisting of 198 questions, including only questions that both experts answer correctly and the majority of non-experts answer incorrectly.

B.2 The Details of Evaluation Metrics

Pass@k Given a total of N problems, for each problem i, we sample k responses. Let C_i be the number of correct responses among the k samples for problem i, and let n_i be the total number of possible distinct completions considered for that problem. Then the Passk accuracy is defined as:

$$\text{Pass@k} = \frac{1}{N}\sum_{i=1}^{N}\left[1 - \frac{\binom{n_i - C_i}{k}}{\binom{n_i}{k}}\right] \qquad (6)$$

B.3 The Details of Models

Qwen-2.5 (Yang et al., 2024b) is the next-generation open-source large language model series developed by the Qwen team. It includes models ranging from 0.5B to 72B parameters, trained on high-quality multilingual corpora. Qwen2.5 models demonstrate strong performance across a wide range of benchmarks, such as MMLU, GSM8K, and HumanEval, and are particularly effective in Chinese language understanding and generation. We utilize Qwen2.5-Math-7B, Qwen2.5-14B, and Qwen2.5-32B.

SimpleRL-Zoo (Zeng et al., 2025) is a model family fine-tuned using the zero RL training algorithm on MATH training datasets, spanning a diverse range of model series and sizes, including Mistral-7B, Mistral-24B, LLaMA3-8B, DeepSeek-Math-7B, Qwen2.5-0.5B/1.5B/7B/14B/32B, and Qwen2.5-Math-7B.

B.4 The Details of Implementation

We conduct model training using LLaMA-Factory (Zheng et al., 2024), applying full-parameter fine-tuning to all student models, ranging from 7B to 32B. The hyperparameters used are summarized in Table 3. Although the global batch size is fixed at 128 for all models, the per-device batch size, gradient accumulation steps, and number of GPUs vary across model scales. Specifically, for the 7B model, we use 4 GPUs with a per-device batch size of 2 and a gradient accumulation step of 16 (4GPU * 2PBS * 16GA); for the 14B model, we use 4 GPUs with a per-device batch size of 1 and a gradient accumulation step of 32 (4GPU * 1PBS * 32GA); and for the 32B model, we adopt 8 GPUs with a per-device batch size of 1 and a gradient accumulation step of 16 (8GPU * 1PBS * 16GA).

Table 3: Hyperparameters used for full-parameter fine-tuning.

Hyper-parameter	Value
Learning Rate	1×10^{-5}
Number of Epochs	10
Global Batch Size	128
Optimizer	`Adamw`
Learning Rate Scheduler	`cosine`
Max Sequence Length	4096

B.5 The Details of Compute Resources

All the training experiments in this paper were conducted on 4 × NVIDIA A100 (80G) GPUs.

C Additional Experiments

C.1 Inference-time Scaling

This section examines the impact of our method on inference-time behavior (Yuan et al., 2024), with a particular focus on response length. We investigate whether W2SR leads to an increase in generation length during inference, similar to trends observed with reinforcement learning. As shown in Table 4, using Qwen2.5-Math-7B, Qwen2.5-14B and Qwen2.5-32B as student and Qwen2.5-1.5B-Reasoner as teacher on AMC, the generation length increases steadily throughout training, indicative of longer CoT and suggesting the emergence of inference-time scaling effects.

C.2 Additional Results for Takeaway 1

This section presents supplementary results supporting Takeaway 1: weak supervision can incentivize reasoning in stronger models. Specifically, we show that reasoning trajectories derived from weaker teacher models can effectively encourage stronger student models to engage in reasoning, achieving performance comparable to or exceeding that of more costly reinforcement learning approaches. Figure 6 presents five additional configurations, each illustrating a distinct teacher–student pairing. Across all settings, student models demonstrate substantial gains when supervised by weaker reasoners. Notably, the performance improvements achieved by our proposed method (W2SR) and its enhanced variant (W2SR-P) consistently surpass those of baseline standalone students, and in some cases, even outperform students trained with expensive reinforcement learning.

C.3 Additional Results for Takeaway 2

This section provides supplementary results supporting Takeaway 2: reasoning capability is more important than model size or overall accuracy. We observe that a teacher's ability to perform explicit reasoning plays a more critical role than its parameter count or average task accuracy in fostering reasoning skills in student models. Figure 7 presents additional configurations where Qwen2.5-14B serves as the student model. Across these settings, students consistently benefit more from teachers with explicit reasoning abilities, even when those teachers are smaller or less accurate, compared to larger teachers lacking such abilities.

C.4 Additional Results for Takeaway 4

This section provides supplementary results supporting Takeaway 4, which suggests that weaker teachers offer a more favorable efficiency–performance trade-off. As shown in Figure 8, weaker teachers can achieve performance comparable to reinforcement learning-based methods while substantially reducing training costs. This demonstrates their practicality in resource-constrained settings.

D Limitations and Broader Impacts

This work focuses on mathematical reasoning, and the effectiveness of W2SR in broader domains such as commonsense reasoning, scientific QA, or legal analysis remains to be verified. The approach assumes that weak teacher models can produce structured reasoning traces, which may not hold in low-resource or complex domains. Additionally, while imperfect reasoning trajectories are often helpful, unfiltered or low-quality supervision may introduce noise and reduce robustness. Current method also relies on SFT and may benefit from more adaptive training strategies in future work.

W2SR provides a practical and efficient approach to improving reasoning abilities in language models without relying on expensive data collection or RL. This can enhance access to powerful models in academic and low-resource settings. However, enabling stronger reasoning capabilities may also increase the risk of models producing convincing but flawed outputs, especially when trained on imperfect supervision. Applications in high-risk domains should be accompanied by safeguards such

Table 4: Average Response Length.

Teacher	Method	Qwen2.5-Math-7B	Qwen2.5-14B	Qwen2.5-32B
Student only	Vanilla	882.78	643.90	600.20
Student only	GRPO	1122.27	1003.05	1120.56
Qwen2.5-1.5B(R)	W2SR	1545.92	1559.47	1373.55

as rigorous evaluation, interpretability tools, and human oversight to mitigate potential harms.

E Insights and Future Directions

Our work shows that structured but imperfect reasoning traces from significantly weaker models can effectively incentivize reasoning in stronger LLMs, rivaling reinforcement learning at a fraction of the cost. Notably, reasoning structure, rather than teacher size or accuracy, plays a pivotal role. Incorrect traces remain pedagogically valuable, and increasing teacher scale yields diminishing returns.

Future work includes adaptive filtering of reasoning trajectories to prioritize useful supervision, leveraging ensembles of weak teachers to provide richer signals, extending W2SR to multi-modal and tool-augmented settings, and developing theoretical frameworks to understand when weak-to-strong succeeds. These directions highlight W2SR as a scalable and practical approach for eliciting strong reasoning capabilities in large language models.

F More Related Works

F.1 Chain-of-Thought Distillation

Chain-of-thought (CoT) distillation has emerged as an effective technique for transferring reasoning abilities from large language models to smaller ones. Early work, such as Symbolic CoT Distillation (SCoTD)(Li et al., 2023), showed that small models can benefit from CoT supervision generated by significantly larger teachers. Subsequent studies introduced methods like Keypoint-based Progressive CoT Distillation (KPOD)(Feng et al., 2024), which incorporates token-level weighting and progressive learning to improve distillation. Other approaches, including CODI (Shen et al., 2025) and DLCoT (Luo et al., 2025), aim to compress CoT into continuous representations or deconstruct long reasoning trajectories for more efficient learning.

These methods typically rely on high-quality CoT traces from strong teacher models. In contrast, our work investigates the potential of leveraging structurally coherent but imperfect CoT traces from significantly weaker teachers to elicit strong reasoning capabilities in student models.

F.2 Imperfect or Noisy Supervision

Learning from imperfect or noisy supervision is a longstanding challenge in machine learning. In the context of LLMs, recent studies have examined the impact of noisy rationales on reasoning performance. For instance, Zhou et al. (Zhou et al., 2024) proposed contrastive denoising methods to enhance robustness against noisy CoT prompts. Guo et al. (Guo and Yang, 2024b) introduced reliability-aware alignment techniques to improve weak-to-strong generalization by assessing the trustworthiness of weak supervision signals.

Our work complements these efforts by demonstrating that even structurally flawed CoT traces from weaker models can effectively supervise stronger student models, highlighting the underexplored potential of leveraging imperfect yet structurally informative reasoning traces.

G Prompt Details

Following (Zeng et al., 2025), for models with weaker instruction-following capabilities, such as Qwen-2.5-0.5B/1.5B, we adopt simpler prompts in previous work (Chern et al., 2023), which require only step-by-step reasoning. In contrast, for models with stronger instruction-following abilities, we employ more complex prompts as proposed by (Yang et al., 2024a), where final answers must be explicitly placed within boxes.

H Case Study

As shown in Table 7, the weak teacher produces a structurally valid but numerically incorrect solution. The W2SR student retains this structured reasoning while correcting the computational errors, ultimately arriving at the correct answer. Its reasoning depth increases significantly, from 307 tokens (base student) to 984 tokens, approaching the length and quality of the RL-trained student (1038 tokens). These findings suggest that W2SR effectively leverages flawed yet structured supervision to elicit strong reasoning capabilities.

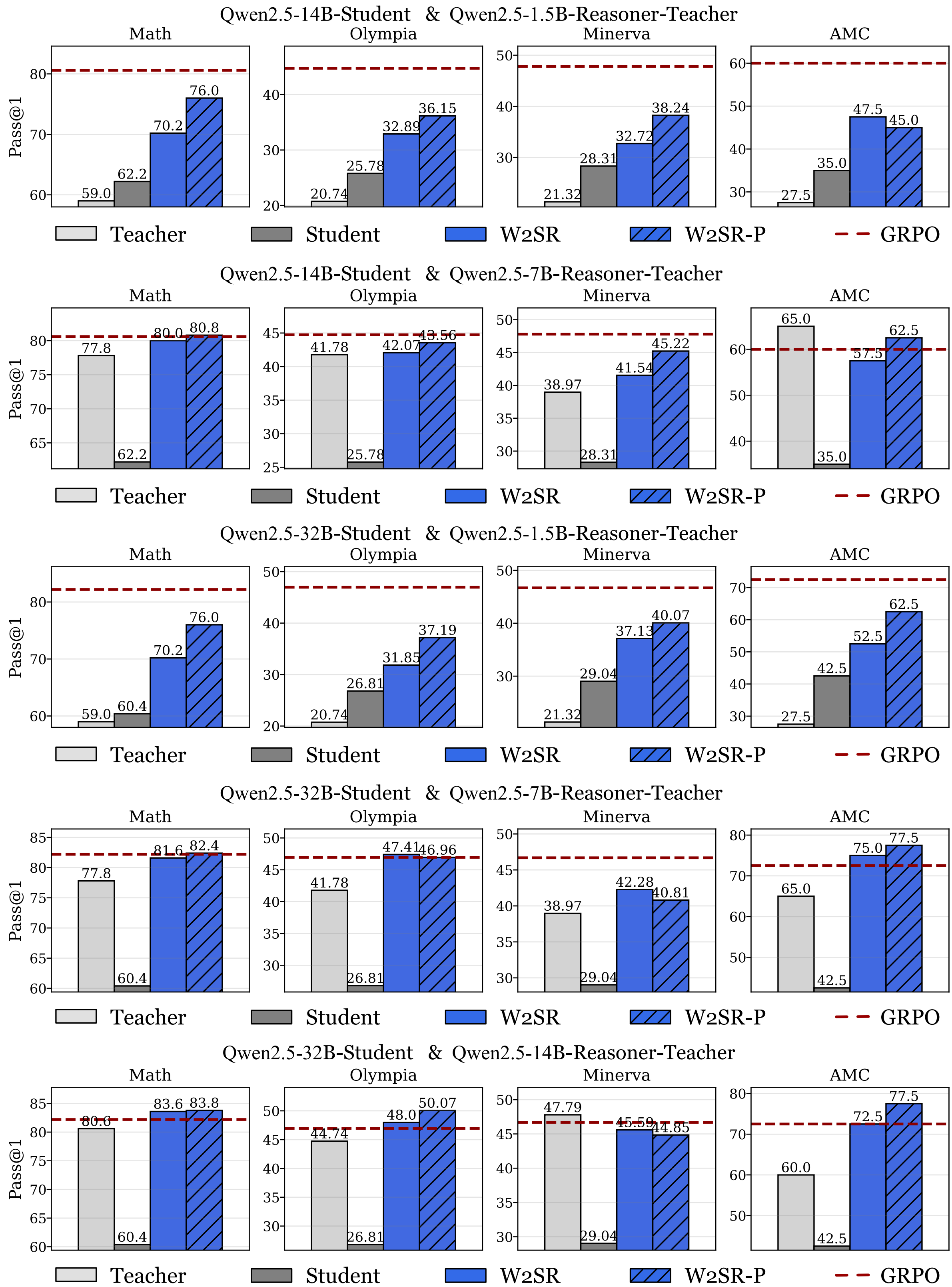

Figure 6: From top to bottom: Using (1) `Qwen2.5-14B` as the strong student and `Qwen2.5-1.5B-Reasoner` as the weak teacher. (2) `Qwen2.5-14B` as the strong student and `Qwen2.5-7B-Reasoner` as the weak teacher. (3) `Qwen2.5-32B` as the strong student and `Qwen2.5-1.5B-Reasoner` as the weak teacher. (4) `Qwen2.5-32B` as the strong student and `Qwen2.5-7B-Reasoner` as the weak teacher. (5) `Qwen2.5-32B` as strong student and `Qwen2.5-14B-Reasoner` as weak teacher. Four bars represent: teacher's standalone performance, student's standalone performance, student performance after `W2SR`, student performance after `W2SR-P`.

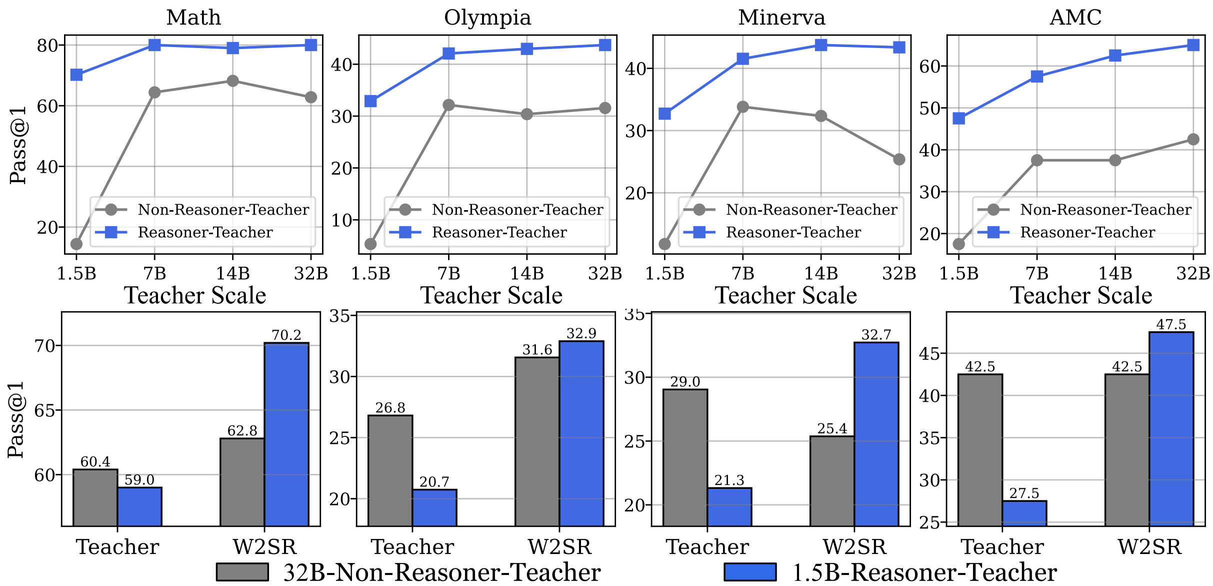

Figure 7: Comparison of Weak-to-Strong Reasoning (`W2SR`) between Reasoner and Non-Reasoner Teachers using `Qwen2.5-14B` as the student. **Upper**: The x-axis represents teacher model scale, with two lines indicating the `W2SR` student performance trained with Reasoner vs. Non-Reasoner teachers of corresponding scales. **Lower**: first two bars show initial performance of `Qwen2.5-1.5B-Reasoner` and `Qwen2.5-32B-Non-Reasoner` teacher, last two bars show the resulting `W2SR` student performance.

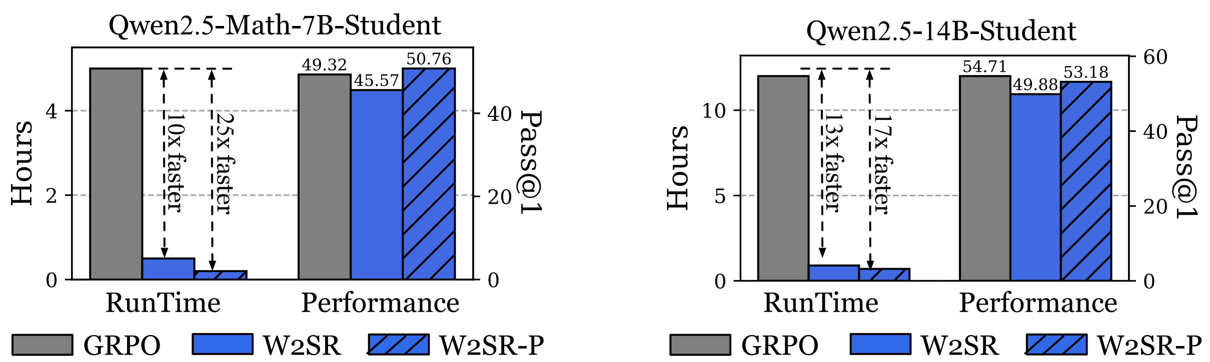

Figure 8: Comparison of efficiency and performance among GRPO, `W2SR`, and `W2SR-P`. Left: `Qwen2.5-Math-7B` as the student and `Qwen2.5-1.5B-Reasoner` as the teacher. Right: `Qwen2.5-14B` as the student and `Qwen2.5-7B-Reasoner` as the teacher. For each, the left three bars represent the time cost of GRPO, `W2SR`, and `W2SR-P` respectively, the right three bars show the corresponding performance outcomes.

```
Question:
{input}
Answer:
Let's think step by step.
```

Table 5: Simple prompts template.

```
<|im_start|>system
You are a helpful assistant.<|im_end|>
<|im_start|>user
{input}
Please reason step by step, and put your final answer within \\boxed{}.<|im_end|>
<|im_start|>assistant
{output}
```

Table 6: Complex prompts template.

Table 7: Case comparison of cognitive behaviors: weak teacher vs. base, RL, and W2SR students.

Question	Let K be the number of sequences A_1, A_2, ..., A_n such that n is a positive integer less than or equal to 10, each A_i is a subset of {1, 2, 3, ..., 10}, and A_{i-1} is a subset of A_i for each i between 2 and n, inclusive. For example, {}, {5, 7}, {2, 5, 7}, {2, 5, 7}, {2, 5, 6, 7, 9} is one such sequence, with n = 5.What is the remainder when K is divided by 10?
Weak Teacher	To determine the number of sequences (A_1, A_2, ..., A_n) where n is a positive integer less than or equal to 10, each A_i is a subset of {1, 2, 3, dots, 10}, and A_{i-1} is a subset of A_i for each i between 2 and n, inclusive, we need to consider the properties of such sequences. Each element in the set {1, 2, 3, dots, 10} can either be in A_i or not, and the choice for each element is independent of the others. Moreover, the condition $A_{i-1} \subseteq A_i$ means that if an element is in A_i, it must also be in A_{i-1} for all i. This problem can be translated into a combinatorial one. For each of the 10 elements, we need to decide in how many ways we can include or exclude it in the subsets A_1, A_2, ..., A_n such that the subset relationship is maintained. This is equivalent to choosing a non-decreasing sequence of subsets, which can be thought of as choosing a subset of {1, 2, ..., 10} and then deciding how many times each element appears in the sequence. For each element, there are n+1 choices (it can appear 0 times, 1 time, 2 times, ..., up to n times). Since there are 10 elements and each element's inclusion can be independently chosen, the total number of such sequences is given by: $K = (n+1)^{10}$ We need to find the value of K for n ranging from 1 to 10 and then find the remainder when the sum of these values is divided by 10. However, since n can be any integer from 1 to 10, we need to sum $n+1)^{10}$ for n from 1 to 10: $K = \sum_{n=1}^{10}(n+1)^{10}$ This simplifies to: $K = 2^{10} + 3^{10} + 4^{10} + \cdots + 11^{10}$ We need to find the remainder of this sum when divided by 10. We can do this by finding the remainder of each term when divided by 10 and then summing these remainders. $2^{10} \equiv 24 \equiv 4 \pmod{10}$ $3^{10} \equiv 59049 \equiv 9 \pmod{10}$ $4^{10} \equiv (4^2)^5 \equiv 16^5 \equiv 6^5 \equiv 7776 \equiv 6 \pmod{10}$ $5^{10} \equiv 0 \pmod{10}$ $6^{10} \equiv 6 \pmod{10}$ $7^{10} \equiv (7^2)^5 \equiv 49^5 \equiv 9^5 \equiv 59049 \equiv 9 \pmod{10}$ $8^{10} \equiv (8^2)^5 \equiv 64^5 \equiv 4^5 \equiv 1024 \equiv 4 \pmod{10}$ $9^{10} \equiv (9^2)^5 \equiv 81^5 \equiv 1^5 \equiv 1 \pmod{10}$ $10^{10} \equiv 0 \pmod{10}$ $11^{10} \equiv 1^{10} \equiv 1 \pmod{10}$ Summing these remainders: 4 + 9 + 6 + 0 + 6 + 9 + 4 + 1 + 0 + 1 = 40 The remainder when 40 is divided by 10 is: $40 \equiv 0 \pmod{10}$ Thus, the remainder when K is divided by 10 is: $\boxed{0}$

Base Student

To solve this problem, we need to count the number of sequences of subsets that satisfy the given conditions. Each subset A_i must be a subset of A_{i+1}, and each subset must be a subset of {1, 2, 3, dots, 10}. This means that we can think of each element in {1, 2, 3, dots, 10} as having three possible states: it can be in none of the subsets, it can be in some of the subsets but not in all of them, or it can be in all of the subsets. For each element, there are 3 choices. Since there are 10 elements, there are 3^{10} possible sequences of subsets. The remainder when K is divided by 10 is the same as the remainder when 3^{10} is divided by 10. We can calculate 3^{10} using Python and then find the remainder when it is divided by 10. Let's do this calculation.

```python
# Calculate 3^10
K = 3**10
# Find the remainder when K is divided by 10
remainder = K % 10
print(remainder)
```

```output
9
```

The remainder when K is divided by 10 is $\boxed{9}$.

RL Student

To find the number of sequences $A_1, A_2, \ldots, A_n$ such that n is a positive integer less than or equal to 10, each A_i is a subset of {1, 2, 3, ..., 10}, and A_{i-1} is a subset of A_i for each i between 2 and n , inclusive, we need to count the number of ways to choose such sequences for each possible value of n.

For a fixed n , we need to count the number of ways to choose a chain of subsets $A_1 \subseteq A_2 \subseteq cdots \subseteq A_n \subseteq$ {1, 2, ..., 10}. Each element in {1, 2, $dots$, 10} can independently choose to be in none of the sets, in exactly one of the sets, in exactly two of the sets, and so on, up to being in all n sets. Therefore, each element has n+1 choices (it can be in 0, 1, 2, ..., or n of the sets).

Since there are 10 elements, and each element has n+1 choices, the total number of such sequences for a fixed n is $(n+1)^{10}$.

We need to sum this over all possible values of n from 1 to 10:

$K = \sum_{n=1}^{10} (n+1)^{10}$

We are asked to find the remainder when K is divided by 10. Therefore, we need to compute:

$K \mod 10 = \sum_{n=1}^{10} (n+1)^{10} \mod 10$

We will compute each term $(n+1)^{10} \mod 10$ for n from 1 to 10:

$2^{10} = 1024 \equiv 4 \mod 10,$

$3^{10} = (3^2)^5 = 9^5 \equiv (-1)^5 = -1 \equiv 9 \mod 10,$

$4^{10} = (4^2)^5 = 16^5 \equiv 6^5 \equiv 6^2 \cdot 6^2 \cdot 6 \equiv 6 \cdot 6 \cdot 6 = 36 \cdot 6 \equiv 6 \cdot 6 = 36 \equiv 6 \mod 10,$

$5^{10} = (5^2)^5 = 25^5 \equiv 5^5 \equiv 5 \mod 10,$

$6^{10} = (6^2)^5 = 36^5 \equiv 6^5 \equiv 6 \mod 10,$

$7^{10} = (7^2)^5 = 49^5 \equiv (-1)^5 = -1 \equiv 9 \mod 10,$

$8^{10} = (8^2)^5 = 64^5 \equiv 4^5 \equiv 4^2 \cdot 4^2 \cdot 4 \equiv 6 \cdot 6 \cdot 4 = 36 \cdot 4 \equiv 6 \cdot 4 = 24 \equiv 4 \mod 10,$

$9^{10} = (9^2)^5 = 81^5 \equiv 1^5 = 1 \mod 10,$

$10^{10} = 10^5 cdot 10^5 \equiv 0^5 \cdot 0^5 = 0 \mod 10,$

$11^{10} = (11^2)^5 = 121^5 \equiv 1^5 = 1 \mod 10.$

Summing these results:

4 + 9 + 6 + 5 + 6 + 9 + 4 + 1 + 0 + 1 = 45

The remainder when 45 is divided by 10 is:

$45 \mod 10 = 5$

Therefore, the remainder when K is divided by 10 is $\boxed{5}$.

W2SR Student

To solve for the number of sequences $A_1, A_2, dots, A_n$ where each A_i is a subset of {1, 2, 3, dots, 10} and $A_{i-1} \subseteq A_i$ for each i between 2 and n, we need to consider the following:

1. **Understanding the Sequence**: Each sequence starts with the empty set A_1 = {} and ends with some subset A_n of {1, 2, 3, dots, 10}. The key property is that each set in the sequence is a subset of the next set.
2. **Counting the Sequences**: For each element in {1, 2, 3, dots, 10}, we need to decide at which step it first appears in the sequence. If an element appears at step k, it must be in all subsequent sets $A_k, A_{k+1}, \ldots, A_n$.
3. **Binary Decision for Each Element**: Each of the 10 elements can first appear at any step from 1 to n. Therefore, for each element, there are n choices (it can first appear at any of the n steps or not appear at all, which is the 0th step if we consider the empty set).
4. **Total Number of Sequences**: Since there are 10 elements and each element has n choices, the total number of sequences is n^{10}.
5. **Summing Over All Possible n**: n can range from 1 to 10. Therefore, we need to sum n^{10} for n from 1 to 10:
$K = sum_{n=1}^{10} n^{10}$
6. **Calculating the Sum**: We need to compute the sum of the 10th powers of the first 10 positive integers:
$K = 1^{10} + 2^{10} + 3^{10} + \cdots + 10^{10}$
7. **Finding the Remainder When K is Divided by 10**: We only need the last digit of K. Therefore, we compute the last digit of each term n^{10} for n from 1 to 10:
$1^{10} = 1 \quad \text{(last digit is 1)}$
$2^{10} = 1024 \quad \text{(last digit is 4)}$
$3^{10} = 59049 \quad \text{(last digit is 9)}$
$4^{10} = 1048576 \quad \text{(last digit is 6)}$
$5^{10} = 9765625 \quad \text{(last digit is 5)}$
$6^{10} = 60466176 \quad \text{(last digit is 6)}$
$7^{10} = 282475249 \quad \text{(last digit is 9)}$
$8^{10} = 1073741824 \quad \text{(last digit is 4)}$
$9^{10} = 3486784401 \quad \text{(last digit is 1)}$
$10^{10} = 10000000000 \quad \text{(last digit is 0)}$
8. **Summing the Last Digits**:
1 + 4 + 9 + 6 + 5 + 6 + 9 + 4 + 1 + 0 = 45
The last digit of 45 is 5.
Therefore, the remainder when K is divided by 10 is ($\boxed{5}$).