Weak Critics Make Strong Learners: On-Policy Critique Distillation for Scalable Oversight

Can Jin [*1] **Jiakang Li** [*1] **Rui Wu** [1] **Eddy Z. Zhang** [†1] **Dimitris N. Metaxas** [†1]

Abstract

As large language models become stronger, weak supervisors may fail to provide reliable labels, preferences, or final judgments for complex outputs, limiting both weak-to-strong generalization and scalable oversight. We study a more tractable form of weak supervision: using a weak model as a critic rather than as a labeler or judge. Instead of solving the task or selecting the correct answer, the weak critic only needs to provide a non-misleading revision direction that helps the strong model better use its own knowledge. We call this setting *weak-critic strong oversight*. We first show that weak critiques can improve frozen strong models at inference time, and that critique quality is key to this improvement. We then propose progressive on-policy critique distillation (OPCD), which filters high-quality critiques and distills critic-guided behavior into the strong model through adaptive self-teacher signals. Experiments on reasoning and alignment benchmarks show that our method improves strong models over training epochs, suggesting an effective path for scalable oversight with weak supervision.

1. Introduction

Modern large language models (LLMs) are commonly aligned with human supervision, such as task demonstrations, preference labels, reward models, and reinforcement learning from human feedback (Christiano et al., 2017; Ouyang et al., 2022; Bai et al., 2022a). These methods work well when the supervisor can reliably judge the model output. However, as models become stronger, they may produce answers, plans, proofs, or code that are difficult for humans or weaker models to fully verify. This creates a central challenge for alignment: how can a weak supervisor guide a stronger model when the final task is too hard for the supervisor to solve or judge?

Two related research directions study this challenge. Weak-to-strong generalization asks whether supervision from a weak model can elicit useful behavior from a stronger pretrained model (Burns et al., 2023). Scalable oversight asks how a weak human or model can provide reliable supervision for a stronger system, often through assistance, interaction, or debate (Amodei et al., 2016; Irving et al., 2018; Bowman et al., 2022; Khan et al., 2024). Although these directions differ in their goals, many existing methods place a similar burden on the weak supervisor: the weak model must provide direct labels, soft logits, preference signals, or final judgments over complete answers. This can be too demanding when the task is beyond the weak supervisor's full ability. In such cases, the supervision signal can be noisy, incomplete, or systematically wrong, which limits both weak-to-strong generalization and scalable oversight.

In this paper, we study a different form of weak supervision: using the weak model as a *critic* rather than as a labeler or judge. A weak critic does not need to solve the task, provide the correct answer, identify every error, or give a detailed revision plan. It can be useful by giving a general but correct revision direction, such as suggesting that the reasoning is incomplete, a condition is missing, a boundary case should be checked, or the response should be safer. This form of feedback is often easier than labeling or judging complete answers, and is close to common human–AI interaction: users may not know the full solution, but they can still give feedback that helps a stronger model revise. As long as the critique is not misleading, it can help the strong model better use its own knowledge without requiring the weak supervisor to provide full supervision. We call this setting *weak-critic strong oversight*.

We first test this idea at inference time. Given a question, the strong model produces an initial answer, the weak model critiques it, and the strong model then revises its answer conditioned on the question, the initial answer, and the critique. This directly evaluates whether weak critiques can improve a frozen strong model. Our results show that they can, even when the critique only gives a general revision

[*]Equal contribution , [†] Equal Advising. [1]Department of Computer Science, Rutgers University, Piscataway, United States. Correspondence to: Can Jin <can.jin@rutgers.edu>.

The 3rd *AI for Math Workshop* at the 43rd International Conference on Machine Learning (ICML), Seoul, South Korea, 2026.
Copyright 2026 by the author(s).

direction rather than a detailed error analysis. This supports the core hypothesis of *weak-critic strong oversight*: weak supervisors may not need to solve or judge the full task to provide useful oversight. We also find that critique quality is central. Helpful critiques improve performance, while misleading critiques can hurt performance, even compared with using no critique. This motivates filtering useful critiques before using them for training.

To internalize the inference-time improvement, we propose a progressive **o**n-**p**olicy **c**ritique **d**istillation method (OPCD). In each epoch, the current strong model generates on-policy answers, and the weak model critiques these answers. We then use an outcome- and rubric-based quality metric to keep only useful critiques. For each kept example, the critic-conditioned strong model serves as a self-teacher, using the critique as guidance to provide dense token-level signals. The student is the same strong model without access to the critique, trained by on-policy distillation. After each update, the strong model produces new answers with new error patterns, and the weak critic provides fresh critiques for the updated model. This process distills the useful critic-guided behavior observed at inference time while keeping the supervision adaptive to the current strong model.

Our experiments show that *weak-critic strong oversight* improves strong-model performance in both inference-time and training-time settings. Compared with standard weak-to-strong methods that directly distill weak-model responses or logits, our method does not force the weak model to provide full supervision. Compared with ground-truth supervised finetuning, our method studies a more realistic oversight setting where reliable labels may be unavailable or too hard for weak supervisors to provide. Across reasoning and alignment benchmarks, progressive on-policy critique distillation improves the strong model over training epochs, showing that critique-based supervision is an effective path for scalable oversight and weak-to-strong generalization.

Our main contributions are:

★ **Critique-based weak supervision.** We identify critiquing as a more tractable form of weak supervision than labeling or judging, and propose *weak-critic strong oversight* for scalable oversight and weak-to-strong generalization.

★ **Inference-time validation.** We show that weak critiques can improve strong-model performance at inference time, even when they provide only general revision directions, and find that critique quality is key to reliable improvement.

★ **Progressive critique distillation.** We introduce OPCD, a progressive on-policy critique distillation strategy that filters high-quality critiques and uses them as adaptive weak feedback to train the strong model.

★ **Strong results.** Across multiple benchmarks, OPCD progressively improves strong-model performance, showing that critique-based supervision can effectively use weak oversight for stronger models.

2. Related Works

Scalable oversight and weak-to-strong generalization. Scalable oversight aims to develop methods to supervise AI systems on tasks that are difficult for humans (Amodei et al., 2016; Bowman et al., 2022). The primary focus has been on designing human-AI collaboration protocols that help humans evaluate AI outputs more accurately, for example through debate and consultancy (Irving et al., 2018; Kenton et al., 2024), and on lowering the cognitive burden of evaluation through critique-assisted review and prover-verifier games (Saunders et al., 2022; McAleese et al., 2024; Kirchner et al., 2024). In contrast, weak-to-strong generalization (W2S) (Burns et al., 2023) explores a complementary direction that designs learning algorithms which let a strong pretrained model generalize correctly from weak supervision as if it had been trained on higher-quality labels. A growing body of work strengthens this elicitation through iterative label refinement, easy-to-hard reward transfer, weak-LLM preference labeling, internal-coherence elicitation, and self-consistency filtering for reasoning (Ye et al., 2025; Sun et al., 2024; Tao & Li, 2024; Wen et al., 2025; Yang et al., 2024; Jin et al., 2024), with inference-time variants contrasting weak and strong distributions or guiding decoding with weak step-level scores (Li et al., 2023; Ding et al., 2025). Theoretical analyses bound the W2S gain by the strong model's misfit on weak labels and characterize pseudolabel correction under an expansion condition (Charikar et al., 2024; Lang et al., 2024), while Yang et al. (2025b) document a failure mode in which strong models pass weak supervision on prompts the supervisor knows but remain misaligned where it does not. Our setting follows the W2S protocol of Burns et al. (2023) but uses weak critiques in place of weak labels and targets generative reasoning rather than classification.

LLM Alignment. Aligning LLMs with human preferences is most commonly done via reinforcement learning from human feedback (Christiano et al., 2017; Ouyang et al., 2022; Bai et al., 2022a), which trains a reward model on preference comparisons and optimizes the policy with PPO (Schulman et al., 2017); direct preference methods bypass the reward model and train on preferences end-to-end (Rafailov et al., 2023). Because human supervision is expensive and difficult to scale, a separate line replaces or supplements it with model-generated signals: AI feedback judged against written principles (Bai et al., 2022b; Lee et al., 2024; Jin et al., 2026), self-judgment loops in which the policy itself plays the role of judge (Yuan et al.,

2024; Wu et al., 2024), and dedicated critic models that surface mistakes the policy or human annotators might miss (McAleese et al., 2024; Ankner et al., 2024). These methods take the supervisor's signal as a training target. We instead treat it as a critique and use only those critiques that demonstrably improve the policy's answer.

LLM Reasoning. Eliciting reasoning from LLMs has been driven by chain-of-thought prompting and its inference-time variants such as self-consistency and tree-search decoding (Wei et al., 2022; Wang et al., 2023; Yao et al., 2023). Self-improvement methods finetune the policy on its own successful traces (Zelikman et al., 2024) or train process verifiers on step-level correctness to densify the reward on hard problems (Lightman et al., 2024; Wang et al., 2024; Jin et al., 2025). A complementary line teaches the model to revise its outputs, either at inference time through verbal feedback (Madaan et al., 2023; Shinn et al., 2023) or during training through multi-turn reinforcement learning (Kumar et al., 2024; Qu et al., 2024). Recent large-scale efforts show that pure outcome-based RL on a strong base model can elicit long chain-of-thought reasoning at scale (DeepSeek-AI, 2025), while online distillation has shifted toward student-side losses that match the teacher under the student's own distribution (Gu et al., 2023). These methods either require ground-truth verification or assume a critic at least as strong as the policy.

3. Preliminary Inference-Time Investigation

Whether a weak model can provide useful critiques to improve a stronger model at inference time is the foundation of OPCD framework. In this section, we conduct a preliminary investigation to answer two questions:

(i) Can weak-model critique improve strong-model performance beyond sampling more responses, and does this effect generalize across reasoning and alignment tasks as well as across thinking and non-thinking models?
(ii) Does the quality of the critique affect the final accuracy?

Experimental Setting. We evaluate the critique-and-refine paradigm on both non-thinking and thinking models. For non-thinking models, we use Phi-4-mini-instruct-3.8B (Abdin et al., 2024) as the weak model and Phi-4-14B (Abdin et al., 2024) as the strong model. The experiments are conducted on GPQA Diamond (Rein et al., 2023) for reasoning task and IFEval (Zhou et al., 2023) for instruction-following alignment task. For thinking models, we use Qwen3-1.7B (Yang et al., 2025a) as the weak model and Qwen3-8B (Yang et al., 2025a) as the strong model. Both models are evaluated with thinking mode enabled on AIME 2024 (Art of Problem Solving, 2024a;b) and AIME 2025 (Art of Problem Solving, 2025a;b), containing 60 problems in total. This setting tests whether the usefulness of weak-model critiques also scales to thinking models.

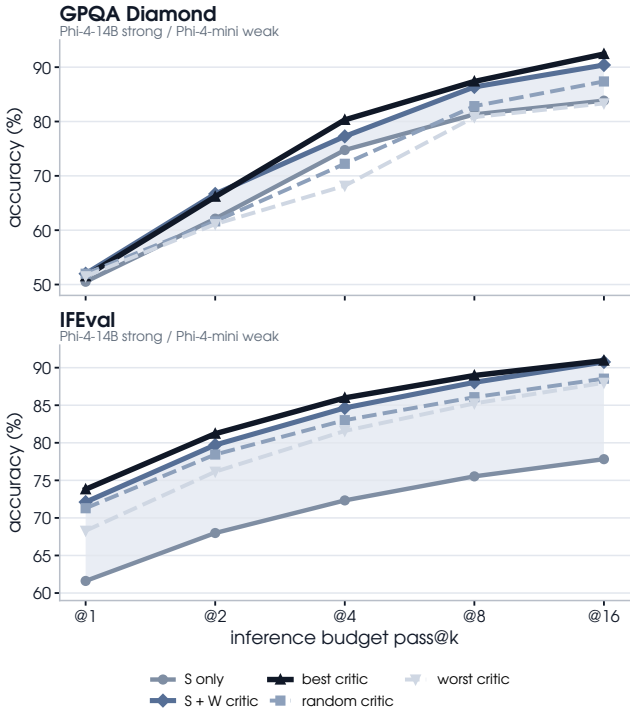

Figure 1. Pass@k performance reported as accuracy percentage under different inference budgets on GPQA Diamond and IFEval. The strong model is Phi-4-14B and the weak model is Phi-4-mini-instruct. The shaded region highlights the performance gap between the $Sonly$ baseline and the $S+Wcritic+refine$ setting.

For each problem, the critique-and-refine pipeline consists of three stages and we denote it as *S+W critic+refine* for easier annotation. First, the strong model generates an initial answer. Then, the weak model receives the original problem and the strong model's initial answer, and produces a critique. Finally, the strong model refines its answer based on the problem, the initial answer, and the weak-model critique. This process is repeated for up to 16 independent chains as reported in Table 1. Formally, for each chain i, the process is written as:

$$a_i = S(x), \quad c_i = W(x, a_i), \quad r_i = S(x, a_i, c_i), \quad (1)$$

where a_i, c_i, and r_i denote the initial answer, weak-model critique, and refined answer of chain i, respectively. For an inference budget of $k \in \{1, 2, 4, 8, 16\}$ chains, we report pass@k, where the prediction is counted as correct if any of the first k final answers is correct.

Baselines. We compare the proposed critique-and-refine pipeline with several inference-time baselines. The basic baseline is *S only*, where the strong model independently generates multiple answers without receiving any critique.

Benchmark	Model Pair	Method	Inference Budget				
			pass@1	pass@2	pass@4	pass@8	pass@16
GPQA Diamond	Phi-4-14B (S) Phi-4-mini-instruct (W)	*S only*	50.51	62.12	74.75	81.31	83.84
		S+W critic+refine	**51.99**	**66.67**	77.27	86.36	90.40
		S+best critic+refine	51.52	66.16	**80.30**	**87.37**	**92.42**
		S+random critic+refine	**51.99**	61.62	72.22	82.83	87.37
		S+worst critic+refine	51.52	61.11	68.18	80.81	83.33
IFEval	Phi-4-14B (S) Phi-4-mini-instruct (W)	*S only*	61.61	67.99	72.33	75.53	77.82
		S+W critic+refine	72.11	79.71	84.64	88.05	90.76
		S+best critic+refine	**73.81**	**81.21**	**85.98**	**88.96**	**90.94**
		S+random critic+refine	71.29	78.44	83.01	86.07	88.54
		S+worst critic+refine	68.29	76.15	81.58	85.24	87.99
AIME24+25	Qwen3-8B (S) Qwen3-1.7B (W)	*S only*	71.67	81.67	81.67	83.33	86.67
		W only	40.00	51.67	61.67	66.67	73.33
		S+W critic+refine	**75.00**	**81.67**	**81.67**	**85.00**	**90.00**

Table 1. Inference-time scaling results on reasoning and alignment benchmarks. All results are reported as accuracy percentages. S and W denote the strong and weak models, respectively. Best results within each benchmark are highlighted in bold.

For the non-thinking model Phi-4 experiments, we further introduce three baselines to analyze how critique quality affects final performance. To study the critic quality effect alone, these baselines are constructed from cached outputs for the complete k chains in *S+ W critic + refine*. For each inference budget k, we select one critique from the first k chains and pair it with each of the k initial answers before refinement. In *S + best critic + refine*, we select a critique from a chain where the initial answer is incorrect but the refined answer becomes correct, if such a chain exists. This setting approximates the effect of a high-quality critique. In *S + random critic + refine*, we uniformly sample one critique from the first k chains, which measures the effect of critique reuse without correctness information. In *S + worst critic + refine*, we select a critique from a chain where the initial answer is correct but the refined answer becomes incorrect, if such a chain exists. For the thinking-model Qwen experiments, we also report *W only* where the weak model independently generates multiple answers as *S only*. We do not include other baselines as the Phi-4 experiments in the thinking model setting, because this experiment is intended as a focused test of whether weak critique can scale thinking-mode generation rather than a complete critique-quality analysis.

Weak Model Critique Framework Improves Strong Model Performance Across Tasks and Model Types. As shown in Table 1 and Figure 1, weak-model critique improves strong-model performance at inference time across both reasoning and alignment tasks. On GPQA Diamond, *S only* achieves 50.51 pass@1 and 83.84 pass@16, while *S+W critic+refine* improves the results to 51.99 pass@1 and 90.40 pass@16. Figure 1 further shows that this improvement is maintained as the inference budget increases, indicating that weak critique provides useful feedback signals beyond sampling more responses. A similar trend appears on IFEval. As reported in Table 1, the *S only* baseline achieves 61.61 pass@1 and 77.82 pass@16. With weak-model critique and refinement, the performance increases to 72.11 pass@1 and 90.76 pass@16. The scaling trend in Figure 1 also shows a consistent gap between *S only* and *S+W critic+refine* across different inference budgets where the strong model can consistently outperform *S only*. These observations suggest that weak model critique framework improves strong model performance across reasoning and alignment tasks.

We further examine whether weak-model critique remains useful under thinking-mode generation. As shown in Table 1, on AIME 2024 and AIME 2025, the weak model itself is substantially weaker than the strong model, achieving 40.00 pass@1 compared with 71.67 pass@1 for the strong model. Nevertheless, when used as a critic, the weak model improves the strong model from 71.67 to 75.00 at pass@1 and from 86.67 to 90.00 at pass@16. These results suggest that weak model critique improves strong model performance across thinking and non-thinking models.

High-Quality Critiques Improve Performance and Generalize Across different Initial Answers. The baselines for Phi-4 in Table 1 show that critique quality matters in improving performance. Notably, on GPQA Diamond *S+best critic+refine* achieves best performance on pass@4, pass@8, and pass@16 with 80.30, 87.37 and 92.42 respectively. While we can also observe that *S+random critic+refine* is consistently weaker than *S+W critic+refine* at larger budgets and *S+worst critic+refine* performs the worst among other critique-quality baselines, with 68.18 pass@4, 80.81 pass@8, and 83.33 pass@16, eventually falling slightly below *S only* at pass@16. Also, a similar pattern appears on

IFEval, S+*best critic*+*refine* achieves the strongest results across all inference budgets, reaching 73.81 pass@1, 85.98 pass@4, and 90.94 pass@16, and is followed by S+*random critic*+*refine* and S+*worst critic*+*refine*. These results indicate that quality of critique matters, high-quality critique can better improve the performance in the framework but low-quality critique can weaken the effect of improvement across different tasks.

Since the selected critiques in baselines are not necessarily paired with the current initial answer, this result also suggests that a high-quality critique can generalize across different initial answers. In other words, useful critiques may provide transferable guidance, such as identifying common reasoning errors, highlighting missing constraints, or suggesting a more reliable solving direction, rather than only correcting one specific response.

4. Method

4.1. Problem Definition and Notation

The overall framework is shown in Figure 2. We consider a weak-to-strong oversight setting with a weak model π_{w} and a strong model π_θ. Given an input question or instruction $x \in \mathcal{X}$, our goal is to improve the strong model using weak supervision, without requiring the weak model to provide the correct answer, a preference label, or a final judgment. Instead, the weak model acts as a critic that gives feedback on the strong model's own answers.

At training epoch e, let π_{θ_e} denote the current strong model before the update. For each input x, we sample a group of G on-policy answers

$$y_i \sim \pi_{\theta_e}(\cdot \mid x), \qquad i = 1, \ldots, G,$$

where $y_i = (y_{i,1}, \ldots, y_{i,T_i})$ and $y_{i,<t}$ denotes the prefix before token $y_{i,t}$. The weak model then critiques each answer:

$$f_i \sim \pi_{\mathrm{w}}(\cdot \mid x, y_i).$$

Each critique f_i provides a revision direction for the corresponding answer y_i. It does not need to give a full solution or a detailed correction; it only needs to offer useful and non-misleading guidance that can help the strong model better use its own knowledge.

Our key idea is to use the critic-conditioned strong model as a self-teacher. For each pair (y_i, f_i), the teacher is the frozen current strong model conditioned on the weak critique,

$$\pi_\theta(\cdot \mid x, f_i, y_{i,<t}),$$

while the student is the trainable strong model without access to the critique,

$$\pi_\theta(\cdot \mid x, y_{i,<t}).$$

The training objective distills the critic-guided behavior of the teacher into the student, so that the strong model can internalize the benefit of weak critiques and improve without requiring critiques at test time.

4.2. Critique Generation and Filtration

Given the on-policy answers $\{y_i\}_{i=1}^{G}$ from the current strong model π_{θ_e}, the weak critic produces a critique f_i for each answer. Since weak critiques can be helpful or misleading, we filter them before using them for training. The goal of filtration is to keep only critiques that provide useful guidance for the strong model.

For each triple (x, y_i, f_i), we first ask the current strong model to generate a refined answer conditioned on the question, the original answer, and the critique:

$$\hat{y}_i \sim \pi_{\theta_e}(\cdot \mid x, y_i, f_i).$$

We then apply an outcome-based check to test whether the refined answer is correct:

$$r_{\mathrm{out}}(x, \hat{y}_i) = \mathbb{1}\{\hat{y}_i \text{ is correct for } x\}.$$

This check directly measures whether the critique helps the strong model revise toward a correct answer. However, outcome correctness alone may keep critiques that are vague, irrelevant, or not actually responsible for the improvement. Therefore, we also use a rubric-based check:

$$r_{\mathrm{rub}}(x, y_i, f_i) = \mathbb{1}\Big\{f_i \text{ is relevant and useful as revision guidance}\Big\}.$$

A critique is kept only if it passes both checks:

$$h_i = r_{\mathrm{out}}(x, \hat{y}_i) \cdot r_{\mathrm{rub}}(x, y_i, f_i).$$

The filtered training set for epoch e is then

$$\mathcal{S}_e = \{(x, y_i, f_i) : h_i = 1\}.$$

Thus, the strong model is trained only on questions and on-policy answers with high-quality critiques. This filtration step is important because OPCD aims to distill useful critic-guided behavior, rather than imitate weak feedback blindly.

4.3. Progressive On-Policy Critique Distillation

After obtaining the filtered set $\mathcal{S}_e$, we distill the critic-guided behavior into the strong model. For each selected triple (x, y, f), we use the critic-conditioned strong model defined in Subsection 4.1 as the self-teacher, and train the same strong model without access to the critique as the student. Both are evaluated along the same on-policy answer trajectory y, so the critique only affects the teacher distribution while the student learns to internalize this guidance.

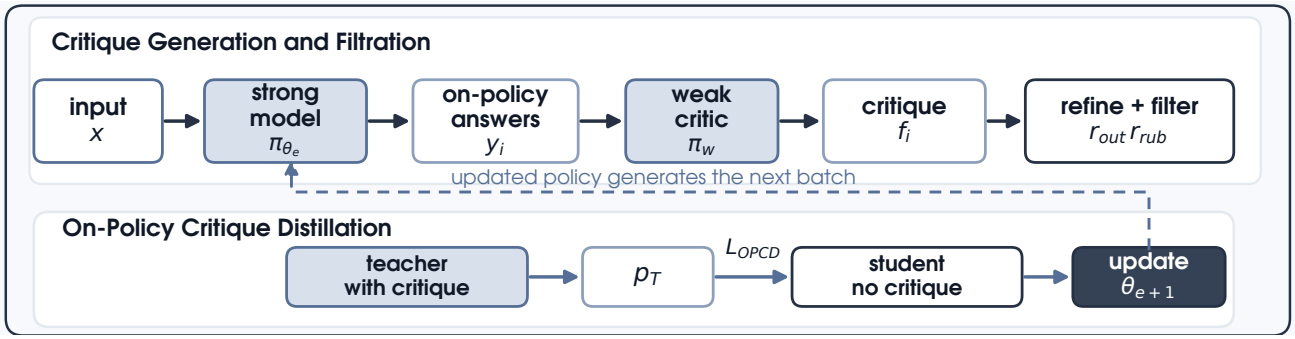

Figure 2. Overview of the OPCD pipeline.

We optimize the student by minimizing the token-level KL divergence:

$$\mathcal{L}_{\text{OPCD}}(\theta) = \frac{1}{|\mathcal{S}_e|} \sum_{(x,y,f)\in\mathcal{S}_e} \sum_{t=1}^{|y|} \mathrm{KL}\Big(\pi_\theta(\cdot \mid x, y_{<t}) \,\Big\|\, \mathrm{stopgrad}[\pi_\theta(\cdot \mid x, f, y_{<t})]\Big).$$

where the KL divergence is defined as

$$\mathrm{KL}(p\|q) = \sum_{v\in\mathcal{V}_{\mathrm{KL}}} p(v) \log \frac{p(v)}{q(v)},$$

with $\mathcal{V}_{\mathrm{KL}}$ denoting the token set used for the KL computation, which can be the full vocabulary or a selected subset of tokens. This objective transfers the dense token-level guidance induced by the weak critique into the strong model, so that the updated model can benefit from critic-guided reasoning without requiring critiques at test time.

The training process is progressive and on-policy. Before each epoch, the current strong model generates new answers, the weak model critiques these answers, and the filtration step constructs a new high-quality set $\mathcal{S}_e$. The model is then updated with $\mathcal{L}_{\text{OPCD}}$. After the update, the next model $\pi_{\theta_{e+1}}$ produces new answers with different error patterns, leading to new critiques and a new filtered training set. In this way, the policy, critiques, and training data are updated together throughout training. This makes OPCD adaptive to the current strong model rather than relying on a fixed offline set of weak critiques. We summarize the full procedure in Algorithm 1.

Algorithm 1 OPCD

Require: Strong model π_{θ_0}, weak critic π_{w}, training set $\mathcal{X}$, epochs E, group size G

$\mathcal{S}_{-1} \leftarrow \emptyset$
for epoch $e = 0, 1, \ldots, E-1$ **do**
  $\mathcal{S}_e \leftarrow \mathcal{S}_{e-1}$
  for each input $x \in \mathcal{X}$ **do**
    Sample on-policy answers $\{y_i \sim \pi_{\theta_e}(\cdot \mid x)\}_{i=1}^{G}$
    Generate weak critiques $\{f_i \sim \pi_{\mathrm{w}}(\cdot \mid x, y_i)\}_{i=1}^{G}$
    for $i = 1, \ldots, G$ **do**
      Refine: $\hat{y}_i \sim \pi_{\theta_e}(\cdot \mid x, y_i, f_i)$
      if $r_{\mathrm{out}}(x, \hat{y}_i) \cdot r_{\mathrm{rub}}(x, y_i, f_i) = 1$ **then**
        $\mathcal{S}_e \leftarrow \mathcal{S}_e \cup \{(x, y_i, f_i)\}$
      end if
    end for
  end for
  Update $\theta_e \to \theta_{e+1}$ by minimizing $\mathcal{L}_{\text{OPCD}}(\theta)$ on $\mathcal{S}_e$
end for
return π_{θ_E}

5. OPCD Training-Time Experiments

We evaluate our method in two weak-to-strong scenarios: reasoning and alignment. The reasoning and alignment scenarios study whether a strong model can learn from weak critique on a reasoning or alignment task and can improve performance on the in-domain validation set.

Experimental Setup. In the reasoning scenario, we use Qwen3-4B-base (Yang et al., 2025a) as the strong model and Qwen3-1.7B-base (Yang et al., 2025a) as the weak model. In the alignment scenario, we use Phi-4-14B (Abdin et al., 2024) as the strong model and Phi-4-mini-instruct (Abdin et al., 2024) as the weak model. To follow the weak to strong generalization (Burns et al., 2023) setting, we supervise fine-tune (SFT) (Radford et al., 2018) the weak model in the alignment scenario and use it to provide critiques in the OPCD training.

Training and Evaluation Data. For reasoning, we use the GPQA main split (Rein et al., 2023) excluding the GPQA-Diamond subset as the source corpus. This produces a non-Diamond GPQA subset with 250 multiple-choice science questions. We split this subset into two disjoint parts: 129 questions are used for training, and the remaining 121 questions are held out for in-domain validation. The training split is used to construct weak supervision from Qwen3-1.7B-

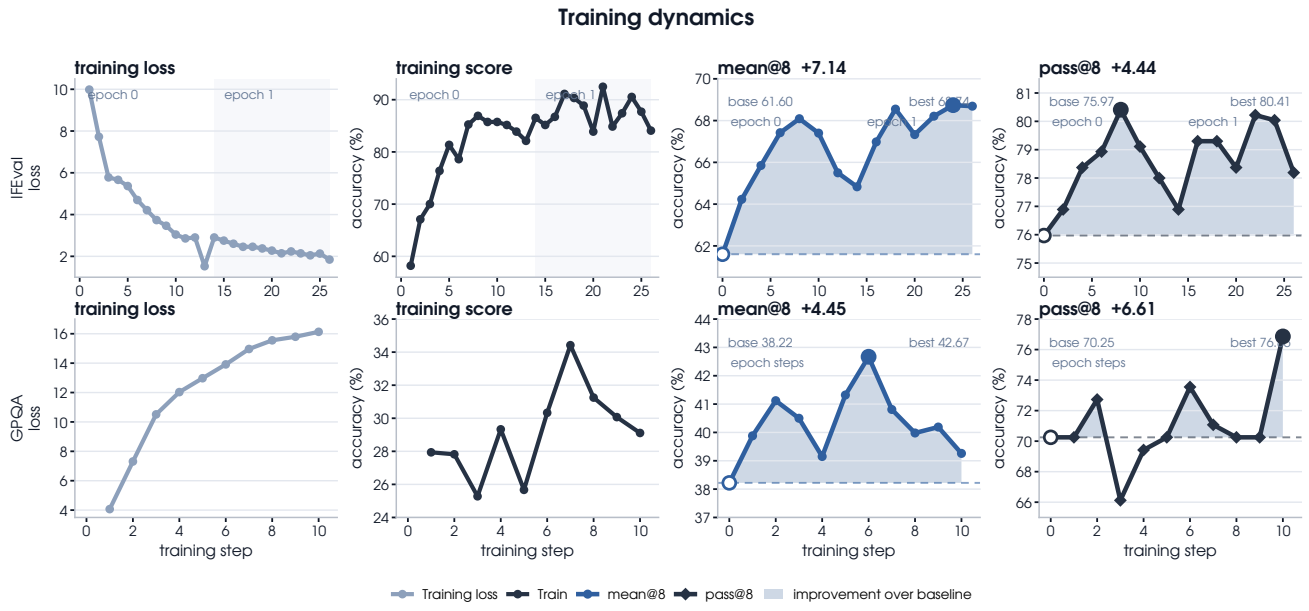

Figure 3. Training dynamics of OPCD in alignment and reasoning scenarios. The left column shows results in the alignment scenario, while the right column shows results in the reasoning scenario. For each scenario, we report the training loss, training score, and validation performance over global training steps. The highlighted regions in Validation indicate the improvement over initial validation on the test set before training.

Base, which is then used to train Qwen3-4B-Base under our weak-to-strong distillation framework. For alignment, we use the Magpie alignment corpus (Xu et al., 2024) which is IFEval-style (Zhou et al., 2023) dataset as the source training corpus and sample 5,000 instruction-response examples. We randomly split these examples into two disjoint subsets of equal size. The first 2,500 examples are used to fine-tune Phi-4-mini-instruct into a weak alignment supervisor, and the remaining 2,500 examples are used to train Phi-4-14B with critiques generated by this weak model. We use the test split of IFEval (Zhou et al., 2023) as the in-domain validation set during training.

Hyperparameters. For the alignment experiment, the weak critic is initialized from a Phi-4-mini-instruct checkpoint supervised fine-tuned on the SFT training split with learning rate 1×10^{-6}, and we use the step-40 checkpoint for later OPCD training. For both alignment and reasoning experiments, the rollout number in on-policy answer generation stage is $n = 8$, the training batch size is 128, the PPO mini-batch size is 32, and the PPO micro-batch size per GPU is 1. The maximum prompt length is 4096. The maximum response length is 8192 for alignment and 6000 for reasoning. We optimize the actor with AdamW using weight decay 0.01 and no learning-rate warmup. The actor learning rate is 1×10^{-6} for alignment experiment and 5×10^{-7} for reasoning experiment. At weak critique generation and refinement stage, we sample initial answer rollouts with temperature 1.0, top-$k = 20$, and top-$p = 0.95$. We use greedy decoding for critique generation, top-$p = 1.0$ and top-$k = -1$. We sample refined answer generation with temperature 0.6, top-$k = 20$, and top-$p = 0.95$. Validation uses 8 sampled responses per prompt with temperature 0.6, top-$k = 20$, and top-$p = 0.95$. For the alignment setting, we evaluate the trained strong model on IFEval during training and report mean@8 and pass@8. For the reasoning setting, we evaluate on the 121 held-out non-Diamond GPQA problems and report the same sampling-based metrics. Here, mean@8 measures the average accuracy over eight sampled responses, while pass@8 measures whether at least one of the eight responses is correct.

OPCD Improves Training-Time Performance. As shown in Table 2 and Table 3, OPCD improves strong-model performance in both alignment and reasoning training. On IFEval, the initial model achieves 61.60 mean@8 and 75.97 pass@8, while OPCD improves mean@8 to 68.74 and pass@8 to 80.41 during training. On GPQA, the initial model achieves 38.22 mean@8 and 70.25 pass@8, while OPCD improves mean@8 to 42.67 and pass@8 to 76.86. These results show that weak-model critique can provide useful training-time supervision for the strong model across both instruction-following alignment and scientific reasoning tasks. This observation is also consistent with our preliminary experiments, where weak-model critiques improve strong-model performance without updating model weights.

Table 2. Training dynamics of OPCD in the alignment experiment. Accuracy-related metrics, including Train, mean@8, and pass@8, are reported as percentages.

Step	Epoch	Training Score	Train Score	mean@8	pass@8
0	–	–	–	61.60	75.97
1	0	9.9845	58.20	–	–
2	0	7.7260	67.09	64.23	76.89
3	0	5.7806	70.02	–	–
4	0	5.6632	76.37	65.85	78.37
5	0	5.3658	81.35	–	–
6	0	4.6986	78.61	67.42	78.93
7	0	4.2116	85.25	–	–
8	0	3.7364	86.91	68.09	**80.41**
9	0	3.4646	85.74	–	–
10	0	3.0429	85.74	67.40	79.11
11	0	2.8580	85.16	–	–
12	0	2.9025	83.89	65.50	78.00
13	0	1.5260	82.12	–	–
14	1	2.9028	86.52	64.83	76.89
15	1	2.7527	85.16	–	–
16	1	2.6034	86.72	66.98	79.30
17	1	2.4570	91.11	–	–
18	1	2.4585	90.33	68.55	79.30
19	1	2.3776	88.87	–	–
20	1	2.2747	83.89	67.33	78.37
21	1	2.1447	92.48	–	–
22	1	2.2345	84.86	68.21	80.22
23	1	2.1421	87.40	–	–
24	1	2.0481	90.53	**68.74**	80.04
25	1	2.1318	87.70	–	–
26	1	1.8460	84.08	68.69	78.19

Reasoning and Alignment Exhibit Different Training Dynamics. Although OPCD improves both settings, the training dynamics are substantially different. In the alignment experiment, the training loss (distillation loss) generally decreases from 9.9845 to around 2, and the Train score rises from 58.20 to over 90 at several steps, indicating relatively stable optimization under weak critique supervision. In contrast, in the reasoning experiment, the policy-gradient loss increases from 4.0672 to 16.1311, and the train score fluctuates between 25.28 and 34.42 rather than increasing smoothly. This suggests that reasoning tasks produce noisier and higher-variance weak-to-strong training signals than alignment tasks. A possible reason is that scientific reasoning requires precise multi-step correctness, so weak critiques may help the strong model find useful reasoning directions but do not always provide high-quality supervision for every rollout. Therefore, compared with alignment training, reasoning training benefits from OPCD but remains more sensitive to critique quality.

6. Conclusion

We introduce weak-critic strong oversight, a scalable oversight setting where a weak model guides a stronger model through critiques rather than labels, preferences, or final judgments. This reduces the burden on the weak supervisor, since it only needs to provide a useful and non-misleading revision direction instead of solving or judging the full task. Our inference-time experiments show that weak critiques can improve frozen strong models on both reasoning and alignment benchmarks, and that critique quality is central to reliable improvement. Motivated by this finding, we proposed On-Policy Critique Distillation (OPCD), which filters high-quality weak critiques and distills critic-guided behavior into the strong model using adaptive self-teacher signals. Training-time results on IFEval and GPQA show that OPCD can improve strong-model behavior over training steps, demonstrating that weak critiques can provide effective supervision even when direct weak labels or judgments are unreliable. These findings suggest that critique-based weak supervision is a promising path for weak-to-strong generalization and scalable oversight. Future work can further improve critique-quality estimation, study broader task domains, and combine weak critiques with other oversight mechanisms such as debate, process supervision, or verifier-assisted training.

Table 3. Training dynamics of OPCD in the reasoning experiment. Accuracy-related metrics, including Train, mean@8, and pass@8, are reported as percentages.

Epoch	Training Loss	Train Score	mean@8	pass@8
–	–	–	38.22	70.25
0	4.0672	27.94	39.88	70.25
1	7.3085	27.82	41.12	72.73
2	10.5132	25.28	40.50	66.12
3	12.0304	29.33	39.15	69.42
4	12.9736	25.67	41.32	70.25
5	13.9145	30.33	**42.67**	73.55
6	14.9623	34.42	40.81	71.07
7	15.5532	31.25	39.98	70.25
8	15.7956	30.07	40.19	70.25
9	16.1311	29.11	39.26	**76.86**

References

Abdin, M., Aneja, J., Behl, H., Bubeck, S., Eldan, R., Gunasekar, S., Harrison, M., Hewett, R. J., Javaheripi, M., Kauffmann, P., et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., and Mané, D. Concrete problems in AI safety. *arXiv preprint arXiv:1606.06565*, 2016.

Ankner, Z., Paul, M., Cui, B., Chang, J. D., and Ammanabrolu, P. Critique-out-Loud reward models. *arXiv preprint arXiv:2408.11791*, 2024.

Art of Problem Solving. 2024 aime i problems and solutions. https://artofproblemsolving.com/wiki/index.php/2024_AIME_I, 2024a. Accessed: 2026-05-24.

Art of Problem Solving. 2024 aime ii problems and so-

lutions. https://artofproblemsolving.com/wiki/index.php/2024_AIME_II, 2024b. Accessed: 2026-05-24.

Art of Problem Solving. 2025 aime i problems and solutions. https://artofproblemsolving.com/wiki/index.php/2025_AIME_I, 2025a. Accessed: 2026-05-24.

Art of Problem Solving. 2025 aime ii problems and solutions. https://artofproblemsolving.com/wiki/index.php/2025_AIME_II, 2025b. Accessed: 2026-05-24.

Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022a.

Bai, Y., Kadavath, S., Kundu, S., Askell, A., Kernion, J., Jones, A., Chen, A., Goldie, A., Mirhoseini, A., McKinnon, C., et al. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*, 2022b.

Bowman, S. R., Hyun, J., Perez, E., Chen, E., Pettit, C., Heiner, S., Lukošiūtė, K., Askell, A., Jones, A., Chen, A., et al. Measuring progress on scalable oversight for large language models. *arXiv preprint arXiv:2211.03540*, 2022.

Burns, C., Izmailov, P., Kirchner, J. H., Baker, B., Gao, L., Aschenbrenner, L., Chen, Y., Ecoffet, A., Joglekar, M., Leike, J., et al. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. *arXiv preprint arXiv:2312.09390*, 2023.

Charikar, M., Pabbaraju, C., and Shiragur, K. Quantifying the gain in weak-to-strong generalization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Ding, Z., Wang, Y., Xiao, T., Wang, H., Jiang, C., and Ding, N. W2S-AlignTree: Weak-to-strong inference-time alignment for large language models via monte carlo tree search. *arXiv preprint arXiv:2511.11518*, 2025.

Gu, Y., Dong, L., Wei, F., and Huang, M. Minillm: Knowledge distillation of large language models. *arXiv preprint arXiv:2306.08543*, 2023.

Irving, G., Christiano, P., and Amodei, D. AI safety via debate. *arXiv preprint arXiv:1805.00899*, 2018.

Jin, C., Che, T., Peng, H., Li, Y., Metaxas, D. N., and Pavone, M. Learning from teaching regularization: Generalizable correlations should be easy to imitate. *Advances in Neural Information Processing Systems*, 37:966–994, 2024.

Jin, C., Zhou, Y., Zhang, Q., Peng, H., Zhang, D., Dong, Z., Pavone, M., Han, L., Hong, Z.-W., Che, T., et al. Your reward function for rl is your best prm for search: Unifying rl and search-based tts. *arXiv preprint arXiv:2508.14313*, 2025.

Jin, C., Wu, R., Che, T., Zhang, Q., Peng, H., Zhao, J., Wang, Z., Wei, W., Han, L., Zhang, Z., et al. Reasoning over precedents alongside statutes: Case-augmented deliberative alignment for llm safety. *arXiv preprint arXiv:2601.08000*, 2026.

Kenton, Z., Siegel, N., Kramár, J., Brown-Cohen, J., Albanie, S., Bulian, J., Agarwal, R., Lindner, D., Tang, Y., Goodman, N., et al. On scalable oversight with weak llms judging strong llms. *Advances in Neural Information Processing Systems*, 37:75229–75276, 2024.

Khan, A., Hughes, J., Valentine, D., Ruis, L., Sachan, K., Radhakrishnan, A., Grefenstette, E., Bowman, S. R., Rocktäschel, T., and Perez, E. Debating with more persuasive llms leads to more truthful answers. *arXiv preprint arXiv:2402.06782*, 2024.

Kirchner, J. H., Chen, Y., Edwards, H., Leike, J., McAleese, N., and Burda, Y. Prover-verifier games improve legibility of LLM outputs. *arXiv preprint arXiv:2407.13692*, 2024.

Kumar, A., Zhuang, V., Agarwal, R., Su, Y., Co-Reyes, J. D., Singh, A., Baumli, K., Iqbal, S., Bishop, C., Roelofs, R., et al. Training language models to self-correct via reinforcement learning. *arXiv preprint arXiv:2409.12917*, 2024.

Lang, H., Sontag, D., and Vijayaraghavan, A. Theoretical analysis of weak-to-strong generalization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Lee, H., Phatale, S., Mansoor, H., Mesnard, T., Ferret, J., Lu, K., Bishop, C., Hall, E., Carbune, V., Rastogi, A., and Prakash, S. RLAIF vs. RLHF: Scaling reinforcement learning from human feedback with AI feedback. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

Li, X. L., Holtzman, A., Fried, D., Liang, P., Eisner, J., Hashimoto, T. B., Zettlemoyer, L., and Lewis, M. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 12286–12312, 2023.

Lightman, H., Kosaraju, V., Burda, Y., Edwards, H., Baker, B., Lee, T., Leike, J., Schulman, J., Sutskever, I., and Cobbe, K. Let's verify step by step. In *International Conference on Learning Representations (ICLR)*, 2024.

Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegreffe, S., Alon, U., Dziri, N., Prabhumoye, S., Yang, Y., et al. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

McAleese, N., Pokorny, R. M., Ceron Uribe, J. F., Nitishinskaya, E., Trebacz, M., and Leike, J. LLM critics help catch LLM bugs. *arXiv preprint arXiv:2407.00215*, 2024.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Qu, Y., Zhang, T., Garg, N., and Kumar, A. Recursive introspection: Teaching language model agents how to self-improve. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. Improving language understanding by generative pre-training. 2018.

Rafailov, R., Sharma, A., Mitchell, E., Manning, C. D., Ermon, S., and Finn, C. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Rein, D., Hou, B. L., Stickland, A. C., Petty, J., Pang, R. Y., Dirani, J., Michael, J., and Bowman, S. R. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

Saunders, W., Yeh, C., Wu, J., Bills, S., Ouyang, L., Ward, J., and Leike, J. Self-critiquing models for assisting human evaluators. *arXiv preprint arXiv:2206.05802*, 2022.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Shinn, N., Cassano, F., Berman, E., Gopinath, A., Narasimhan, K., and Yao, S. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Sun, Z., Yu, L., Shen, Y., Liu, W., Yang, Y., Welleck, S., and Gan, C. Easy-to-hard generalization: Scalable alignment beyond human supervision. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

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

Wang, P., Li, L., Shao, Z., Xu, R., Dai, D., Li, Y., Chen, D., Wu, Y., and Sui, Z. Math-Shepherd: Verify and reinforce LLMs step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., Chowdhery, A., and Zhou, D. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q. V., and Zhou, D. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Wen, J., Ankner, Z., Somani, A., Hase, P., Marks, S., Goldman-Wetzler, J., Petrini, L., Sleight, H., Burns, C., He, H., Feng, S., Perez, E., and Leike, J. Unsupervised elicitation of language models. *arXiv preprint arXiv:2506.10139*, 2025.

Wu, T., Yuan, W., Golovneva, O., Xu, J., Tian, Y., Jiao, J., Weston, J., and Sukhbaatar, S. Meta-rewarding language models: Self-improving alignment with LLM-as-a-Meta-Judge. *arXiv preprint arXiv:2407.19594*, 2024.

Xu, Z., Jiang, F., Niu, L., Deng, Y., Poovendran, R., Choi, Y., and Lin, B. Y. Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing, 2024. URL https://arxiv.org/abs/2406.08464.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Yang, W., Shen, S., Shen, G., Gong, W., Yao, Y., and Lin, Y. Super(ficial)-alignment: Strong models may deceive weak models in weak-to-strong generalization. In *International Conference on Learning Representations (ICLR)*, 2025b.

Yang, Y., Ma, Y., and Liu, P. Weak-to-strong reasoning. In *Findings of the Association for Computational Linguistics: EMNLP*, 2024.

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Ye, Y., Laidlaw, C., and Steinhardt, J. Iterative label refinement matters more than preference optimization under

weak supervision. In *International Conference on Learning Representations (ICLR)*, 2025.

Yuan, W., Pang, R. Y., Cho, K., Sukhbaatar, S., Xu, J., and Weston, J. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*, 2024.

Zelikman, E., Wu, Y., Mu, J., and Goodman, N. D. Star: Self-taught reasoner bootstrapping reasoning with reasoning. In *Proc. the 36th International Conference on Neural Information Processing Systems*, volume 1126, 2024.

Zhou, J., Lu, T., Mishra, S., Brahma, S., Basu, S., Luan, Y., Zhou, D., and Hou, L. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.