Self-Evolving Critique Abilities in Large Language Models

Zhengyang Tang[1,4*] **Ziniu Li**[1,2*] **Zhenyang Xiao**[1*] **Tian Ding**[2,3†] **Ruoyu Sun**[1,2,3]
Benyou Wang[1†] **Dayiheng Liu**[4†] **Fei Huang**[4] **Tianyu Liu**[4] **Bowen Yu**[4] **Junyang Lin**[4]

[1]The Chinese University of Hong Kong, Shenzhen
[2]Shenzhen Research Institute of Big Data
[3]Shenzhen International Center for Industrial and Applied Mathematics
[4]Qwen Team, Alibaba Inc.
`dingtian@sribd.cn, wangbenyou@cuhk.edu.cn, liudayiheng.ldyh@alibaba-inc.com`

Abstract

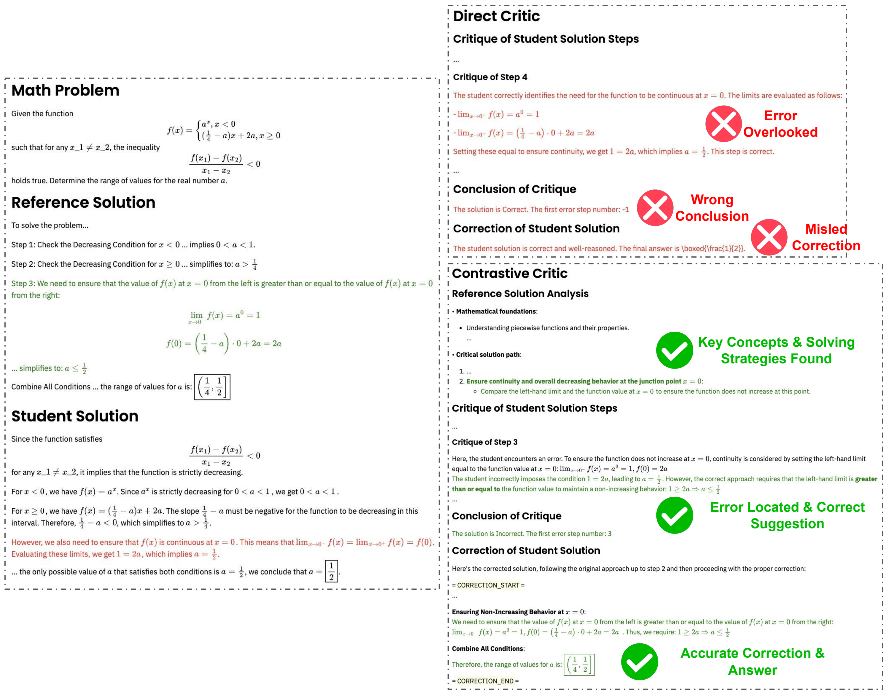
Despite their remarkable performance, Large Language Models (LLMs) face a critical challenge: providing feedback for tasks where human evaluation is difficult or where LLMs potentially outperform humans. In such scenarios, leveraging the *critique* ability of LLMs themselves—identifying and correcting flaws—shows considerable promise. This paper explores enhancing critique abilities of LLMs, noting that current approaches rely on human annotations or more powerful models, leaving the challenge of improving critique abilities *without* external supervision *unresolved*. We introduce SCRIT (Self-evolving CRITic), a framework that trains LLMs with self-generated data to evolve their critique abilities. To address the low quality of naively generated data, we propose a contrastive-critic approach that uses reference solutions during data synthesis to enhance the model's understanding of key concepts, and incorporates a self-validation scheme to ensure data quality. The final trained model operates without any reference solutions at inference time. Implemented with Qwen2.5-72B-Instruct, a leading LLM, SCRIT demonstrates consistent improvements across a wide range of benchmarks spanning both mathematical and scientific reasoning: achieving a 10.0% relative gain in critique-correction accuracy and a 19.0% relative improvement in error identification F1-score. Our analysis reveals that SCRIT's performance scales positively with data and model size and enables continuous improvement through multi-round iterations.

1 Introduction

Large Language Models (LLMs) (Achiam et al., 2023; Anthropic, 2024; Qwen-Team, 2024) represent significant milestones in the development of artificial intelligence. They rely on human supervision through methods such as Supervised Fine-Tuning (SFT) (Wei et al., 2021; Li et al., 2025) and Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022; Bai et al., 2022; Li et al., 2024). As a result, these models have evolved at an unprecedented pace, surpassing human capabilities in certain challenging domains. However, this framework encounters a fundamental challenge: how to provide effective and scalable feedback for LLMs in tasks that are not only difficult for humans to evaluate but where LLMs may outperform humans. This challenge, known as scalable oversight (Bowman et al., 2022), remains critical, yet progress in this area has been limited.

To address this challenge, leveraging LLMs for evaluation can help refine model outputs (Saunders et al., 2022; McAleese et al., 2024). Central to this approach is the *critique* ability—identifying and correcting flaws in responses. Accurate critique feedback enables LLMs to improve, advancing toward higher-order intelligence. Yet, studies show LLMs underperform in critique tasks (Zheng et al., 2024b; Yang et al., 2024; Tang et al., 2025). Thus, enhancing critique abilities is a key research problem, which this paper aims to tackle.

*Equal contribution.
†Corresponding author.



Math Problem

Given the function

$$f(x) = \begin{cases} a^x, x < 0 \\ (\frac{1}{4} - a)x + 2a, x \geq 0 \end{cases}$$

such that for any $x\_1 \neq x\_2$, the inequality

$$\frac{f(x_1) - f(x_2)}{x_1 - x_2} < 0$$

holds true. Determine the range of values for the real number a.

Reference Solution

To solve the problem...

Step 1: Check the Decreasing Condition for $x < 0$ ... implies $0 < a < 1$.

Step 2: Check the Decreasing Condition for $x \geq 0$ ... simplifies to: $a > \frac{1}{4}$

Step 3: We need to ensure that the value of $f(x)$ at $x = 0$ from the left is greater than or equal to the value of $f(x)$ at $x = 0$ from the right:

$$\lim_{x \to 0^-} f(x) = a^0 = 1$$

$$f(0) = \left(\frac{1}{4} - a\right) \cdot 0 + 2a = 2a$$

... simplifies to: $a \leq \frac{1}{2}$

Combine All Conditions ... the range of values for a is: $\boxed{\left(\frac{1}{4}, \frac{1}{2}\right]}$

Student Solution

Since the function satisfies

$$\frac{f(x_1) - f(x_2)}{x_1 - x_2} < 0$$

for any $x\_1 \neq x\_2$, it implies that the function is strictly decreasing.

For $x < 0$, we have $f(x) = a^x$. Since a^x is strictly decreasing for $0 < a < 1$, we get $0 < a < 1$.

For $x \geq 0$, we have $f(x) = (\frac{1}{4} - a)x + 2a$. The slope $\frac{1}{4} - a$ must be negative for the function to be decreasing in this interval. Therefore, $\frac{1}{4} - a < 0$, which simplifies to $a > \frac{1}{4}$.

However, we also need to ensure that $f(x)$ is continuous at $x = 0$. This means that $\lim_{x \to 0^-} f(x) = \lim_{x \to 0^+} f(x) = f(0)$. Evaluating these limits, we get $1 = 2a$, which implies $a = \frac{1}{2}$.

... the only possible value of a that satisfies both conditions is $a = \frac{1}{2}$, we conclude that $a = \boxed{\frac{1}{2}}$.

Direct Critic

Critique of Student Solution Steps

...

Critique of Step 4

The student correctly identifies the need for the function to be continuous at $x = 0$. The limits are evaluated as follows:

- $\lim_{x \to 0^-} f(x) = a^0 = 1$

- $\lim_{x \to 0^+} f(x) = (\frac{1}{4} - a) \cdot 0 + 2a = 2a$

Setting these equal to ensure continuity, we get $1 = 2a$, which implies $a = \frac{1}{2}$. This step is correct.

...

Conclusion of Critique

The solution is Correct. The first error step number: -1

Correction of Student Solution

The student solution is correct and well-reasoned. The final answer is \boxed{\frac{1}{2}}.

Contrastive Critic

Reference Solution Analysis

- **Mathematical foundations**:
  - Understanding piecewise functions and their properties.
    ...
- **Critical solution path**:
  1. ...
  2. **Ensure continuity and overall decreasing behavior at the junction point** $x = 0$:
     - Compare the left-hand limit and the function value at $x = 0$ to ensure the function does not increase at this point.

Critique of Student Solution Steps

...

Critique of Step 3

Here, the student encounters an error. To ensure the function does not increase at $x = 0$, continuity is considered by setting the left-hand limit equal to the function value at $x = 0$: $\lim_{x \to 0^-} f(x) = a^0 = 1, f(0) = 2a$
The student incorrectly imposes the condition $1 = 2a$, leading to $a = \frac{1}{2}$. However, the correct approach requires that the left-hand limit is **greater than or equal to** the function value to maintain a non-increasing behavior: $1 \geq 2a \Rightarrow a \leq \frac{1}{2}$
...

Conclusion of Critique

The solution is Incorrect. The first error step number: 3

Correction of Student Solution

Here's the corrected solution, following the original approach up to step 2 and then proceeding with the proper correction:

= CORRECTION_START =
...

Ensuring Non-Increasing Behavior at $x = 0$:
We need to ensure that the value of $f(x)$ at $x = 0$ from the left is greater than or equal to the value of $f(x)$ at $x = 0$ from the right:
$\lim_{x \to 0^-} f(x) = a^0 = 1, f(0) = (\frac{1}{4} - a) \cdot 0 + 2a = 2a$. Thus, we require: $1 \geq 2a \Rightarrow a \leq \frac{1}{2}$

Combine All Conditions:

Therefore, the range of values for a is: $\boxed{\left(\frac{1}{4}, \frac{1}{2}\right]}$

= CORRECTION_END =

Key Concepts & Solving Strategies Found
Error Located & Correct Suggestion
Accurate Correction & Answer

Figure 1: Direct critic (baseline) v.s. contrastive critic (ours). Left panel: input materials prepared for critique generation. Right panel: outputs from both approaches. The direct critic exhibits ”rubber-stamping” behavior, incorrectly validating flawed solutions and providing misled feedback. The contrastive critic, however, utilizes reference solutions to grasp key concepts and strategies, enabling accurate error identification and correction.

Current approaches to improving the critique abilities of LLMs rely on two sources of supervision: human annotations (Saunders et al., 2022; McAleese et al., 2024) and stronger LLMs that serve as human proxy (e.g., GPT-4 and o1-mini) (Lan et al., 2024; Zhang et al., 2024; Zheng et al., 2024b; Yang et al., 2024)). While these methods have shown promise, they face three fundamental limitations. First, the quality of generated critiques is inherently bounded by the capabilities of the supervisors. Second, the dependence on human annotations or API calls to stronger models introduces significant costs, limiting the scalability of these approaches. Most critically, these approaches fail to address a fundamental question in scalable oversight: how can we enhance the critique abilities of our most capable models when stronger supervisors are no longer available?

In this work, we introduce SCRIT (Self-evolving CRITic), a framework that enables LLMs to develop self-evolving critique abilities in domains with verifiable solutions. We focus on **mathematical and scientific reasoning** as ideal testbeds for this approach. A key insight is that problems in these domains typically have well-defined reference solutions and corresponding final answers. These resources, leveraged *only during the data synthesis phase*, guide the critique of a student's solution and help verify the quality of the generated critique.

Our framework has two key steps to generate high-quality critique data for self-training.

- First, we develop a contrastive critique technique, where the model is provided with a reference solution to analyze and critique a student's solution. This step is grounded in our first philosophy: by conditioning on a correct reference solution first, the LLM develops a comprehensive understanding of the relevant concepts and problem-solving strategies, allowing it to accurately identify and address errors in student solutions. Our evidence shows that without this reference point, the model tends to exhibit "rubber-stamping" behavior—uncritically approving incorrect solutions and offering misleading feedback (see Figure 1 for examples).

- Next, the LLM is tasked with self-validating the generated critique to improve the data quality. Specifically, the model checks whether the proposed corrections lead to valid solutions. This step is based on our second philosophy: critiques that result in internally consistent and correct correction are considered high-quality, which has also been widely adopted by recent works (Zheng et al., 2024b; Yang et al., 2024; Tang et al., 2025).

These two steps together enable the generation of high-quality critique data without human supervisions in writing good critiques for student solutions. Finally, we leverage the generated data to enhance the model's critique abilities through self-training. We clarify that while our framework requires reference solutions as input, it does not depend on ground truth critiques themselves, thus remaining within the scalable oversight paradigm.

We use Qwen2.5-72B-Instruct (Qwen-Team, 2024), a leading 70B model, to implement SCRIT. Our goal is to test whether our framework can further improve its performance. It is important to note that this is a non-trivial task, as Qwen2.5-72B-Instruct has already undergone extensive pre-training and post-training. Experiments show that SCRIT enables substantial improvements across different evaluation protocols as shown in Tables 1 and 2.

- On critic and correction tasks from (Tang et al., 2025), spanning 8 mathematical reasoning datasets across 3 scenarios, SCRIT consistently enhances the base Qwen2.5-72B-Instruct model: improving from 39.7% to 50.0% on deliberately incorrect solutions, from 57.7% to 62.1% on balanced solutions, and from 61.7% to 62.9% on the base model's self-generated solutions, representing a 10.0% relative gain in critique-correction accuracy on average.
- For error identification tasks on PRM800K (Lightman et al., 2023) and ProcessBench (Zheng et al., 2024a), two benchmarks with human-labeled error steps, SCRIT achieves consistent improvements across all datasets, raising the average F1 score from 37.8% to 45.0%, a 19.0% relative improvement.

In addition to these advancements, we provide a systematic analysis, which will be elaborated on in the main text. Our framework and methodology are detailed in the subsequent sections. Due to space constraints, the related work is discussed in Appendix A.

2 SCRIT: Self-Evolving Critic

2.1 Problem Formulation and Overview

Let $\mathcal{P}$ denote a set of problems from a structured domain (e.g., mathematics, science), where each problem $p \in \mathcal{P}$ is paired with an answer a_p. For each problem p, we collect a set of solutions $\mathcal{S}_p = \{s_1, s_2, ..., s_n\}$ from different models, where each solution s_i consists of:

- A sequence of reasoning steps $\mathbf{r}_i = [r_i^1, r_i^2, ..., r_i^{k_i}]$, where k_i is the number of steps
- A final answer a_{s_i}

A critique c is defined as a tuple $c = (\mathbf{e}, l, t)$, where:

- $\mathbf{e} = [e_1, e_2, ..., e_k]$ is a sequence of step-wise critiques, where each e_i corresponds to the analysis of step r^i
- $l = (y, j)$ is the conclusion, where $y \in \{0, 1\}$ indicates solution correctness and $j \in \{-1\} \cup \mathbb{N}$ denotes the first error step ($j = -1$ means no error)
- t is the correction, consisting of a sequence of corrected steps and a final answer a_t

Our objective is to learn a critique function $f_\theta : \mathcal{P} \times \mathcal{S} \to \mathcal{C}$ that maps a problem p and a solution s to an effective critique c, where θ denotes the parameter to learn.

To achieve this objective, we propose SCRIT (Self-evolving CRITic), a framework that systematically leverages the shared mathematical understanding across different solutions to enable truly self-evolving critique abilities. SCRIT operates through a complete self-evolving cycle: it takes a problem and solutions as input, generates critiques through analyzing reference solutions, validates their quality, and uses the validated critiques for self-training. This forms a complete self-evolving cycle without any external supervision.

2.2 Solution Collection

Dataset The first step in our framework is to collect a diverse set of solutions. We build our collection process on the NuminaMath dataset (LI et al., 2024), a large-scale mathematical problem dataset covering various topics from elementary mathematics to competition-level problems. To ensure data quality, we develop a robust pipeline to compute reliable ground truth answers (detailed in Appendix B), resulting in 452K validated problem-answer pairs.

Solution Generation Models To enhance the diversity of generated data, we gather solutions from seven models: deepseek-math-7b-rl (Shao et al., 2024), mathstral-7B-v0.1 (Mistral-AI, 2024a), Mistral-Large-Instruct-2411 (Mistral-AI, 2024b), DeepSeek-V2-Chat-0628 (DeepSeek-AI, 2024), Qwen2.5-Math-7B-Instruct (Qwen-Team, 2024), Qwen2.5-Math-1.5B-Instruct (Qwen-Team, 2024), and Qwen2-Math-1.5B-Instruct (Qwen-Team, 2024). It is important to note that the outputs from these models serve as inputs for the critic model, with no external supervision involved in the critic's learning process.

Data Filtering For each problem $p \in \mathcal{P}$, we classify its collected solutions into correct solutions $\mathcal{S}_p^+$ and incorrect solutions $\mathcal{S}_p^-$ based on answer correctness. A crucial filtering criterion in our framework is that each problem must have at least one correct solution and one incorrect solution to enable later contrastive critic. Formally, we only retain problems that satisfy: $\mathcal{P}_{valid} = \{p \in \mathcal{P} || \mathcal{S}_p^+| > 0 \land |\mathcal{S}_p^-| > 0\}$.

2.3 Self-Critic Generation

A key challenge in enabling effective critique generation is to ensure the model can identify and correct errors in complex mathematical reasoning, particularly when the problem difficulty approaches or exceeds the model's current capabilities. Our preliminary experiments reveal that the model often exhibits "rubber-stamping behavior" - blindly approving incorrect steps without genuine understanding of the mathematical concepts involved, as illustrated in Figures 1 and 5. This also aligns with findings in (Huang et al., 2023).

We initially explored two approaches from previous works: (1) **Direct Critic** (Zheng et al., 2024a), where a language model directly critiques a solution; and (2) **Bug-Injection Critic** (McAleese et al., 2024), a two-stage approach of first injecting errors into a correct solution and then ask the LLM to critic and correct it. However, both approaches showed limited effectiveness (detailed in Section 4.3).

To address these issues, we develop a new technique called **Contrastive Critic**. Our key insight stems from a fundamental property of mathematical reasoning: *while problems may have multiple valid solutions, they share the same underlying mathematical concepts and key solving strategies*. By explicitly providing a correct reference solution during critique generation, we enable the model to first understand these core mathematical concepts and solving strategies, then leverage this understanding to perform step-by-step critique of the target solution. This approach addresses the rubber-stamping issue by grounding the critique process in concrete mathematical understanding derived from correct references.

For each valid problem, we generate critiques using two solution pairing approaches:

- **Correct-Incorrect Pairs.** For each incorrect solution, we randomly select a correct reference solution and generate a critique by comparing the incorrect solution against the reference.
- **Correct-Correct Pairs.** For each correct solution, we randomly select a different correct solution as reference and generate a critique comparing the two.

Both pairing strategies promote diversity in the generated critiques, which we empirically validate for effectiveness in subsequent experiments. The self-critic function (prompt template in Appendix C) decomposes critique generation into four sequential stages. **Stage 1 (Reference Analysis):** Generate a reference analysis that captures key mathematical concepts, critical solution steps, and potential pitfalls. **Stage 2 (Step-wise Critique):** For each step in the solution, generate a critique by verifying mathematical and logical validity using the reference analysis, identifying error type and suggesting corrections if found, and stopping analysis upon first error detection. **Stage 3 (Conclusion):** Generate a conclusion indicating both solution correctness (binary) and the first error step (if any). **Stage 4 (Correction):** Generate a correction by following the original approach up to the error step (if any), then completing with proper correction.

2.4 Self-Validation

With self-generated critique data, we apply post-validation techniques to further enhance the quality of generated outputs. This process specifically filters out low-quality cases where

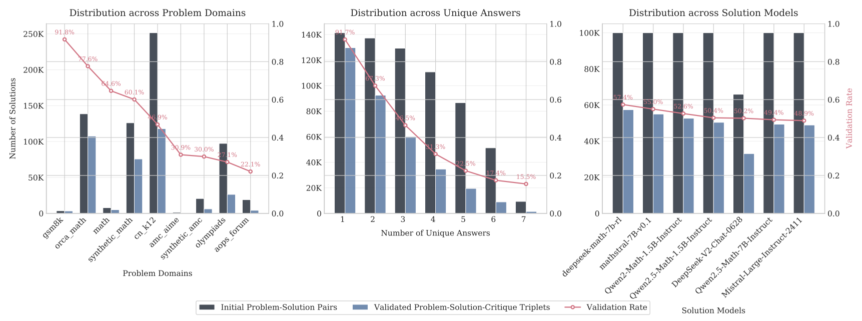


Figure 2: Data statistics before and after self-critic and self-validation filtering.

the model blindly approves all intermediate steps, only to suddenly reject the final answer upon detecting a discrepancy (see Appendix E).

To address these challenges, we employ direct validation on the correction part of the critique. Formally, we have that:

$$v_\theta(c) = \begin{cases} 1 & \text{if } g^l_\theta(p,t) = (1,-1) \\ 0 & \text{otherwise} \end{cases}$$

where t is the correction part of critique c, and g^l_θ (prompt template in Appendix C) denotes direct critic's conclusion generation function that outputs a tuple (y, j) as defined in Section 2.1. Here $g^l_\theta(p,t) = (1,-1)$ indicates that Direct Critic validates the correction t as a fully correct solution with no errors ($y = 1, j = -1$). This validation mechanism ensures that only critiques leading to verifiably correct solutions are used for self-training.

2.5 Self-Training

Let $\mathcal{V}$ denote the set of validated solution-critique pairs across all problems: $\mathcal{V} = \{(p,s,c) | p \in \mathcal{P}_{valid}, s \in \mathcal{S}_p, v_\theta(c) = 1\}$. For each validated triplet (p,s,c), we construct a training instance. The input to the model consists of the problem p and the student solution s. The target for fine-tuning consists of the critique components (e,l,t) from c. Crucially, the reference solution used during data generation is **not** provided as input during training. This ensures the model learns a general critique ability independent of reference solutions at inference time. We then fine-tune Qwen2.5-72B-Instruct to minimize the cross-entropy loss.

3 Experiments

3.1 Statistics of SCRIT

We present detailed statistics of data flow through each component of our framework.

Solution Collection We start with 452K problem-answer pairs from our own NuminaMath dataset (see Appendix B). For solution generation, we employ 7 models of varying capabilities as described in Section 2.2. Each model generates one solution per problem, with solutions classified as correct or incorrect based on their final answers using Qwen2.5-72B-Instruct (detailed in Appendix I). Then we apply two filtering criteria: (a) Each problem must have at least one correct and one incorrect solution to enable contrastive learning; (b) Solutions from each model are capped at 50K for both correct and incorrect categories. After filtering, we obtain 665K problem-solution pairs, evenly split between good solutions (332K) and bad solutions (332K).

Self-Critic & Self-Validation To analyze the self-critic and self-validation step, we track the data flow from the initial 665K problem-solution pairs through these steps. Out of these pairs, 342K (51.4%) successfully pass the self-critic and self-validation step, yielding high-quality problem-solution-critique triplets. Detailed analysis in Figure 2 reveals systematic patterns: validation rates decrease from elementary domains (GSM8K: 91.8%, ORCA Math: 77.6%) to competition-level problems (Olympiads: 27.1%), show strong negative correlation with problem complexity (91.7% for single-answer problems to 15.5% for seven-answer problems), while remaining relatively consistent across solution models (48.9% to 57.4%).

Table 1: Performance comparison on Critic and Correct protocol.

Model	RealCritic								Avg.
	ARC-C	College Math	GPQA	GSM8K	MATH	Minerva Math	MMLU STEM	Olympiad Bench	
Critic on deliberately incorrect solutions									
Qwen2.5-72B-Instruct	80.6	27.6	16.3	79.5	51.1	15.7	27.4	19.5	39.7
+ SCRIT	**86.7**	**32.6**	**25.3**	**88.3**	**66.0**	**23.4**	**50.7**	**27.0**	**50.0**
o1-mini	74.9	34.8	26.3	88.6	78.0	23.8	45.5	40.8	51.6
Critic on balanced solutions									
Qwen2.5-72B-Instruct	85.2	**50.9**	**31.1**	88.3	72.0	**47.1**	42.1	44.6	57.7
+ SCRIT	**90.1**	50.5	29.5	**94.1**	**75.7**	45.6	**64.7**	**46.4**	**62.1**
o1-mini	83.7	52.7	45.3	93.0	85.8	49.8	57.9	57.3	65.7
Critic on Qwen2.5-72B-Instruct's own solution									
Qwen2.5-72B-Instruct	**93.5**	45.9	32.6	96.7	**83.6**	38.3	59.6	43.4	61.7
+ SCRIT	91.3	45.9	**35.3**	96.7	82.5	**38.7**	**67.5**	**45.3**	**62.9**
o1-mini	93.9	47.0	36.8	96.7	89.9	40.2	68.5	53.6	65.8

This suggests our self-validation process is more sensitive to problem difficulty than to the source model. Analysis of error positions in critiqued solutions (see Figure 16) reveals that a majority of errors occur in earlier steps, aligning well with human-labeled error distributions in ProcessBench (Zheng et al., 2024a). This correlation suggests that our self-critic framework captures human-like error identification patterns.

Self-Training We maintain a balanced 1:1 ratio between correct and incorrect solutions, resulting in 170K training examples. These balanced training data are used to fine-tune Qwen2.5-72B-Instruct following Section 2.5 (complete training details in Appendix J).

Table 2: Performance comparison on Critic and Correct with Error Identification protocol.

Model	PRM800K	ProcessBench				Avg.
		GSM8K	MATH	Olympiad Bench	OmniMath	
Qwen2.5-72B-Instruct	23.7	68.9	50.9	25.5	20.0	37.8
+ SCRIT	**24.6**	**80.2**	**60.0**	**32.5**	**27.8**	**45.0**
o1-mini	34.0	88.0	81.1	53.0	38.6	58.9

3.2 Evaluation

We present two complementary evaluation protocols to assess critique capabilities:

Critic and Correct The first protocol evaluates a model's ability to critic and correct a given solution, following the assumption (Zheng et al., 2024b) that effective critiques should guide the correction of errors. We conduct experiments on RealCritic (Tang et al., 2025), which spans benchmarks from two key domains: **Mathematical Reasoning** (GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), College Math (Tang et al., 2024), Minerva Math (Lewkowycz et al., 2022), OlympiadBench (He et al., 2024)) and **Scientific Reasoning** (ARC-C (Clark et al., 2018), GPQA (Rein et al., 2023), MMLU-STEM (Hendrycks et al., 2020)). Evaluation is conducted across 3 scenarios: critic on incorrect solutions, balanced solutions, and the base model's self-generated solutions (i.e., Qwen2.5-72B-Instruct's own solutions).

Critic and Correct with Error Identification The second protocol requires models to provide accurate correction and identify the first error step. We evaluate on PRM800K (Lightman et al., 2023)[1] and ProcessBench (Zheng et al., 2024a), which contain human-labeled error steps from advanced models across GSM8K, MATH, OlympiadBench, and Omni-Math. Following ProcessBench, we use the F1 score of accuracies on incorrect and correct samples as our metric, with two adaptations to ensure critique effectiveness (See Appendix G).

Baselines Since our goal is to improve Qwen2.5-72B-Instruct's critique ability through self-evolution, we use the original Qwen2.5-72B-Instruct as our primary baseline. Additionally, we compare against o1-mini (OpenAI, 2024), currently one of the most capable models in terms of critique ability (Zheng et al., 2024a).

[1]`https://github.com/openai/prm800k/blob/main/prm800k/data/phase2_test.jsonl`

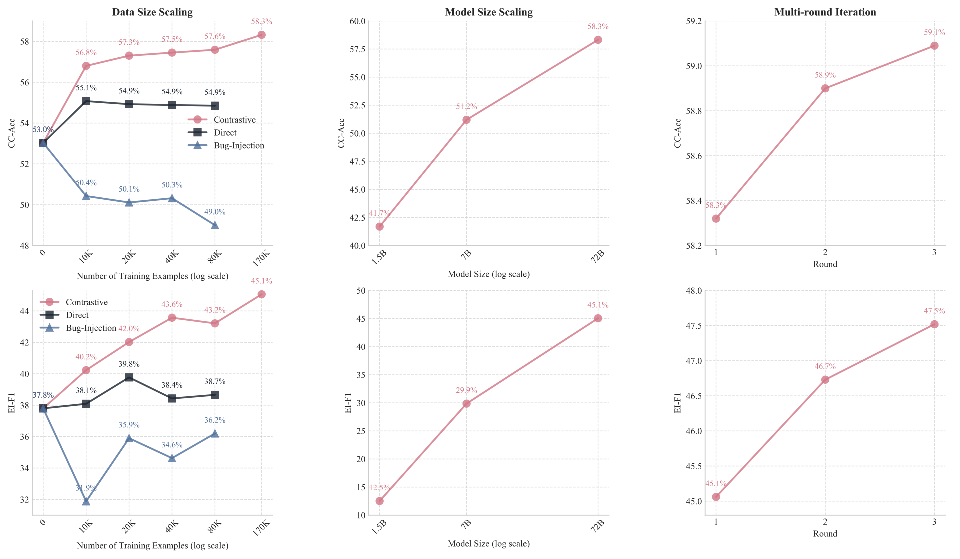


Figure 3: Scaling and multi-round performance analysis. **Left panel:** Data size scaling of Contrastive Critic, Direct Critic, and Bug-Injection Critic. **Middle panel:** Model size scaling from 1.5B to 72B parameters. **Right panel:** Multi-round self-evolving over 3 iterations.

3.3 Main Results

Critic and Correct Table 1 presents results across three increasingly challenging scenarios. SCRIT substantially improves over the base Qwen2.5-72B-Instruct model on deliberately incorrect solutions (50.0% vs 39.7%), maintains a 4.4% advantage on balanced solutions, and even improves when critiquing the base model's own solutions (62.9% vs 61.7%). These results represent a 10.0% relative gain in critique-correction accuracy across all scenarios, approaching the performance of o1-mini.

Critic and Correct with Error Identification As shown in Table 2, SCRIT shows strong capabilities in error identification, achieving consistent improvements across all datasets in both PRM800K and ProcessBench. The average F1 score improves from 37.8% to 45.0% (a 19.0% relative improvement), with strong gains on mathematical reasoning tasks (GSM8K: +11.3%, MATH: +9.1%). While there remains a gap with o1-mini, SCRIT's improvements are notable given its self-evolving nature without reliance on external supervision.

4 Analysis

Throughout this section, we report two metrics: critique-correction accuracy (CC-Acc) from the Critic and Correct protocol, which is averaged across three scenarios, and error identification F1-score (EI-F1) from the Critic and Correct with Error Identification protocol.

4.1 Generalization to Scientific Reasoning

A key question is whether SCRIT's self-evolving mechanism, primarily trained on mathematical data, can generalize to other complex reasoning domains. Our evaluation already includes scientific reasoning benchmarks (ARC-C, GPQA, MMLU-STEM), where Table 1 shows consistent improvements, confirming cross-domain generalization.

To further investigate this, we conducted an additional experiment where we trained a separate SCRIT model exclusively on 10K critique examples synthesized from scientific reasoning problems (from AM-Thinking-v1 (Ji et al., 2025)). As shown in Table 3, SCRIT trained on scientific data yields even stronger performance on scientific benchmarks (+10.6% on balanced solutions) while remaining competitive on math tasks. This not only demonstrates the framework's effectiveness beyond mathematics but also highlights its ability to capture and leverage domain-specific error patterns.

Table 3: Cross-domain generalization. SCRIT trained on domain-specific data shows strong in-domain performance and effective cross-domain transfer. CC-Acc is reported on balanced solutions.

Model	CC-Acc (Balanced)		Overall Avg
	Math Reasoning	Scientific Reasoning	
Qwen2.5-72B-Instruct	60.2	52.8	57.7
+ SCRIT (Math Data)	**64.5**	61.4	62.1
+ SCRIT (Scientific Data)	61.8	**67.4**	63.9

4.2 Scaling Behavior of SCRIT

We investigate SCRIT's performance scaling with training data and model size (see Figure 3).

Data Size Scaling For data scaling experiments, we train SCRIT with different amounts of training examples, ranging from 10K to 170K. Both CC-Acc and EI-F1 show consistent improvements with increased training data. The CC-Acc improves from 53.0% to 58.3%, with the steepest gains in the early stage (0-20K examples) and continued but more gradual improvements afterwards. Similarly, EI-F1 increases from 37.8% to 45.1%, demonstrating that SCRIT can effectively leverage more training data to evolve its critique capabilities.

Model Size Scaling We evaluate SCRIT across three model sizes of Qwen2.5: 1.5B, 7B, and 72B. Both metrics show strong positive correlation with model scale. The CC-Acc increases substantially from 41.7% (1.5B) to 51.2% (7B) and further to 58.3% (72B). The improvement is more pronounced for EI-F1, where metric rises from 12.5% to 29.9% and then to 45.1%, suggesting that larger models are particularly better at error identification.

To further verify that SCRIT is not only beneficial for large models, we applied the framework to a mid-sized model, Qwen2.5-32B-Instruct. SCRIT improved its performance meaningfully, with CC-Acc increasing from 53.9% to 56.5% and EI-F1 from 35.8% to 41.5%. This shows that the self-evolving mechanism is robust and effective across different model scales.

4.3 Which Critic Mechanism is Most Effective?

To identify the most effective critic mechanism for our self-evolving framework, we conduct strictly controlled experiments comparing three different critic approaches described in Section 2.3 using identical sets of problems and solutions.

Our experiments in Figure 3 reveal several key findings. First, Contrastive Critic shows strong performance from the early stages across both metrics: with just 10K training examples, it achieves 56.8% CC-Acc and 40.2% EI-F1, outperforming both Direct Critic and Bug-Injection Critic. More importantly, as training data increases to 170K examples, Contrastive Critic continues to show positive scaling behavior, reaching 58.3% CC-Acc and 45.1% EI-F1. In contrast, Direct Critic quickly plateaus at around 55.1% CC-Acc and 38.7% EI-F1, while Bug-Injection Critic exhibits performance degradation in CC-Acc (dropping to 49.0%) and unstable performance in EI-F1).

Through case studies (detailed in Appendices D and F), we identify the key mechanisms behind these performance differences. Direct Critic often falls into superficial critiquing, tending to blindly agree with solutions without deep understanding. Contrastive Critic avoids this pitfall by first analyzing reference solutions, enabling the model to develop a deeper understanding of the underlying mathematical concepts and solution strategies before attempting critique. While Bug-Injection Critic has the theoretical advantage of known error descriptions, our analysis reveals that model-injected bugs tend to be simplistic and repetitive, predominantly focusing on basic arithmetic errors and variable confusions, limiting its effectiveness in real-world scenarios where errors are more diverse and subtle.

4.4 Does Multi-round Iteration Foster Improvement?

A unique advantage of SCRIT is its ability to support multi-round self-evolution. After collecting the initial solutions, we can iteratively apply the self-critic generation, self-validation, and self-training steps to continuously improve the model's critique abilities. Specifically,

Table 4: Controlled ablation studies on SCRIT. Each experiment varies only the target component while keeping all other settings fixed at baseline: 10K training examples with contrastive critic and self-validation, diverse domains, all solution models, and balanced solution ratio. Red/green numbers indicate the relative performance decrease/increase.

Setting	CC-Acc	EI-F1
Baseline	56.8	40.2
Self-Validation		
Without Self-Validation	56.0 (-0.8)	37.2 (-3.0)
Problem Domain		
Limited to GSM8K + MATH	55.4 (-1.4)	38.8 (-1.4)
Problem Difficulty		
More Unique Answers First	55.8 (-1.0)	38.1 (-2.1)
Less Unique Answers First	56.2 (-0.6)	42.3 (+2.1)
Single Solution Model		
deepseek-math-7b-rl	56.5 (-0.3)	39.8 (-0.4)
mathstral-7B-v0.1	56.0 (-0.8)	39.2 (-1.0)
Mistral-Large-Instruct	56.3 (-0.5)	40.3 (+0.1)
DeepSeek-V2-Chat	56.3 (-0.5)	40.0 (-0.2)
Qwen2.5-Math-7B	56.2 (-0.6)	40.7 (+0.5)
Qwen2.5-Math-1.5B	56.2 (-0.6)	40.9 (+0.7)
Qwen2-Math-1.5B	55.9 (-0.9)	40.9 (+0.7)
Good:Bad Solution Ratio		
0.75:0.25	55.1 (-1.7)	38.1 (-2.1)
0.25:0.75	56.6 (-0.2)	41.0 (+0.8)

we conduct experiments with three rounds of iterations. Starting with Qwen2.5-72B-Instruct as the base model, we apply SCRIT to obtain an enhanced model with improved critique capabilities. As shown in Figure 3, using this enhanced model as the new base for Round 2, we observe further improvements in both metrics. Continuing with the Round 2 model for the third iteration, we achieve additional gains.

The performance demonstrates consistent positive scaling across both metrics through multiple rounds of iteration. This sustained improvement suggests that SCRIT can effectively leverage its own enhanced critique capabilities to generate increasingly higher-quality training data, enabling genuine self-evolution without external supervision.

4.5 How Important is Self-Validation?

To assess the necessity of self-validation in SCRIT, we conduct controlled experiments by removing the self-validation component while keeping all other settings identical. The results in Table 4 show clear performance degradation across both evaluation metrics: the CC-Acc drops by 0.8%, and more significantly, the EI-F1 decreases by 3.0%. Case analysis (see Appendix E) shows that the self-critic may still generate low-quality critiques, often blindly approving all intermediate steps only to suddenly claim ”the final step is incorrect” when encountering answer discrepancies. By incorporating self-validation, we are able to further enhance the quality of data for self-training.

4.6 Does Problem Domain Diversity Matter?

To investigate the importance of problem domain diversity, we conduct controlled experiments by restricting the training data to only GSM8K and MATH, while keeping other settings unchanged. This represents a significant reduction in domain coverage compared to our full setting which spans 9 sources ranging from elementary to competition-level mathematics. The results in Table 4 show the value of domain diversity: when training with limited domains, the CC-Acc drops by 1.4% and the EI-F1 decreases by 1.4%. It suggests that exposure to diverse problem-solving patterns and error types is crucial for developing robust critique abilities.

4.7 How Does Problem Difficulty Impact Performance?

To understand the impact of problem difficulty, we conduct experiments by selecting training examples based on the number of unique answers generated across solution models - a proxy for problem complexity. We study two settings: training with problems that have more unique answers (indicating higher complexity) versus those with fewer unique answers (indicating lower complexity). Interestingly, training with less complex problems leads to better performance in EI-F1 in Table 4. This result suggests that SCRIT can generate more effective critiques on simpler problems, possibly because the mathematical concepts and solution strategies in these problems are more structured and well-defined, enabling the model to develop more precise and reliable critique patterns.

This finding leaves space for future work: how to optimally select training examples based on difficulty levels in a self-evolving framework. While our current approach uses all available data, a more sophisticated curriculum that gradually increases problem complexity might lead to more effective self-evolution.

4.8 Does the Choice of Solution Model Matter?

To study whether critiquing solutions from different models affects SCRIT's performance, we conduct controlled experiments by restricting the solutions being critiqued to those from a single model while keeping other settings identical. Our results in Table 4 show that the source model of solutions has limited impact on SCRIT's final performance.

Since solution generation models only provide the solutions for constructing contrastive critique pairs and do not directly participate in improving critique effectiveness, their individual capabilities have less influence on the final performance. What matters more is how to construct diverse and informative contrastive pairs that help the model learn effective critique strategies, regardless of the solution models.

4.9 Optimal Ratio between Good and Bad Solutions?

Finally, we investigate the impact of good-to-bad solution ratio in the training data. As shown in Table 4, training with a higher proportion of bad solutions (0.25:0.75) shows better performance than using more good solutions (0.75:0.25). This suggests that exposure to more bad solutions helps SCRIT develop stronger error identification capabilities, likely because it provides more diverse examples of mathematical mistakes and their corresponding corrections. More importantly, analyzing incorrect solutions forces the model to actively engage in error detection and correction, rather than simply validating correct steps.

5 Conclusion

In this work, we present SCRIT, a self-evolving critique framework that enhances critique-correction accuracy and error detection in domains with verifiable solutions. By leveraging a contrastive-critic mechanism during data synthesis, SCRIT improves its capabilities without external supervision. Our experiments, spanning both mathematical and scientific reasoning, demonstrate that SCRIT scales with data and model size, shows strong cross-domain generalization, and benefits from self-validation. Future work could consider using SCRIT's high-quality critiques to label reasoning steps and optimize student models via reinforcement learning (e.g., (Saunders et al., 2022; McAleese et al., 2024)), or extending the framework to other structured domains like coding and logic.

Acknowledgments

The work of Tian Ding is supported by Hetao Shenzhen-Hong Kong Science and Technology Innovation Cooperation Zone Project (No.HZQSWS-KCCYB-2024016). The work of Ruoyu Sun is supported by NSFC (No. 12326608); Hetao Shenzhen-Hong Kong Science and Technology Innovation Cooperation Zone Project (No.HZQSWS-KCCYB-2024016); University Development Fund UDF01001491, the Chinese University of Hong Kong, Shenzhen; Guangdong Provincial Key Laboratory of Mathematical Foundations for Artificial Intelligence (2023B1212010001). The work of Benyou Wang is supported by Shenzhen Doctoral Startup Funding (RCBS20221008093330065), Tianyuan Fund for Mathematics of National Natural

Science Foundation of China (NSFC) (12326608), Shenzhen Science and Technology Program (Shenzhen Key Laboratory Grant No. ZDSYS20230626091302006), and Shenzhen Stability Science Program 2023, Shenzhen Key Lab of Multi-Modal Cognitive Computing.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Anthropic. The claude 3 model family: Opus, sonnet, haiku, 2024. URL https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Samuel R Bowman, Jeeyoon Hyun, Ethan Perez, Edwin Chen, Craig Pettit, Scott Heiner, Kamilė Lukošiūtė, Amanda Askell, Andy Jones, Anna Chen, et al. Measuring progress on scalable oversight for large language models. *arXiv preprint arXiv:2211.03540*, 2022.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

DeepSeek-AI. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model, 2024.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Minlie Huang, Nan Duan, and Weizhu Chen. Tora: A tool-integrated reasoning agent for mathematical problem solving. *arXiv preprint arXiv:2309.17452*, 2023.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint arXiv:2402.14008*, 2024.

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

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. *arXiv preprint arXiv:2103.03874*, 2021.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. *arXiv preprint arXiv:2310.01798*, 2023.

Yunjie Ji, Xiaoyu Tian, Sitong Zhao, Haotian Wang, Shuaiting Chen, Yiping Peng, Han Zhao, and Xiangang Li. Am-thinking-v1: Advancing the frontier of reasoning at 32b scale, 2025. URL https://arxiv.org/abs/2505.08311.

Tian Lan, Wenwei Zhang, Chengqi Lyu, Shuaibin Li, Chen Xu, Heyan Huang, Dahua Lin, Xian-Ling Mao, and Kai Chen. Training language models to critique with multi-agent feedback. *arXiv preprint arXiv:2410.15287*, 2024.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857, 2022.

Jia LI, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Costa Huang, Kashif Rasul, Longhui Yu, Albert Jiang, Ziju Shen, Zihan Qin, Bin Dong, Li Zhou, Yann Fleureau, Guillaume Lample, and Stanislas Polu. Numinamath. [https://huggingface.co/AI-MO/NuminaMath-CoT](https://github.com/project-numina/aimo-progress-prize/blob/main/report/numina_dataset.pdf), 2024.

Ziniu Li, Tian Xu, Yushun Zhang, Zhihang Lin, Yang Yu, Ruoyu Sun, and Zhi-Quan Luo. Remax: A simple, effective, and efficient reinforcement learning method for aligning large language models. In *Forty-first International Conference on Machine Learning*, 2024.

Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Zhi-Quan Luo, and Ruoyu Sun. Preserving diversity in supervised fine-tuning of large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=NQEe7B7bSw.

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

Nat McAleese, Rai Michael Pokorny, Juan Felipe Ceron Uribe, Evgenia Nitishinskaya, Maja Trebacz, and Jan Leike. Llm critics help catch llm bugs. *arXiv preprint arXiv:2407.00215*, 2024.

Mistral-AI. Mathstral, July 2024a. URL https://mistral.ai/news/mathstral/.

Mistral-AI. Mistral-large-2407, July 2024b. URL https://mistral.ai/news/mistral-large-2407/.

OpenAI. Learning to reason with LLMs. OpenAI Blog, Feb 2024. https://openai.com/index/learning-to-reason-with-llms.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems 35*, pp. 27730–27744, 2022.

Qwen-Team. Qwen2.5: A party of foundation models, September 2024. URL https://qwenlm.github.io/blog/qwen2.5/.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. Self-critiquing models for assisting human evaluators. *arXiv preprint arXiv:2206.05802*, 2022.

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

Zhengyang Tang, Xingxing Zhang, Benyou Wang, and Furu Wei. Mathscale: Scaling instruction tuning for mathematical reasoning. *arXiv preprint arXiv:2403.02884*, 2024.

Zhengyang Tang, Ziniu Li, Zhenyang Xiao, Tian Ding, Ruoyu Sun, Benyou Wang, Dayiheng Liu, Fei Huang, Tianyu Liu, Bowen Yu, et al. Realcritic: Towards effectiveness-driven evaluation of language model critiques. *arXiv preprint arXiv:2501.14492*, 2025.

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Raghavi Chandu, David Wadden, Kelsey MacMillan, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. How far can camels go? exploring the state of instruction tuning on open resources, 2023.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

Ling Yang, Zhaochen Yu, Tianjun Zhang, Minkai Xu, Joseph E Gonzalez, Bin Cui, and Shuicheng Yan. Supercorrect: Supervising and correcting language models with error-driven insights. *arXiv preprint arXiv:2410.09008*, 2024.

Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. Generative verifiers: Reward modeling as next-token prediction. *arXiv preprint arXiv:2408.15240*, 2024.

Xiaotian Zhang, Chunyang Li, Yi Zong, Zhengyu Ying, Liang He, and Xipeng Qiu. Evaluating the performance of large language models on gaokao benchmark. *arXiv preprint arXiv:2305.12474*, 2023.

Chujie Zheng, Zhenru Zhang, Beichen Zhang, Runji Lin, Keming Lu, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Processbench: Identifying process errors in mathematical reasoning. *arXiv preprint arXiv:2412.06559*, 2024a.

Xin Zheng, Jie Lou, Boxi Cao, Xueru Wen, Yuqiu Ji, Hongyu Lin, Yaojie Lu, Xianpei Han, Debing Zhang, and Le Sun. Critic-cot: Boosting the reasoning abilities of large language model via chain-of-thoughts critic. *arXiv preprint arXiv:2408.16326*, 2024b.

A Related Work

Scalable Oversight and Critic Models The challenge of providing effective feedback to language models on tasks difficult for humans to evaluate has attracted significant research attention. Early work by (Saunders et al., 2022) proposed fine-tuning LLMs to generate natural language critiques, introducing key components including critique generation, discrimination, and correction. Building on this direction, CriticGPT (McAleese et al., 2024) applied similar principles to code review tasks, incorporating RLHF and specialized human supervision through a "Tampering" step. These works established the importance of critique ability in enabling scalable oversight of language models.

Sources of Critique Supervision Existing approaches to developing critique abilities primarily rely on two types of supervision sources. The first category uses human supervision, as demonstrated in (Saunders et al., 2022) through direct human annotation and in (McAleese et al., 2024) through human-injected errors. The second category employs strong model supervision, exemplified by MultiCritique (Lan et al., 2024), which utilizes feedback from advanced models like GPT-4 to generate critiques for fine-tuning smaller models. Recent work GenRM (Zhang et al., 2024) proposes Chain-of-Thought Verifiers that generate step-wise critiques for mathematical reasoning, though still relying on human or stronger model supervision. While these approaches have shown promise, they are fundamentally limited by either the capabilities of their supervisors or the substantial costs associated with obtaining supervision.

Critic and Correct An important challenge in developing critique systems is how to evaluate the quality of critiques themselves, as directly measuring critique effectiveness is often as difficult as the original task. A key insight that has emerged in recent work is that truly effective critiques should be able to guide the correction of errors and lead to correct answers. This assumption provides a validation mechanism for critique quality and has been widely adopted in the field. For instance, Critic-CoT (Zheng et al., 2024b) combines step-wise critique generation with correction validation using GPT4-Turbo. Similarly, SuperCorrect (Yang et al., 2024) collects critique and corrections from teacher models like o1-mini. These works show the value of using correction as an objective mechanism to verify critique quality, though they still rely on stronger models for supervision.

In contrast to existing approaches that rely on either human annotations or stronger models for supervision, our work introduces SCRIT, a framework that enables self-evolution of critique abilities. By analyzing correct reference solutions to understand key mathematical concepts and strategies, then validating critiques through correction outcomes, our approach creates a closed-loop learning system that can improve its critique capabilities without external supervision.

B Computing Ground Truth Answers for NuminaMath

A large-scale dataset with reliable ground truth answers is fundamental to our work. We choose NuminaMath (LI et al., 2024) for its diversity, difficulty distribution, and scale (860K problems). However, as the correctness of solutions in the original dataset is not guaranteed, we develop a robust pipeline to compute reliable ground truth answers.

B.1 Answer Generation and Validation Pipeline

We employ Qwen2.5-Math-72B-Instruct (Qwen-Team, 2024) under tool-integrated (Gou et al., 2023) settings to generate solutions, as it demonstrates state-of-the-art performance across multiple mathematical reasoning benchmarks. The solutions are then evaluated using Qwen2.5-Math-RM-72B (Qwen-Team, 2024), a specialized reward model for mathematical reasoning. We consider a solution correct if its reward score exceeds a predefined threshold, and use its final answer as the ground truth.

B.2 Threshold Selection and Validation

To determine an appropriate reward threshold, we conduct extensive experiments:

- **Benchmark Validation:** We evaluate the threshold's effectiveness across multiple standard benchmarks including GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), GAOKAO2023-EN (Zhang et al., 2023), OlympiadBench (He et al., 2024), and College Math (Tang et al., 2024). With a threshold of 1.0, we achieve approximately 75% accuracy.

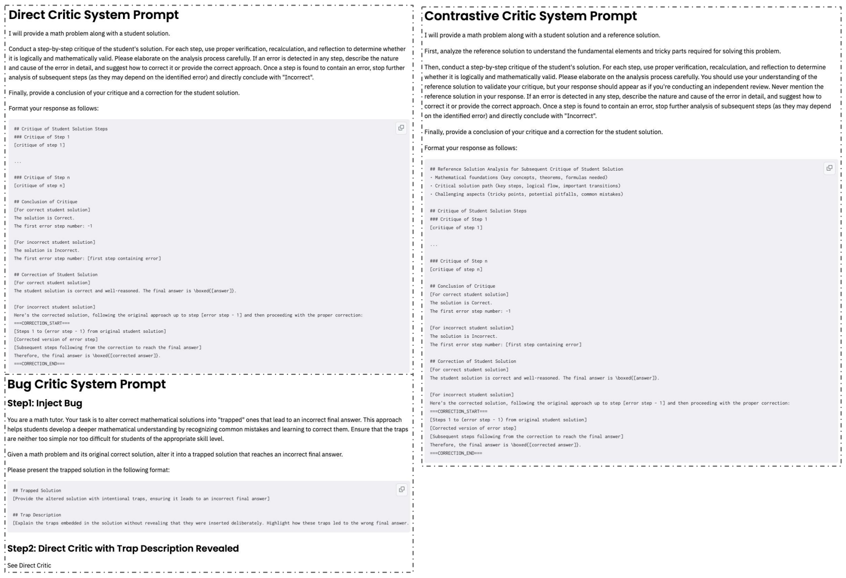
- **Human Evaluation:** We randomly sample 100 NuminaMath problems and conduct human evaluation of the answers selected using our threshold. The results show approximately 85% accuracy.

- **Comparison with Alternative Methods:** We explore majority voting among solutions from NuminaMath, Qwen2.5-Math-72B-Instruct, and Deepseek-V2-Chat-0628. However, this approach yields lower accuracy compared to our reward-based selection method.

After applying our pipeline with the validated threshold, we obtain a filtered dataset of 452K problem-answer pairs, which serves as the foundation for our work.

C Prompting Templates for Direct Critic, Bug-Injection Critic and Contrastive Critic

Here we present system prompts used for different critic mechanisms in Figure 4.



Direct Critic System Prompt

I will provide a math problem along with a student solution.

Conduct a step-by-step critique of the student's solution. For each step, use proper verification, recalculation, and reflection to determine whether it is logically and mathematically valid. Please elaborate on the analysis process carefully. If an error is detected in any step, describe the nature and cause of the error in detail, and suggest how to correct it or provide the correct approach. Once a step is found to contain an error, stop further analysis of subsequent steps (as they may depend on the identified error) and directly conclude with "Incorrect".

Finally, provide a conclusion of your critique and a correction for the student solution.

Format your response as follows:

```
## Critique of Student Solution Steps
### Critique of Step 1
[critique of step 1]

...

### Critique of Step n
[critique of step n]

## Conclusion of Critique
[For correct student solution]
The solution is Correct.
The first error step number: -1

[For incorrect student solution]
The solution is Incorrect.
The first error step number: [first step containing error]

## Correction of Student Solution
[For correct student solution]
The student solution is correct and well-reasoned. The final answer is \boxed{[answer]}.

[For incorrect student solution]
Here's the corrected solution, following the original approach up to step [error step - 1] and then proceeding with the proper correction:
===CORRECTION_START===
[Steps 1 to (error step - 1) from original student solution]
[Corrected version of error step]
[Subsequent steps following from the correction to reach the final answer]
Therefore, the final answer is \boxed{[corrected answer]}.
===CORRECTION_END===
```

Bug Critic System Prompt

Step1: Inject Bug

You are a math tutor. Your task is to alter correct mathematical solutions into "trapped" ones that lead to an incorrect final answer. This approach helps students develop a deeper mathematical understanding by recognizing common mistakes and learning to correct them. Ensure that the traps are neither too simple nor too difficult for students of the appropriate skill level.

Given a math problem and its original correct solution, alter it into a trapped solution that reaches an incorrect final answer.

Please present the trapped solution in the following format:

```
## Trapped Solution
[Provide the altered solution with intentional traps, ensuring it leads to an incorrect final answer]

## Trap Description
[Explain the traps embedded in the solution without revealing that they were inserted deliberately. Highlight how these traps led to the wrong final answer.
```

Step2: Direct Critic with Trap Description Revealed

See Direct Critic

Contrastive Critic System Prompt

I will provide a math problem along with a student solution and a reference solution.

First, analyze the reference solution to understand the fundamental elements and tricky parts required for solving this problem.

Then, conduct a step-by-step critique of the student's solution. For each step, use proper verification, recalculation, and reflection to determine whether it is logically and mathematically valid. Please elaborate on the analysis process carefully. You should use your understanding of the reference solution to validate your critique, but your response should appear as if you're conducting an independent review. Never mention the reference solution in your response. If an error is detected in any step, describe the nature and cause of the error in detail, and suggest how to correct it or provide the correct approach. Once a step is found to contain an error, stop further analysis of subsequent steps (as they may depend on the identified error) and directly conclude with "Incorrect".

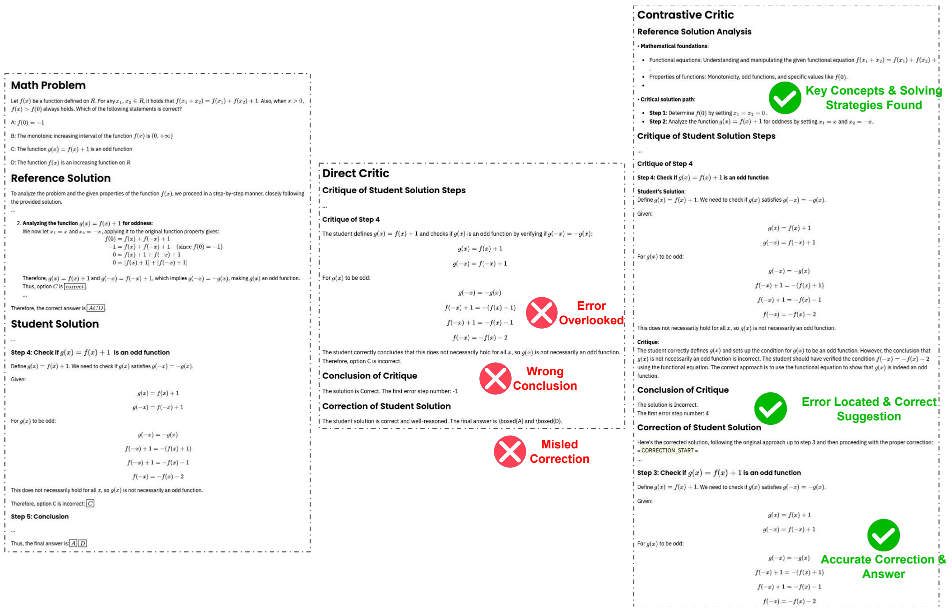
Finally, provide a conclusion of your critique and a correction for the student solution.

Format your response as follows:

```
## Reference Solution Analysis for Subsequent Critique of Student Solution
· Mathematical foundations (key concepts, theorems, formulas needed)
· Critical solution path (key steps, logical flow, important transitions)
· Challenging aspects (tricky points, potential pitfalls, common mistakes)

## Critique of Student Solution Steps
### Critique of Step 1
[critique of step 1]

...

### Critique of Step n
[critique of step n]

## Conclusion of Critique
[For correct student solution]
The solution is Correct.
The first error step number: -1

[For incorrect student solution]
The solution is Incorrect.
The first error step number: [first step containing error]

## Correction of Student Solution
[For correct student solution]
The student solution is correct and well-reasoned. The final answer is \boxed{[answer]}.

[For incorrect student solution]
Here's the corrected solution, following the original approach up to step [error step - 1] and then proceeding with the proper correction:
===CORRECTION_START===
[Steps 1 to (error step - 1) from original student solution]
[Corrected version of error step]
[Subsequent steps following from the correction to reach the final answer]
Therefore, the final answer is \boxed{[corrected answer]}.
===CORRECTION_END===
```

Figure 4: System prompts used for different critic mechanisms. **Top Left:** Direct Critic directly analyzes solution correctness without any additional context. **Bottom Left:** Bug-Injection Critic first injects bugs (Step 1) then direct critic on bug-injected solution (Step 2). **Right:** Contrastive Critic first analyzes a reference solution to understand key mathematical concepts before conducting step-wise critique.

D More Comparison between Direct Critic and Contrastive Critic



Math Problem

Let $f(x)$ be a function defined on R. For any $x_1, x_2 \in R$, it holds that $f(x_1+x_2) = f(x_1) + f(x_2) + 1$. Also, when $x > 0$, $f(x) > f(0)$ always holds. Which of the following statements is correct?

A: $f(0) = -1$

B: The monotonic increasing interval of the function $f(x)$ is $(0, +\infty)$

C: The function $g(x) = f(x) + 1$ is an odd function

D: The function $f(x)$ is an increasing function on R

Reference Solution

To analyze the problem and the given properties of the function $f(x)$, we proceed in a step-by-step manner, closely following the provided solution.

...

2. **Analyzing the function $g(x) = f(x) + 1$ for oddness**:
   We now let $x_1 = x$ and $x_2 = -x$, applying it to the original function property gives:
$$\begin{aligned} f(0) &= f(x) + f(-x) + 1 \\ -1 &= f(x) + f(-x) + 1 \quad (\text{since } f(0) = -1) \\ 0 &= f(x) + 1 + f(-x) + 1 \\ 0 &= [f(x)+1] + [f(-x)+1] \end{aligned}$$
   Therefore, $g(x) = f(x) + 1$ and $g(-x) = f(-x) + 1$, which implies $g(-x) = -g(x)$, making $g(x)$ an odd function. Thus, option C is $\boxed{\text{correct}}$.
   ...

Therefore, the correct answer is $\boxed{ACD}$.

Student Solution

...

Step 4: Check if $g(x) = f(x) + 1$ is an odd function

Define $g(x) = f(x) + 1$. We need to check if $g(x)$ satisfies $g(-x) = -g(x)$.

Given:

$$g(x) = f(x) + 1$$
$$g(-x) = f(-x) + 1$$

For $g(x)$ to be odd:

$$g(-x) = -g(x)$$
$$f(-x) + 1 = -(f(x) + 1)$$
$$f(-x) + 1 = -f(x) - 1$$
$$f(-x) = -f(x) - 2$$

This does not necessarily hold for all x, so $g(x)$ is not necessarily an odd function.

Therefore, option C is incorrect: $\boxed{C}$

Step 5: Conclusion

...

Thus, the final answer is: $\boxed{A}$ $\boxed{D}$

Direct Critic

Critique of Student Solution Steps

...

Critique of Step 4

The student defines $g(x) = f(x) + 1$ and checks if $g(x)$ is an odd function by verifying if $g(-x) = -g(x)$:

$$g(x) = f(x) + 1$$
$$g(-x) = f(-x) + 1$$

For $g(x)$ to be odd:

$$g(-x) = -g(x)$$
$$f(-x) + 1 = -(f(x) + 1)$$
$$f(-x) + 1 = -f(x) - 1$$
$$f(-x) = -f(x) - 2$$

The student correctly concludes that this does not necessarily hold for all x, so $g(x)$ is not necessarily an odd function. Therefore, option C is incorrect.

Conclusion of Critique

The solution is Correct. The first error step number: -1

Correction of Student Solution

The student solution is correct and well-reasoned. The final answer is \boxed{A} and \boxed{D}.

Contrastive Critic

Reference Solution Analysis

• **Mathematical foundations:**
- Functional equations: Understanding and manipulating the given functional equation $f(x_1+x_2) = f(x_1) + f(x_2) +$.
- Properties of functions: Monotonicity, odd functions, and specific values like $f(0)$.
-

• **Critical solution path:**
- **Step 1**: Determine $f(0)$ by setting $x_1 = x_2 = 0$.
- **Step 2**: Analyze the function $g(x) = f(x) + 1$ for oddness by setting $x_1 = x$ and $x_2 = -x$.

Critique of Student Solution Steps

...

Critique of Step 4

Step 4: Check if $g(x) = f(x) + 1$ is an odd function

Student's Solution:
Define $g(x) = f(x) + 1$. We need to check if $g(x)$ satisfies $g(-x) = -g(x)$.

Given:

$$g(x) = f(x) + 1$$
$$g(-x) = f(-x) + 1$$

For $g(x)$ to be odd:

$$g(-x) = -g(x)$$
$$f(-x) + 1 = -(f(x) + 1)$$
$$f(-x) + 1 = -f(x) - 1$$
$$f(-x) = -f(x) - 2$$

This does not necessarily hold for all x, so $g(x)$ is not necessarily an odd function.

Critique:
The student correctly defines $g(x)$ and sets up the condition for $g(x)$ to be an odd function. However, the conclusion that $g(x)$ is not necessarily an odd function is incorrect. The student should have verified the condition $f(-x) = -f(x) - 2$ using the functional equation. The correct approach is to use the functional equation to show that $g(x)$ is indeed an odd function.

Conclusion of Critique

The solution is Incorrect.
The first error step number: 4

Correction of Student Solution

Here's the corrected solution, following the original approach up to step 3 and then proceeding with the proper correction:
= CORRECTION_START =
...

Step 3: Check if $g(x) = f(x) + 1$ is an odd function

Define $g(x) = f(x) + 1$. We need to check if $g(x)$ satisfies $g(-x) = -g(x)$.

Given:

$$g(x) = f(x) + 1$$
$$g(-x) = f(-x) + 1$$

For $g(x)$ to be odd:

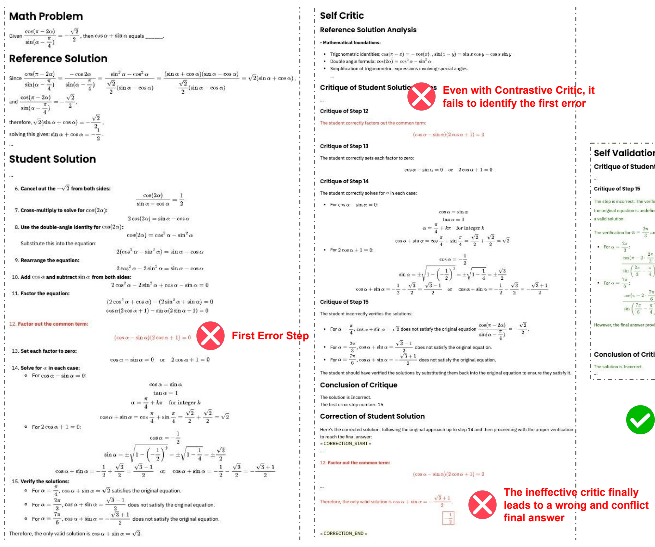
$$g(-x) = -g(x)$$
$$f(-x) + 1 = -(f(x) + 1)$$
$$f(-x) + 1 = -f(x) - 1$$
$$f(-x) = -f(x) - 2$$

Error Overlooked
Wrong Conclusion
Misled Correction
Key Concepts & Solving Strategies Found
Error Located & Correct Suggestion
Accurate Correction & Answer

Figure 5: Comparison between Direct Critic and Contrastive Critic. Direct Critic shows blind approval of the student solution, failing to identify any errors and providing misleading approval. In contrast, Contrastive Critic first analyzes the reference solution to understand key mathematical concepts, enabling it to precisely locate the error in the student solution. By developing understanding of the underlying mathematical concepts, Contrastive Critic successfully generate an effective critique that guides the correction process to reach the correct final answer.

E Self-Validation Cases

We present two cases demonstrating the effectiveness of our Self-Validation mechanism in filtering critiques based on Self-Critic's correction in Figures 6 and 7.



Math Problem

Given $\frac{\cos(\pi-2\alpha)}{\sin(\alpha-\frac{\pi}{4})}=-\frac{\sqrt{2}}{2}$, then $\cos\alpha+\sin\alpha$ equals ______.

Reference Solution

Since $\frac{\cos(\pi-2\alpha)}{\sin(\alpha-\frac{\pi}{4})}=\frac{-\cos 2\alpha}{\sin(\alpha-\frac{\pi}{4})}=\frac{\sin^2\alpha-\cos^2\alpha}{\frac{\sqrt{2}}{2}(\sin\alpha-\cos\alpha)}=\frac{(\sin\alpha+\cos\alpha)(\sin\alpha-\cos\alpha)}{\frac{\sqrt{2}}{2}(\sin\alpha-\cos\alpha)}=\sqrt{2}(\sin\alpha+\cos\alpha)$,

and $\frac{\cos(\pi-2\alpha)}{\sin(\alpha-\frac{\pi}{4})}=-\frac{\sqrt{2}}{2}$,

therefore, $\sqrt{2}(\sin\alpha+\cos\alpha)=-\frac{\sqrt{2}}{2}$,

solving this gives: $\sin\alpha+\cos\alpha=-\frac{1}{2}$.

...

Student Solution

...

6. **Cancel out the $-\sqrt{2}$ from both sides:**
$$\frac{\cos(2\alpha)}{\sin\alpha-\cos\alpha}=\frac{1}{2}$$
7. **Cross-multiply to solve for $\cos(2\alpha)$:**
$$2\cos(2\alpha)=\sin\alpha-\cos\alpha$$
8. **Use the double-angle identity for $\cos(2\alpha)$:**
$$\cos(2\alpha)=\cos^2\alpha-\sin^2\alpha$$
Substitute this into the equation:
$$2(\cos^2\alpha-\sin^2\alpha)=\sin\alpha-\cos\alpha$$
9. **Rearrange the equation:**
$$2\cos^2\alpha-2\sin^2\alpha=\sin\alpha-\cos\alpha$$
10. **Add $\cos\alpha$ and subtract $\sin\alpha$ from both sides:**
$$2\cos^2\alpha-2\sin^2\alpha+\cos\alpha-\sin\alpha=0$$
11. **Factor the equation:**
$$(2\cos^2\alpha+\cos\alpha)-(2\sin^2\alpha+\sin\alpha)=0$$
$$\cos\alpha(2\cos\alpha+1)-\sin\alpha(2\sin\alpha+1)=0$$
12. **Factor out the common term:**
$$(\cos\alpha-\sin\alpha)(2\cos\alpha+1)=0$$
13. **Set each factor to zero:**
$$\cos\alpha-\sin\alpha=0 \quad\text{or}\quad 2\cos\alpha+1=0$$
14. **Solve for α in each case:**
   - For $\cos\alpha-\sin\alpha=0$:
$$\cos\alpha=\sin\alpha$$
$$\tan\alpha=1$$
$$\alpha=\frac{\pi}{4}+k\pi \quad \text{for integer } k$$
$$\cos\alpha+\sin\alpha=\cos\frac{\pi}{4}+\sin\frac{\pi}{4}=\frac{\sqrt{2}}{2}+\frac{\sqrt{2}}{2}=\sqrt{2}$$
   - For $2\cos\alpha+1=0$:
$$\cos\alpha=-\frac{1}{2}$$
$$\sin\alpha=\pm\sqrt{1-\left(-\frac{1}{2}\right)^2}=\pm\sqrt{1-\frac{1}{4}}=\pm\frac{\sqrt{3}}{2}$$
$$\cos\alpha+\sin\alpha=-\frac{1}{2}+\frac{\sqrt{3}}{2}=\frac{\sqrt{3}-1}{2} \quad\text{or}\quad \cos\alpha+\sin\alpha=-\frac{1}{2}-\frac{\sqrt{3}}{2}=-\frac{\sqrt{3}+1}{2}$$
15. **Verify the solutions:**
   - For $\alpha=\frac{\pi}{4}$, $\cos\alpha+\sin\alpha=\sqrt{2}$ satisfies the original equation.
   - For $\alpha=\frac{2\pi}{3}$, $\cos\alpha+\sin\alpha=\frac{\sqrt{3}-1}{2}$ does not satisfy the original equation.
   - For $\alpha=\frac{7\pi}{6}$, $\cos\alpha+\sin\alpha=-\frac{\sqrt{3}+1}{2}$ does not satisfy the original equation.

Therefore, the only valid solution is $\cos\alpha+\sin\alpha=\sqrt{2}$.

Self Critic

Reference Solution Analysis

- **Mathematical foundations:**
  - Trigonometric identities: $\cos(\pi-x)=-\cos(x)$, $\sin(x-y)=\sin x\cos y-\cos x\sin y$
  - Double angle formula: $\cos(2\alpha)=\cos^2\alpha-\sin^2\alpha$
  - Simplification of trigonometric expressions involving special angles

  ...

Critique of Student Solution Steps

...

Critique of Step 12

The student correctly factors out the common term:
$$(\cos\alpha-\sin\alpha)(2\cos\alpha+1)=0$$

Critique of Step 13

The student correctly sets each factor to zero:
$$\cos\alpha-\sin\alpha=0 \quad\text{or}\quad 2\cos\alpha+1=0$$

Critique of Step 14

The student correctly solves for α in each case:

- For $\cos\alpha-\sin\alpha=0$:
$$\cos\alpha=\sin\alpha$$
$$\tan\alpha=1$$
$$\alpha=\frac{\pi}{4}+k\pi \quad \text{for integer } k$$
$$\cos\alpha+\sin\alpha=\cos\frac{\pi}{4}+\sin\frac{\pi}{4}=\frac{\sqrt{2}}{2}+\frac{\sqrt{2}}{2}=\sqrt{2}$$
- For $2\cos\alpha+1=0$:
$$\cos\alpha=-\frac{1}{2}$$
$$\sin\alpha=\pm\sqrt{1-\left(-\frac{1}{2}\right)^2}=\pm\sqrt{1-\frac{1}{4}}=\pm\frac{\sqrt{3}}{2}$$
$$\cos\alpha+\sin\alpha=-\frac{1}{2}+\frac{\sqrt{3}}{2}=\frac{\sqrt{3}-1}{2} \quad\text{or}\quad \cos\alpha+\sin\alpha=-\frac{1}{2}-\frac{\sqrt{3}}{2}=-\frac{\sqrt{3}+1}{2}$$

Critique of Step 15

The student incorrectly verifies the solutions:

- For $\alpha=\frac{\pi}{4}$, $\cos\alpha+\sin\alpha=\sqrt{2}$ does not satisfy the original equation $\frac{\cos(\pi-2\alpha)}{\sin(\alpha-\frac{\pi}{4})}=-\frac{\sqrt{2}}{2}$.
- For $\alpha=\frac{2\pi}{3}$, $\cos\alpha+\sin\alpha=\frac{\sqrt{3}-1}{2}$ does not satisfy the original equation.
- For $\alpha=\frac{7\pi}{6}$, $\cos\alpha+\sin\alpha=-\frac{\sqrt{3}+1}{2}$ does not satisfy the original equation.

The student should have verified the solutions by substituting them back into the original equation to ensure they satisfy it.

Conclusion of Critique

The solution is Incorrect.
The first error step number: 15

Correction of Student Solution

Here's the corrected solution, following the original approach up to step 14 and then proceeding with the proper verification to reach the final answer:
= CORRECTION_START =
...

12. **Factor out the common term:**
$$(\cos\alpha-\sin\alpha)(2\cos\alpha+1)=0$$

...

Therefore, the only valid solution is $\cos\alpha+\sin\alpha=-\frac{\sqrt{3}+1}{2}$.
$\boxed{-\frac{1}{2}}$

= CORRECTION_END =

Self Validation

Critique of Student Solution Steps

...

Critique of Step 15

The step is incorrect. The verification of the solutions is flawed. Specifically, the verification for $\alpha=\frac{\pi}{4}$ is incorrect because the original equation is undefined at $\alpha=\frac{\pi}{4}$ due to the denominator $\sin\left(\frac{\pi}{4}-\frac{\pi}{4}\right)=\sin(0)=0$. Therefore, $\alpha=\frac{\pi}{4}$ is not a valid solution.

The verification for $\alpha=\frac{2\pi}{3}$ and $\alpha=\frac{7\pi}{6}$ is also incorrect. The correct verification should be:

- For $\alpha=\frac{2\pi}{3}$:
$$\frac{\cos(\pi-2\cdot\frac{2\pi}{3})}{\sin\left(\frac{2\pi}{3}-\frac{\pi}{4}\right)}=\frac{\cos\left(\frac{\pi}{3}\right)}{\sin\left(\frac{5\pi}{12}\right)}=\frac{\frac{1}{2}}{\frac{\sqrt{6}+\sqrt{2}}{4}}=\frac{2}{\sqrt{6}+\sqrt{2}}\neq-\frac{\sqrt{2}}{2} \quad \text{(not a valid solution)}$$
- For $\alpha=\frac{7\pi}{6}$:
$$\frac{\cos(\pi-2\cdot\frac{7\pi}{6})}{\sin\left(\frac{7\pi}{6}-\frac{\pi}{4}\right)}=\frac{\cos\left(\frac{5\pi}{3}\right)}{\sin\left(\frac{13\pi}{12}\right)}=\frac{\frac{1}{2}}{-\frac{\sqrt{6}-\sqrt{2}}{4}}=-\frac{2}{\sqrt{6}-\sqrt{2}}=-\frac{\sqrt{2}}{2} \quad \text{(valid solution)}$$

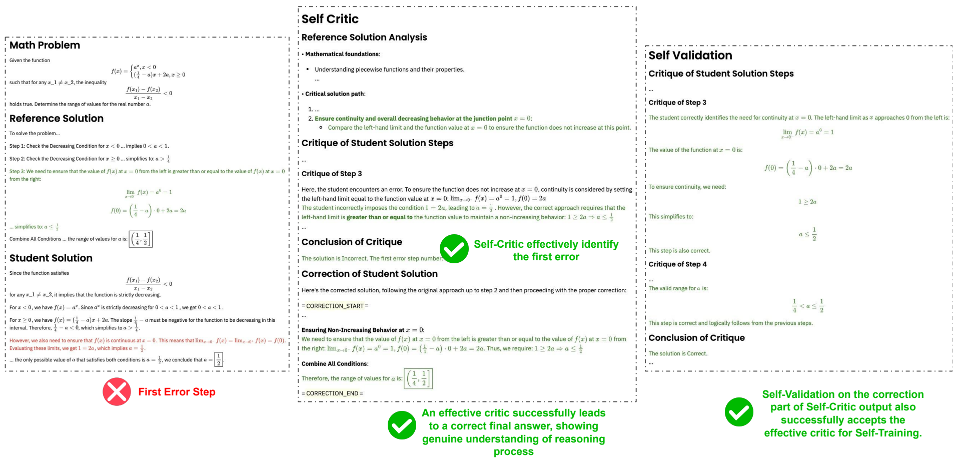
However, the final answer provided by the student is incorrect. The correct value of $\cos\alpha+\sin\alpha$ for $\alpha=\frac{7\pi}{6}$ is:
$$\cos\alpha+\sin\alpha=-\frac{1}{2}-\frac{\sqrt{3}}{2}=-\frac{\sqrt{3}+1}{2}$$

Conclusion of Critique

The solution is Incorrect.
...

First Error Step
Even with Contrastive Critic, it fails to identify the first error
The ineffective critic finally leads to a wrong and conflict final answer
Self-Validation on the correction part of Self-Critic output successfully notices the wrong and conflict part in correction, and rejects the ineffective critic for Self-Training.

Figure 6: Case1: Self-Validation rejects an ineffective critic: Despite having access to a reference solution and using contrastive learning, the critic fails to identify Step 12 as the first error in solving a trigonometric equation. The subsequent correction leads to a conflicting final answer. The self-validation mechanism successfully detects this inconsistency and rejects this ineffective critique from the training data.



Math Problem

Given the function
$$f(x)=\begin{cases}a^x, x<0\\ (\frac{1}{4}-a)x+2a, x\geq 0\end{cases}$$
such that for any $x\_1\neq x\_2$, the inequality
$$\frac{f(x_1)-f(x_2)}{x_1-x_2}<0$$
holds true. Determine the range of values for the real number a.

Reference Solution

To solve the problem...

Step 1: Check the Decreasing Condition for $x<0$ ... implies $0<a<1$.

Step 2: Check the Decreasing Condition for $x\geq 0$ ... simplifies to: $a>\frac{1}{4}$

Step 3: We need to ensure that the value of $f(x)$ at $x=0$ from the left is greater than or equal to the value of $f(x)$ at $x=0$ from the right:
$$\lim_{x\to 0^-} f(x)=a^0=1$$
$$f(0)=\left(\frac{1}{4}-a\right)\cdot 0+2a=2a$$
... simplifies to: $a\leq\frac{1}{2}$

Combine All Conditions ... the range of values for a is: $\boxed{\left(\frac{1}{4},\frac{1}{2}\right]}$

Student Solution

Since the function satisfies
$$\frac{f(x_1)-f(x_2)}{x_1-x_2}<0$$
for any $x\_1\neq x\_2$, it implies that the function is strictly decreasing.

For $x<0$, we have $f(x)=a^x$. Since a^x is strictly decreasing for $0<a<1$, we get $0<a<1$.

For $x\geq 0$, we have $f(x)=(\frac{1}{4}-a)x+2a$. The slope $\frac{1}{4}-a$ must be negative for the function to be decreasing in this interval. Therefore, $\frac{1}{4}-a<0$, which simplifies to $a>\frac{1}{4}$.

However, we also need to ensure that $f(x)$ is continuous at $x=0$. This means that $\lim_{x\to 0^-} f(x)=\lim_{x\to 0^+} f(x)=f(0)$. Evaluating these limits, we get $1=2a$, which implies $a=\frac{1}{2}$.

... the only possible value of a that satisfies both conditions is $a=\frac{1}{2}$, we conclude that $a=\boxed{\frac{1}{2}}$.

Self Critic

Reference Solution Analysis

- **Mathematical foundations:**
  - Understanding piecewise functions and their properties.

  ...
- **Critical solution path:**
  1. ...
  2. **Ensure continuity and overall decreasing behavior at the junction point $x=0$:**
     - Compare the left-hand limit and the function value at $x=0$ to ensure the function does not increase at this point.

Critique of Student Solution Steps

...

Critique of Step 3

Here, the student encounters an error. To ensure the function does not increase at $x=0$, continuity is considered by setting the left-hand limit equal to the function value at $x=0$: $\lim_{x\to 0^-} f(x)=a^0=1, f(0)=2a$
The student incorrectly imposes the condition $1=2a$, leading to $a=\frac{1}{2}$. However, the correct approach requires that the left-hand limit is **greater than or equal to** the function value to maintain a non-increasing behavior: $1\geq 2a\Rightarrow a\leq\frac{1}{2}$
...

Conclusion of Critique

The solution is Incorrect. The first error step number:

Correction of Student Solution

Here's the corrected solution, following the original approach up to step 2 and then proceeding with the proper correction:

= CORRECTION_START =
...

Ensuring Non-Increasing Behavior at $x=0$:
We need to ensure that the value of $f(x)$ at $x=0$ from the left is greater than or equal to the value of $f(x)$ at $x=0$ from the right: $\lim_{x\to 0^-} f(x)=a^0=1, f(0)=(\frac{1}{4}-a)\cdot 0+2a=2a$. Thus, we require: $1\geq 2a\Rightarrow a\leq\frac{1}{2}$

Combine All Conditions:

Therefore, the range of values for a is: $\boxed{\left(\frac{1}{4},\frac{1}{2}\right]}$

= CORRECTION_END =

Self Validation

Critique of Student Solution Steps

...

Critique of Step 3

The student correctly identifies the need for continuity at $x=0$. The left-hand limit as x approaches 0 from the left is:
$$\lim_{x\to 0^-} f(x)=a^0=1$$
The value of the function at $x=0$ is:
$$f(0)=\left(\frac{1}{4}-a\right)\cdot 0+2a=2a$$
To ensure continuity, we need:
$$1\geq 2a$$
This simplifies to:
$$a\leq\frac{1}{2}$$
This step is also correct.

Critique of Step 4

...
The valid range for a is:
$$\frac{1}{4}<a\leq\frac{1}{2}$$
This step is correct and logically follows from the previous steps.

Conclusion of Critique

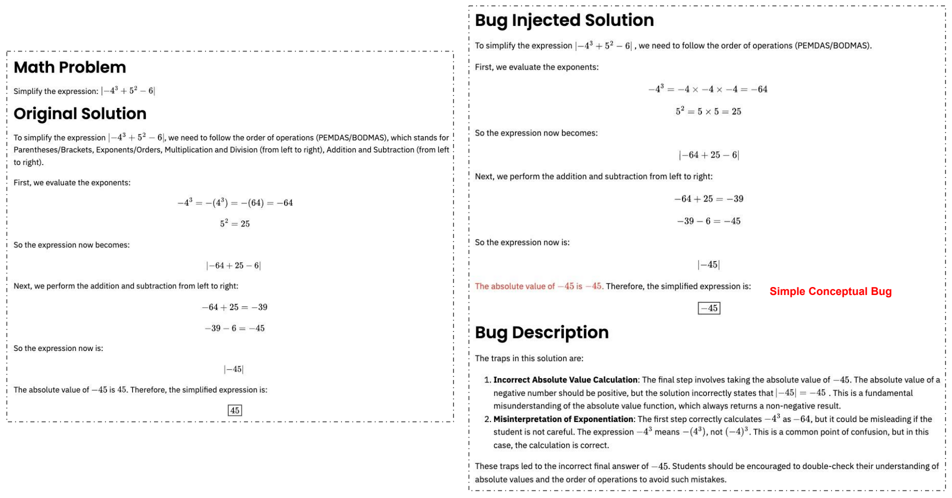
The solution is Correct.
...

First Error Step
Self-Critic effectively identify the first error
An effective critic successfully leads to a correct final answer, showing genuine understanding of reasoning process
Self-Validation on the correction part of Self-Critic output also successfully accepts the effective critic for Self-Training.

Figure 7: Case2: Self-Validation accepts an effective critic: An example of effective critique that correctly identifies Step 3 as the error point where continuity requirements are mishandled. The correction follows logical mathematical reasoning and arrives at the correct final answer, which is then verified and accepted by the self-validation mechanism for training.

F Bug-Injection Case Study

Here we show examples of oversimplified bugs injected by Bug-Injection Critic. These examples illustrate how Bug-Injection Critic tends to generate overly simplistic errors (e.g., misunderstanding basic math properties, variable confusion) rather than more sophisticated mathematical reasoning errors that typically occur in complex problem-solving.



Math Problem

Simplify the expression: $|-4^3 + 5^2 - 6|$

Original Solution

To simplify the expression $|-4^3 + 5^2 - 6|$, we need to follow the order of operations (PEMDAS/BODMAS), which stands for Parentheses/Brackets, Exponents/Orders, Multiplication and Division (from left to right), Addition and Subtraction (from left to right).

First, we evaluate the exponents:

$$-4^3 = -(4^3) = -(64) = -64$$

$$5^2 = 25$$

So the expression now becomes:

$$|-64 + 25 - 6|$$

Next, we perform the addition and subtraction from left to right:

$$-64 + 25 = -39$$

$$-39 - 6 = -45$$

So the expression now is:

$$|-45|$$

The absolute value of -45 is 45. Therefore, the simplified expression is:

$$\boxed{45}$$

Bug Injected Solution

To simplify the expression $|-4^3 + 5^2 - 6|$, we need to follow the order of operations (PEMDAS/BODMAS).

First, we evaluate the exponents:

$$-4^3 = -4 \times -4 \times -4 = -64$$

$$5^2 = 5 \times 5 = 25$$

So the expression now becomes:

$$|-64 + 25 - 6|$$

Next, we perform the addition and subtraction from left to right:

$$-64 + 25 = -39$$

$$-39 - 6 = -45$$

So the expression now is:

$$|-45|$$

The absolute value of -45 is -45. Therefore, the simplified expression is:

$$\boxed{-45}$$

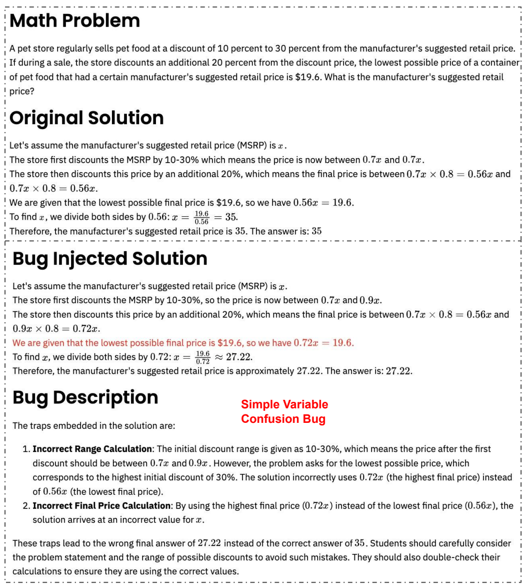
Bug Description

The traps in this solution are:

1. **Incorrect Absolute Value Calculation**: The final step involves taking the absolute value of -45. The absolute value of a negative number should be positive, but the solution incorrectly states that $|-45| = -45$. This is a fundamental misunderstanding of the absolute value function, which always returns a non-negative result.
2. **Misinterpretation of Exponentiation**: The first step correctly calculates -4^3 as -64, but it could be misleading if the student is not careful. The expression -4^3 means $-(4^3)$, not $(-4)^3$. This is a common point of confusion, but in this case, the calculation is correct.

These traps led to the incorrect final answer of -45. Students should be encouraged to double-check their understanding of absolute values and the order of operations to avoid such mistakes.

Figure 8: An example of oversimplified bugs injected by Bug-Injection Critic: A conceptual bug involving basic misunderstanding of absolute value property.



Math Problem

A pet store regularly sells pet food at a discount of 10 percent to 30 percent from the manufacturer's suggested retail price. If during a sale, the store discounts an additional 20 percent from the discount price, the lowest possible price of a container of pet food that had a certain manufacturer's suggested retail price is $19.6. What is the manufacturer's suggested retail price?

Original Solution

Let's assume the manufacturer's suggested retail price (MSRP) is x .
The store first discounts the MSRP by 10-30% which means the price is now between $0.7x$ and $0.7x$.
The store then discounts this price by an additional 20%, which means the final price is between $0.7x \times 0.8 = 0.56x$ and $0.7x \times 0.8 = 0.56x$.
We are given that the lowest possible final price is $19.6, so we have $0.56x = 19.6$.
To find x, we divide both sides by 0.56: $x = \frac{19.6}{0.56} = 35$.
Therefore, the manufacturer's suggested retail price is 35. The answer is: 35

Bug Injected Solution

Let's assume the manufacturer's suggested retail price (MSRP) is x.
The store first discounts the MSRP by 10-30%, so the price is now between $0.7x$ and $0.9x$.
The store then discounts this price by an additional 20%, which means the final price is between $0.7x \times 0.8 = 0.56x$ and $0.9x \times 0.8 = 0.72x$.
We are given that the lowest possible final price is $19.6, so we have $0.72x = 19.6$.
To find x, we divide both sides by 0.72: $x = \frac{19.6}{0.72} \approx 27.22$.
Therefore, the manufacturer's suggested retail price is approximately 27.22. The answer is: 27.22.

Bug Description

The traps embedded in the solution are:

1. **Incorrect Range Calculation**: The initial discount range is given as 10-30%, which means the price after the first discount should be between $0.7x$ and $0.9x$. However, the problem asks for the lowest possible price, which corresponds to the highest initial discount of 30%. The solution incorrectly uses $0.72x$ (the highest final price) instead of $0.56x$ (the lowest final price).
2. **Incorrect Final Price Calculation**: By using the highest final price ($0.72x$) instead of the lowest final price ($0.56x$), the solution arrives at an incorrect value for x.

These traps lead to the wrong final answer of 27.22 instead of the correct answer of 35. Students should carefully consider the problem statement and the range of possible discounts to avoid such mistakes. They should also double-check their calculations to ensure they are using the correct values.

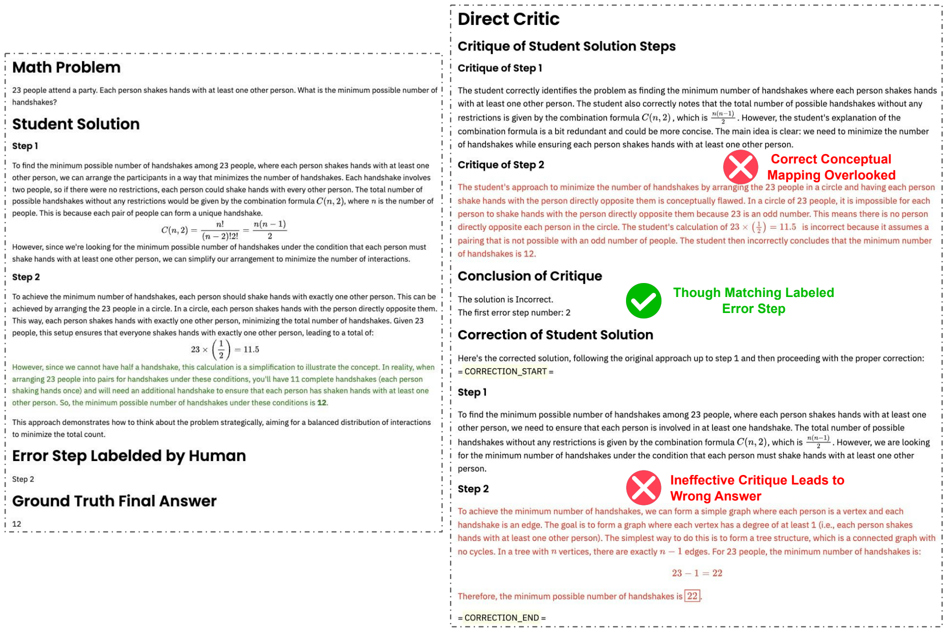
Figure 9: An example of oversimplified bugs injected by Bug-Injection Critic: A variable confusion bug where the wrong price range is used.

G Adaptations to ProcessBench's Evaluation Protocol

In evaluating models' error identification capabilities, we make two adaptations to ProcessBench's original evaluation protocol. These modifications are designed to ensure that models demonstrate genuine understanding of mathematical errors rather than superficial critique.

G.1 Requiring Effective Correction

Our first adaptation stems from the core assumption behind critic and correct tasks: a truly effective critique should not only identify errors but also guide their correction towards an correct answer. Through extensive case studies, we found that models can sometimes correctly identify the error step (matching human annotations) without actually understanding the mathematical mistake. As shown in Figures 10 to 12, these cases highlight that merely matching human-labeled error steps is insufficient for ensuring genuine understanding of mathematical errors.


Correct Conceptual Mapping Overlooked
Though Matching Labeled Error Step
Ineffective Critique Leads to Wrong Answer

Math Problem

23 people attend a party. Each person shakes hands with at least one other person. What is the minimum possible number of handshakes?

Student Solution

Step 1

To find the minimum possible number of handshakes among 23 people, where each person shakes hands with at least one other person, we can arrange the participants in a way that minimizes the number of handshakes. Each handshake involves two people, so if there were no restrictions, each person could shake hands with every other person. The total number of possible handshakes without any restrictions would be given by the combination formula $C(n, 2)$, where n is the number of people. This is because each pair of people can form a unique handshake.

$$C(n,2) = \frac{n!}{(n-2)!2!} = \frac{n(n-1)}{2}$$

However, since we're looking for the minimum possible number of handshakes under the condition that each person must shake hands with at least one other person, we can simplify our arrangement to minimize the number of interactions.

Step 2

To achieve the minimum number of handshakes, each person should shake hands with exactly one other person. This can be achieved by arranging the 23 people in a circle. In a circle, each person shakes hands with the person directly opposite them. This way, each person shakes hands with exactly one other person, minimizing the total number of handshakes. Given 23 people, this setup ensures that everyone shakes hands with exactly one other person, leading to a total of:

$$23 \times \left(\frac{1}{2}\right) = 11.5$$

However, since we cannot have half a handshake, this calculation is a simplification to illustrate the concept. In reality, when arranging 23 people into pairs for handshakes under these conditions, you'll have 11 complete handshakes (each person shaking hands once) and will need an additional handshake to ensure that each person has shaken hands with at least one other person. So, the minimum possible number of handshakes under these conditions is **12**.

This approach demonstrates how to think about the problem strategically, aiming for a balanced distribution of interactions to minimize the total count.

Error Step Labelled by Human

Step 2

Ground Truth Final Answer

12

Direct Critic

Critique of Student Solution Steps

Critique of Step 1

The student correctly identifies the problem as finding the minimum number of handshakes where each person shakes hands with at least one other person. The student also correctly notes that the total number of possible handshakes without any restrictions is given by the combination formula $C(n, 2)$, which is $\frac{n(n-1)}{2}$. However, the student's explanation of the combination formula is a bit redundant and could be more concise. The main idea is clear: we need to minimize the number of handshakes while ensuring each person shakes hands with at least one other person.

Critique of Step 2

The student's approach to minimize the number of handshakes by arranging the 23 people in a circle and having each person shake hands with the person directly opposite them is conceptually flawed. In a circle of 23 people, it is impossible for each person to shake hands with the person directly opposite them because 23 is an odd number. This means there is no person directly opposite each person in the circle. The student's calculation of $23 \times \left(\frac{1}{2}\right) = 11.5$ is incorrect because it assumes a pairing that is not possible with an odd number of people. The student then incorrectly concludes that the minimum number of handshakes is 12.

Conclusion of Critique

The solution is Incorrect.
The first error step number: 2

Correction of Student Solution

Here's the corrected solution, following the original approach up to step 1 and then proceeding with the proper correction:
= CORRECTION_START =

Step 1

To find the minimum possible number of handshakes among 23 people, where each person shakes hands with at least one other person, we need to ensure that each person is involved in at least one handshake. The total number of possible handshakes without any restrictions is given by the combination formula $C(n, 2)$, which is $\frac{n(n-1)}{2}$. However, we are looking for the minimum number of handshakes under the condition that each person must shake hands with at least one other person.

Step 2

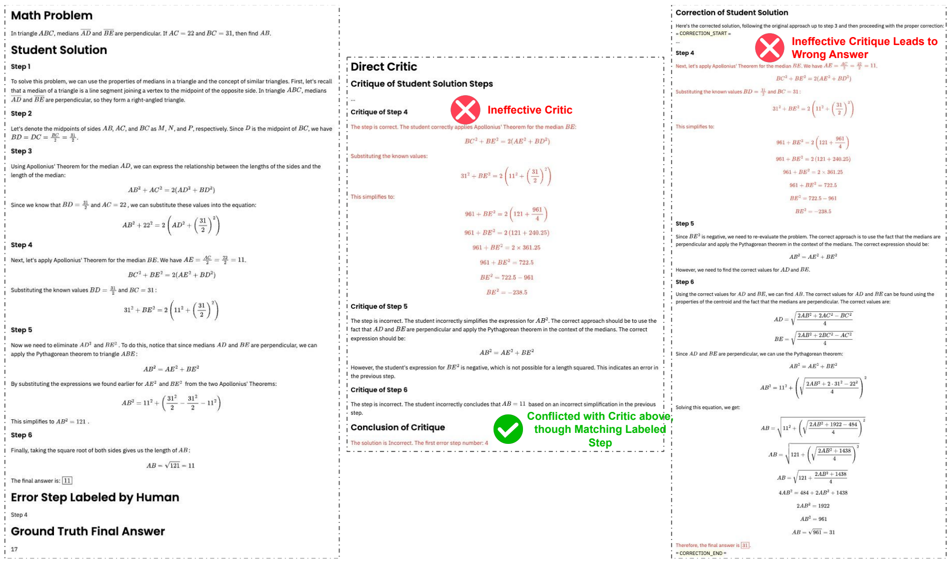
To achieve the minimum number of handshakes, we can form a simple graph where each person is a vertex and each handshake is an edge. The goal is to form a graph where each vertex has a degree of at least 1 (i.e., each person shakes hands with at least one other person). The simplest way to do this is to form a tree structure, which is a connected graph with no cycles. In a tree with n vertices, there are exactly $n-1$ edges. For 23 people, the minimum number of handshakes is:

$$23 - 1 = 22$$

Therefore, the minimum possible number of handshakes is $\boxed{22}$.

= CORRECTION_END =

Figure 10: Although the critic correctly identifies Step 2 as the error step (matching human annotation), it fails to understand the underlying mathematical concept of graph theory, leading to an incorrect correction of 22 handshakes instead of the true answer 12.


Ineffective Critic
Conflicted with Critic above, though Matching Labeled Step
Ineffective Critique Leads to Wrong Answer

Math Problem

In triangle ABC, medians $\overline{AD}$ and $\overline{BE}$ are perpendicular. If $AC = 22$ and $BC = 31$, then find AB.

Student Solution

Step 1

To solve this problem, we can use the properties of medians in a triangle and the concept of similar triangles. First, let's recall that a median of a triangle is a line segment joining a vertex to the midpoint of the opposite side. In triangle ABC, medians $\overline{AD}$ and $\overline{BE}$ are perpendicular, so they form a right-angled triangle.

Step 2

Let's denote the midpoints of sides AB, AC, and BC as M, N, and P, respectively. Since D is the midpoint of BC, we have $BD = DC = \frac{BC}{2} = \frac{31}{2}$.

Step 3

Using Apollonius' Theorem for the median AD, we can express the relationship between the lengths of the sides and the length of the median:

$$AB^2 + AC^2 = 2(AD^2 + BD^2)$$

Since we know that $BD = \frac{31}{2}$ and $AC = 22$, we can substitute these values into the equation:

$$AB^2 + 22^2 = 2\left(AD^2 + \left(\frac{31}{2}\right)^2\right)$$

Step 4

Next, let's apply Apollonius' Theorem for the median BE. We have $AE = \frac{AC}{2} = \frac{22}{2} = 11$.

$$BC^2 + BE^2 = 2(AE^2 + BD^2)$$

Substituting the known values $BD = \frac{31}{2}$ and $BC = 31$:

$$31^2 + BE^2 = 2\left(11^2 + \left(\frac{31}{2}\right)^2\right)$$

Step 5

Now we need to eliminate AD^2 and BE^2. To do this, notice that since medians AD and BE are perpendicular, we can apply the Pythagorean theorem to triangle ABE:

$$AB^2 = AE^2 + BE^2$$

By substituting the expressions we found earlier for AE^2 and BE^2 from the two Apollonius' Theorems:

$$AB^2 = 11^2 + \left(\frac{31^2}{2} - \frac{31^2}{2} - 11^2\right)$$

This simplifies to $AB^2 = 121$.

Step 6

Finally, taking the square root of both sides gives us the length of AB:

$$AB = \sqrt{121} = 11$$

The final answer is: $\boxed{11}$

Error Step Labeled by Human

Step 4

Ground Truth Final Answer

17

Direct Critic

Critique of Student Solution Steps

...

Critique of Step 4

The step is correct. The student correctly applies Apollonius' Theorem for the median BE:

$$BC^2 + BE^2 = 2(AE^2 + BD^2)$$

Substituting the known values:

$$31^2 + BE^2 = 2\left(11^2 + \left(\frac{31}{2}\right)^2\right)$$

This simplifies to:

$$961 + BE^2 = 2\left(121 + \frac{961}{4}\right)$$
$$961 + BE^2 = 2\,(121 + 240.25)$$
$$961 + BE^2 = 2 \times 361.25$$
$$961 + BE^2 = 722.5$$
$$BE^2 = 722.5 - 961$$
$$BE^2 = -238.5$$

Critique of Step 5

The step is incorrect. The student incorrectly simplifies the expression for AB^2. The correct approach should be to use the fact that AD and BE are perpendicular and apply the Pythagorean theorem in the context of the medians. The correct expression should be:

$$AB^2 = AE^2 + BE^2$$

However, the student's expression for BE^2 is negative, which is not possible for a length squared. This indicates an error in the previous step.

Critique of Step 6

The step is incorrect. The student incorrectly concludes that $AB = 11$ based on an incorrect simplification in the previous step.

Conclusion of Critique

The solution is Incorrect. The first error step number: 4

Correction of Student Solution

Here's the corrected solution, following the original approach up to step 3 and then proceeding with the proper correction:
= CORRECTION_START =

...

Step 4

Next, let's apply Apollonius' Theorem for the median BE. We have $AE = \frac{AC}{2} = \frac{22}{2} = 11$.

$$BC^2 + BE^2 = 2(AE^2 + BD^2)$$

Substituting the known values $BD = \frac{31}{2}$ and $BC = 31$:

$$31^2 + BE^2 = 2\left(11^2 + \left(\frac{31}{2}\right)^2\right)$$

This simplifies to:

$$961 + BE^2 = 2\left(121 + \frac{961}{4}\right)$$
$$961 + BE^2 = 2\,(121 + 240.25)$$
$$961 + BE^2 = 2 \times 361.25$$
$$961 + BE^2 = 722.5$$
$$BE^2 = 722.5 - 961$$
$$BE^2 = -238.5$$

Step 5

Since BE^2 is negative, we need to re-evaluate the problem. The correct approach is to use the fact that the medians are perpendicular and apply the Pythagorean theorem in the context of the medians. The correct expression should be:

$$AB^2 = AE^2 + BE^2$$

However, we need to find the correct values for AD and BE.

Step 6

Using the correct values for AD and BE, we can find AB. The correct values for AD and BE can be found using the properties of the centroid and the fact that the medians are perpendicular. The correct values are:

$$AD = \sqrt{\frac{2AB^2 + 2AC^2 - BC^2}{4}}$$
$$BE = \sqrt{\frac{2AB^2 + 2BC^2 - AC^2}{4}}$$

Since AD and BE are perpendicular, we can use the Pythagorean theorem:

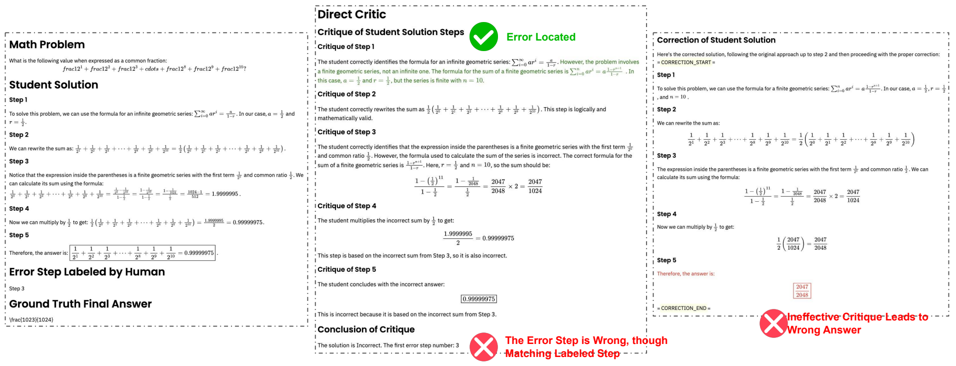
$$AB^2 = AE^2 + BE^2$$
$$AB^2 = 11^2 + \left(\sqrt{\frac{2AB^2 + 2\cdot 31^2 - 22^2}{4}}\right)^2$$

Solving this equation, we get:

$$AB = \sqrt{11^2 + \left(\sqrt{\frac{2AB^2 + 1922 - 484}{4}}\right)^2}$$
$$AB = \sqrt{121 + \left(\sqrt{\frac{2AB^2 + 1438}{4}}\right)^2}$$
$$AB = \sqrt{121 + \frac{2AB^2 + 1438}{4}}$$
$$4AB^2 = 484 + 2AB^2 + 1438$$
$$2AB^2 = 1922$$
$$AB^2 = 961$$
$$AB = \sqrt{961} = 31$$

Therefore, the final answer is $\boxed{31}$.
= CORRECTION_END =

Figure 11: Despite matching the human-labeled error step (Step 4), the critic provides conflicting feedback and fails to recognize the fundamental issue in applying the Pythagorean theorem with perpendicular medians, leading to an incorrect solution.



Math Problem

What is the following value when expressed as a common fraction:

$$frac12^1 + frac12^2 + frac12^3 + cdots + frac12^8 + frac12^9 + frac12^{10}?$$

Student Solution

Step 1

To solve this problem, we can use the formula for an infinite geometric series: $\sum_{i=0}^{\infty} ar^i = \frac{a}{1-r}$. In our case, $a = \frac{1}{2}$ and $r = \frac{1}{2}$.

Step 2

We can rewrite the sum as: $\frac{1}{2^1} + \frac{1}{2^2} + \frac{1}{2^3} + \cdots + \frac{1}{2^8} + \frac{1}{2^9} + \frac{1}{2^{10}} = \frac{1}{2}\left(\frac{1}{2^0} + \frac{1}{2^1} + \frac{1}{2^2} + \cdots + \frac{1}{2^8} + \frac{1}{2^9} + \frac{1}{2^{10}}\right)$.

Step 3

Notice that the expression inside the parentheses is a finite geometric series with the first term $\frac{1}{2^0}$ and common ratio $\frac{1}{2}$. We can calculate its sum using the formula:
$\frac{1}{2^0} + \frac{1}{2^1} + \frac{1}{2^2} + \cdots + \frac{1}{2^8} + \frac{1}{2^9} + \frac{1}{2^{10}} = \frac{\frac{1}{2^0} - \frac{1}{2^{10}}}{1-\frac{1}{2}} = \frac{1-\frac{1}{2^{10}}}{1-\frac{1}{2}} = \frac{1-\frac{1}{1024}}{\frac{1}{2}} = \frac{1024-1}{512} = 1.9999995$.

Step 4

Now we can multiply by $\frac{1}{2}$ to get: $\frac{1}{2}\left(\frac{1}{2^0} + \frac{1}{2^1} + \frac{1}{2^2} + \cdots + \frac{1}{2^8} + \frac{1}{2^9} + \frac{1}{2^{10}}\right) = \frac{1.9999995}{2} = 0.99999975$.

Step 5

Therefore, the answer is: $\boxed{\frac{1}{2^1} + \frac{1}{2^2} + \frac{1}{2^3} + \cdots + \frac{1}{2^8} + \frac{1}{2^9} + \frac{1}{2^{10}} = 0.99999975}$.

Error Step Labeled by Human

Step 3

Ground Truth Final Answer

\frac{1023}{1024}

Direct Critic

Critique of Student Solution Steps

Critique of Step 1

The student correctly identifies the formula for an infinite geometric series: $\sum_{i=0}^{\infty} ar^i = \frac{a}{1-r}$. However, the problem involves a finite geometric series, not an infinite one. The formula for the sum of a finite geometric series is $\sum_{i=0}^{n} ar^i = a\frac{1-r^{n+1}}{1-r}$. In this case, $a = \frac{1}{2}$ and $r = \frac{1}{2}$, but the series is finite with $n = 10$.

Critique of Step 2

The student correctly rewrites the sum as $\frac{1}{2}\left(\frac{1}{2^0} + \frac{1}{2^1} + \frac{1}{2^2} + \cdots + \frac{1}{2^8} + \frac{1}{2^9} + \frac{1}{2^{10}}\right)$. This step is logically and mathematically valid.

Critique of Step 3

The student correctly identifies that the expression inside the parentheses is a finite geometric series with the first term $\frac{1}{2^0}$ and common ratio $\frac{1}{2}$. However, the formula used to calculate the sum of the series is incorrect. The correct formula for the sum of a finite geometric series is $\frac{1-r^{n+1}}{1-r}$. Here, $r = \frac{1}{2}$ and $n = 10$, so the sum should be:

$$\frac{1-\left(\frac{1}{2}\right)^{11}}{1-\frac{1}{2}} = \frac{1-\frac{1}{2048}}{\frac{1}{2}} = \frac{2047}{2048} \times 2 = \frac{2047}{1024}$$

Critique of Step 4

The student multiplies the incorrect sum by $\frac{1}{2}$ to get:

$$\frac{1.9999995}{2} = 0.99999975$$

This step is based on the incorrect sum from Step 3, so it is also incorrect.

Critique of Step 5

The student concludes with the incorrect answer:

$$\boxed{0.99999975}$$

This is incorrect because it is based on the incorrect sum from Step 3.

Conclusion of Critique

The solution is Incorrect. The first error step number: 3

Correction of Student Solution

Here's the corrected solution, following the original approach up to step 2 and then proceeding with the proper correction:
= CORRECTION_START =

Step 1

To solve this problem, we can use the formula for a finite geometric series: $\sum_{i=0}^{n} ar^i = a\frac{1-r^{n+1}}{1-r}$. In our case, $a = \frac{1}{2}, r = \frac{1}{2}$, and $n = 10$.

Step 2

We can rewrite the sum as:

$$\frac{1}{2^1} + \frac{1}{2^2} + \frac{1}{2^3} + \cdots + \frac{1}{2^8} + \frac{1}{2^9} + \frac{1}{2^{10}} = \frac{1}{2}\left(\frac{1}{2^0} + \frac{1}{2^1} + \frac{1}{2^2} + \cdots + \frac{1}{2^8} + \frac{1}{2^9} + \frac{1}{2^{10}}\right)$$

Step 3

The expression inside the parentheses is a finite geometric series with the first term $\frac{1}{2^0}$ and common ratio $\frac{1}{2}$. We can calculate its sum using the formula:

$$\frac{1-\left(\frac{1}{2}\right)^{11}}{1-\frac{1}{2}} = \frac{1-\frac{1}{2048}}{\frac{1}{2}} = \frac{2047}{2048} \times 2 = \frac{2047}{1024}$$

Step 4

Now we can multiply by $\frac{1}{2}$ to get:

$$\frac{1}{2}\left(\frac{2047}{1024}\right) = \frac{2047}{2048}$$

Step 5

Therefore, the answer is:

$$\boxed{\frac{2047}{2048}}$$

= CORRECTION_END =

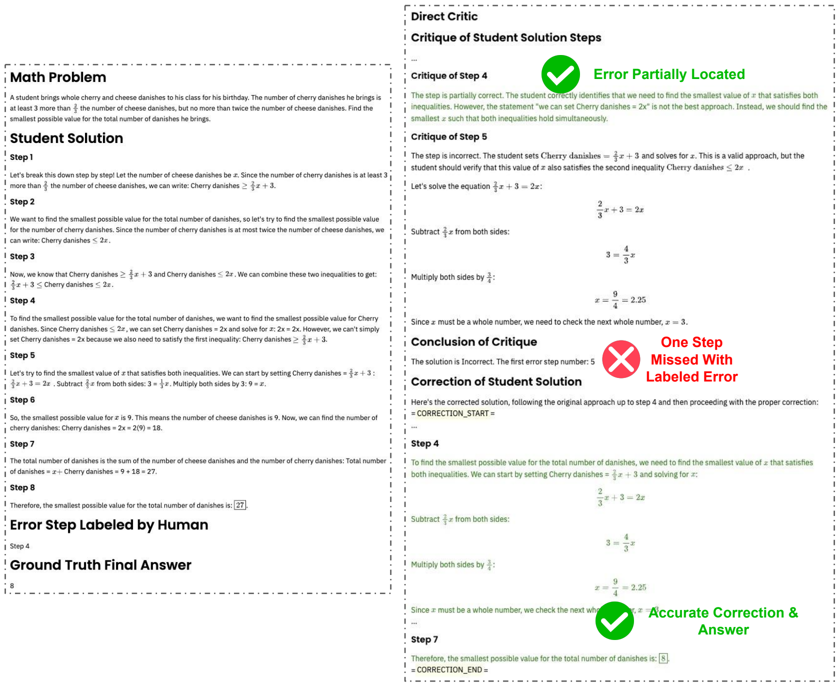
Error Located
The Error Step is Wrong, though Matching Labeled Step
Ineffective Critique Leads to Wrong Answer

Figure 12: The critic matches Step 3 as problematic but misunderstands the key issue in finite geometric series calculation, resulting in an incorrect final value of 2047/2048.

Therefore, we augment ProcessBench's protocol by requiring that models must not only identify the correct error step but also provide correction that leads to a mathematically valid solution. This stricter requirement helps ensure that models demonstrate genuine understanding of the mathematical concepts and errors involved.

G.2 Allowing Step-Level Flexibility

Our second adaptation addresses an inherent ambiguity in error identification: in many cases, mathematical errors can reasonably be attributed to multiple consecutive steps. Through our analysis, we found numerous instances where the exact "error step" is debatable, with both the preceding and following steps being valid points of identification. As shown in Figures 13 to 15, these cases illustrate how mathematical errors often span multiple steps, making strict step-level matching overly rigid for meaningful evaluation..



Math Problem

A student brings whole cherry and cheese danishes to his class for his birthday. The number of cherry danishes he brings is at least 3 more than $\frac{2}{3}$ the number of cheese danishes, but no more than twice the number of cheese danishes. Find the smallest possible value for the total number of danishes he brings.

Student Solution

Step 1

Let's break this down step by step! Let the number of cheese danishes be x. Since the number of cherry danishes is at least 3 more than $\frac{2}{3}$ the number of cheese danishes, we can write: Cherry danishes $\geq \frac{2}{3}x + 3$.

Step 2

We want to find the smallest possible value for the total number of danishes, so let's try to find the smallest possible value for the number of cherry danishes. Since the number of cherry danishes is at most twice the number of cheese danishes, we can write: Cherry danishes $\leq 2x$.

Step 3

Now, we know that Cherry danishes $\geq \frac{2}{3}x + 3$ and Cherry danishes $\leq 2x$. We can combine these two inequalities to get: $\frac{2}{3}x + 3 \leq$ Cherry danishes $\leq 2x$.

Step 4

To find the smallest possible value for the total number of danishes, we want to find the smallest possible value for Cherry danishes. Since Cherry danishes $\leq 2x$, we can set Cherry danishes = 2x and solve for x: 2x = 2x. However, we can't simply set Cherry danishes = 2x because we also need to satisfy the first inequality: Cherry danishes $\geq \frac{2}{3}x + 3$.

Step 5

Let's try to find the smallest value of x that satisfies both inequalities. We can start by setting Cherry danishes = $\frac{2}{3}x + 3$: $\frac{2}{3}x + 3 = 2x$. Subtract $\frac{2}{3}x$ from both sides: 3 = $\frac{1}{3}x$. Multiply both sides by 3: 9 = x.

Step 6

So, the smallest possible value for x is 9. This means the number of cheese danishes is 9. Now, we can find the number of cherry danishes: Cherry danishes = 2x = 2(9) = 18.

Step 7

The total number of danishes is the sum of the number of cheese danishes and the number of cherry danishes: Total number of danishes = $x+$ Cherry danishes = 9 + 18 = 27.

Step 8

Therefore, the smallest possible value for the total number of danishes is: $\boxed{27}$.

Error Step Labeled by Human

Step 4

Ground Truth Final Answer

8

Direct Critic

Critique of Student Solution Steps

...

Critique of Step 4

The step is partially correct. The student correctly identifies that we need to find the smallest value of x that satisfies both inequalities. However, the statement "we can set Cherry danishes = 2x" is not the best approach. Instead, we should find the smallest x such that both inequalities hold simultaneously.

Critique of Step 5

The step is incorrect. The student sets $\text{Cherry danishes} = \frac{2}{3}x + 3$ and solves for x. This is a valid approach, but the student should verify that this value of x also satisfies the second inequality $\text{Cherry danishes} \leq 2x$.

Let's solve the equation $\frac{2}{3}x + 3 = 2x$:

$$\frac{2}{3}x + 3 = 2x$$

Subtract $\frac{2}{3}x$ from both sides:

$$3 = \frac{4}{3}x$$

Multiply both sides by $\frac{3}{4}$:

$$x = \frac{9}{4} = 2.25$$

Since x must be a whole number, we need to check the next whole number, $x = 3$.

Conclusion of Critique

The solution is Incorrect. The first error step number: 5

Correction of Student Solution

Here's the corrected solution, following the original approach up to step 4 and then proceeding with the proper correction:
= CORRECTION_START =

...

Step 4

To find the smallest possible value for the total number of danishes, we need to find the smallest value of x that satisfies both inequalities. We can start by setting Cherry danishes = $\frac{2}{3}x + 3$ and solving for x:

$$\frac{2}{3}x + 3 = 2x$$

Subtract $\frac{2}{3}x$ from both sides:

$$3 = \frac{4}{3}x$$

Multiply both sides by $\frac{3}{4}$:

$$x = \frac{9}{4} = 2.25$$

Since x must be a whole number, we check the next wh r, $x = 3$

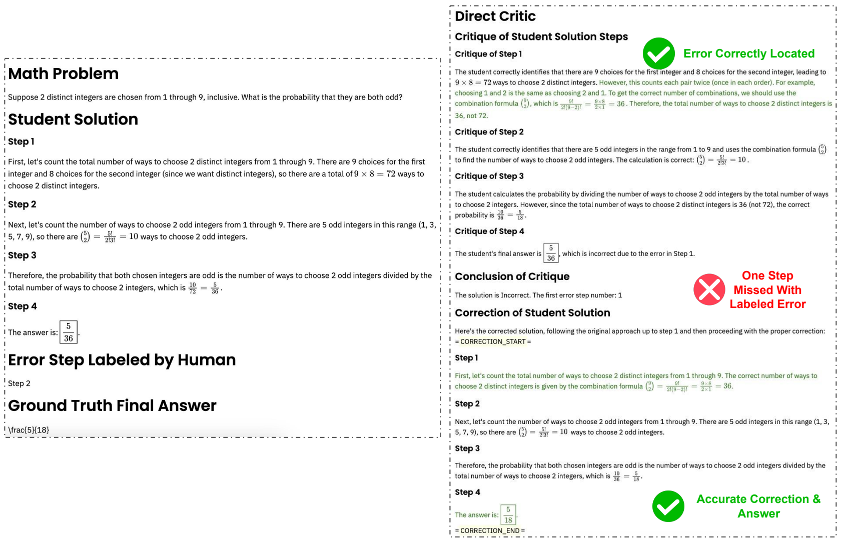
...

Step 7

Therefore, the smallest possible value for the total number of danishes is: $\boxed{8}$.
= CORRECTION_END =

Error Partially Located
One Step Missed With Labeled Error
Accurate Correction & Answer

Figure 13: In this cherry-and-cheese danishes problem, while the human annotator labels Step 4 as the error, the true conceptual error begins in Step 5 where the student miscalculates the solution. The model still achieves correct final answer despite identifying a different step.


Error Correctly Located
One Step Missed With Labeled Error
Accurate Correction & Answer

Math Problem

Suppose 2 distinct integers are chosen from 1 through 9, inclusive. What is the probability that they are both odd?

Student Solution

Step 1

First, let's count the total number of ways to choose 2 distinct integers from 1 through 9. There are 9 choices for the first integer and 8 choices for the second integer (since we want distinct integers), so there are a total of $9 \times 8 = 72$ ways to choose 2 distinct integers.

Step 2

Next, let's count the number of ways to choose 2 odd integers from 1 through 9. There are 5 odd integers in this range (1, 3, 5, 7, 9), so there are $\binom{5}{2} = \frac{5!}{2!3!} = 10$ ways to choose 2 odd integers.

Step 3

Therefore, the probability that both chosen integers are odd is the number of ways to choose 2 odd integers divided by the total number of ways to choose 2 integers, which is $\frac{10}{72} = \frac{5}{36}$.

Step 4

The answer is: $\boxed{\frac{5}{36}}$.

Error Step Labeled by Human

Step 2

Ground Truth Final Answer

\frac{5}{18}

Direct Critic

Critique of Student Solution Steps

Critique of Step 1

The student correctly identifies that there are 9 choices for the first integer and 8 choices for the second integer, leading to $9 \times 8 = 72$ ways to choose 2 distinct integers. However, this counts each pair twice (once in each order). For example, choosing 1 and 2 is the same as choosing 2 and 1. To get the correct number of combinations, we should use the combination formula $\binom{9}{2}$, which is $\frac{9!}{2!(9-2)!} = \frac{9\times8}{2\times1} = 36$. Therefore, the total number of ways to choose 2 distinct integers is 36, not 72.

Critique of Step 2

The student correctly identifies that there are 5 odd integers in the range from 1 to 9 and uses the combination formula $\binom{5}{2}$ to find the number of ways to choose 2 odd integers. The calculation is correct: $\binom{5}{2} = \frac{5!}{2!3!} = 10$.

Critique of Step 3

The student calculates the probability by dividing the number of ways to choose 2 odd integers by the total number of ways to choose 2 integers. However, since the total number of ways to choose 2 distinct integers is 36 (not 72), the correct probability is $\frac{10}{36} = \frac{5}{18}$.

Critique of Step 4

The student's final answer is $\boxed{\frac{5}{36}}$, which is incorrect due to the error in Step 1.

Conclusion of Critique

The solution is Incorrect. The first error step number: 1

Correction of Student Solution

Here's the corrected solution, following the original approach up to step 1 and then proceeding with the proper correction:
= CORRECTION_START =

Step 1

First, let's count the total number of ways to choose 2 distinct integers from 1 through 9. The correct number of ways to choose 2 distinct integers is given by the combination formula $\binom{9}{2} = \frac{9!}{2!(9-2)!} = \frac{9\times8}{2\times1} = 36.$

Step 2

Next, let's count the number of ways to choose 2 odd integers from 1 through 9. There are 5 odd integers in this range (1, 3, 5, 7, 9), so there are $\binom{5}{2} = \frac{5!}{2!3!} = 10$ ways to choose 2 odd integers.

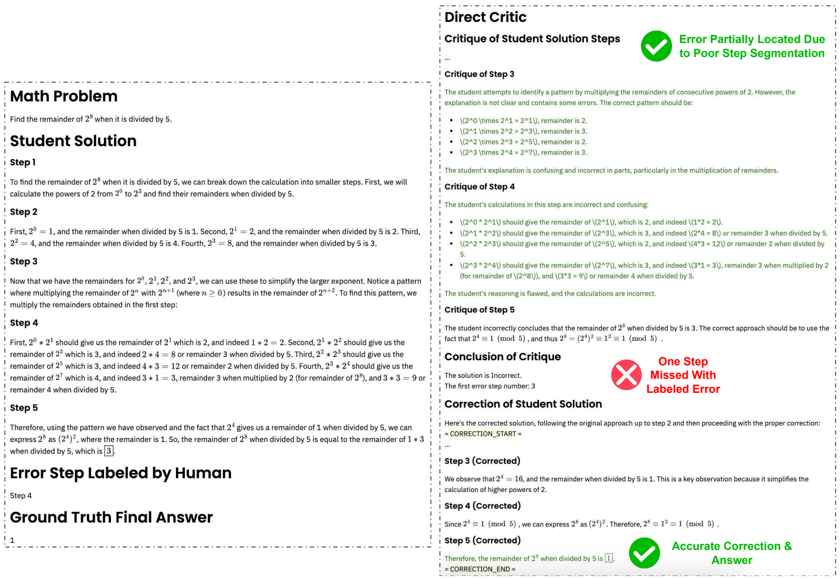
Step 3

Therefore, the probability that both chosen integers are odd is the number of ways to choose 2 odd integers divided by the total number of ways to choose 2 integers, which is $\frac{10}{36} = \frac{5}{18}$.

Step 4

The answer is: $\boxed{\frac{5}{18}}$.
= CORRECTION_END =

Figure 14: In this probability problem, while the annotator marks Step 2 as the error, the fundamental misconception in Step 1 (overcounting combinations) directly leads to the final incorrect probability.


Error Partially Located Due to Poor Step Segmentation
One Step Missed With Labeled Error
Accurate Correction & Answer

Math Problem

Find the remainder of 2^8 when it is divided by 5.

Student Solution

Step 1

To find the remainder of 2^8 when it is divided by 5, we can break down the calculation into smaller steps. First, we will calculate the powers of 2 from 2^0 to 2^3 and find their remainders when divided by 5.

Step 2

First, $2^0 = 1$, and the remainder when divided by 5 is 1. Second, $2^1 = 2$, and the remainder when divided by 5 is 2. Third, $2^2 = 4$, and the remainder when divided by 5 is 4. Fourth, $2^3 = 8$, and the remainder when divided by 5 is 3.

Step 3

Now that we have the remainders for 2^0, 2^1, 2^2, and 2^3, we can use these to simplify the larger exponent. Notice a pattern where multiplying the remainder of 2^n with 2^{n+1} (where $n \geq 0$) results in the remainder of 2^{n+2}. To find this pattern, we multiply the remainders obtained in the first step:

Step 4

First, $2^0 * 2^1$ should give us the remainder of 2^1 which is 2, and indeed $1 * 2 = 2$. Second, $2^1 * 2^2$ should give us the remainder of 2^3 which is 3, and indeed $2 * 4 = 8$ or remainder 3 when divided by 5. Third, $2^2 * 2^3$ should give us the remainder of 2^5 which is 3, and indeed $4 * 3 = 12$ or remainder 2 when divided by 5. Fourth, $2^3 * 2^4$ should give us the remainder of 2^7 which is 4, and indeed $3 * 1 = 3$, remainder 3 when multiplied by 2 (for remainder of 2^8), and $3 * 3 = 9$ or remainder 4 when divided by 5.

Step 5

Therefore, using the pattern we have observed and the fact that 2^4 gives us a remainder of 1 when divided by 5, we can express 2^8 as $(2^4)^2$, where the remainder is 1. So, the remainder of 2^8 when divided by 5 is equal to the remainder of $1 * 3$ when divided by 5, which is $\boxed{3}$.

Error Step Labeled by Human

Step 4

Ground Truth Final Answer

1

Direct Critic

Critique of Student Solution Steps

...

Critique of Step 3

The student attempts to identify a pattern by multiplying the remainders of consecutive powers of 2. However, the explanation is not clear and contains some errors. The correct pattern should be:

- \(2^0 \times 2^1 = 2^1\), remainder is 2.
- \(2^1 \times 2^2 = 2^3\), remainder is 3.
- \(2^2 \times 2^3 = 2^5\), remainder is 2.
- \(2^3 \times 2^4 = 2^7\), remainder is 3.

The student's explanation is confusing and incorrect in parts, particularly in the multiplication of remainders.

Critique of Step 4

The student's calculations in this step are incorrect and confusing:

- \(2^0 * 2^1\) should give the remainder of \(2^1\), which is 2, and indeed \(1*2 = 2\).
- \(2^1 * 2^2\) should give the remainder of \(2^3\), which is 3, and indeed \(2*4 = 8\) or remainder 3 when divided by 5.
- \(2^2 * 2^3\) should give the remainder of \(2^5\), which is 2, and indeed \(4*3 = 12\) or remainder 2 when divided by 5.
- \(2^3 * 2^4\) should give the remainder of \(2^7\), which is 3, and indeed \(3*1 = 3\), remainder 3 when multiplied by 2 (for remainder of \(2^8\)), and \(3*3 = 9\) or remainder 4 when divided by 5.

The student's reasoning is flawed, and the calculations are incorrect.

Critique of Step 5

The student incorrectly concludes that the remainder of 2^8 when divided by 5 is 3. The correct approach should be to use the fact that $2^4 \equiv 1 \pmod{5}$, and thus $2^8 = (2^4)^2 \equiv 1^2 \equiv 1 \pmod{5}$.

Conclusion of Critique

The solution is Incorrect.
The first error step number: 3

Correction of Student Solution

Here's the corrected solution, following the original approach up to step 2 and then proceeding with the proper correction:
= CORRECTION_START =
...

Step 3 (Corrected)

We observe that $2^4 = 16$, and the remainder when divided by 5 is 1. This is a key observation because it simplifies the calculation of higher powers of 2.

Step 4 (Corrected)

Since $2^4 \equiv 1 \pmod{5}$, we can express 2^8 as $(2^4)^2$. Therefore, $2^8 \equiv 1^2 \equiv 1 \pmod{5}$.

Step 5 (Corrected)

Therefore, the remainder of 2^8 when divided by 5 is $\boxed{1}$.
= CORRECTION_END =

Figure 15: In this remainder calculation problem, the error could be attributed to either Step 3 (pattern identification) or Step 4 (pattern application), as they form a continuous chain of incorrect reasoning.

To account for this ambiguity, we introduce a ±1 step tolerance in matching model predictions with human annotations. This modification better reflects the reality of mathematical error analysis while still maintaining rigor in evaluation.

These adaptations result in a more meaningful evaluation protocol that better captures models' true understanding of mathematical errors and their ability to guide effective corrections.

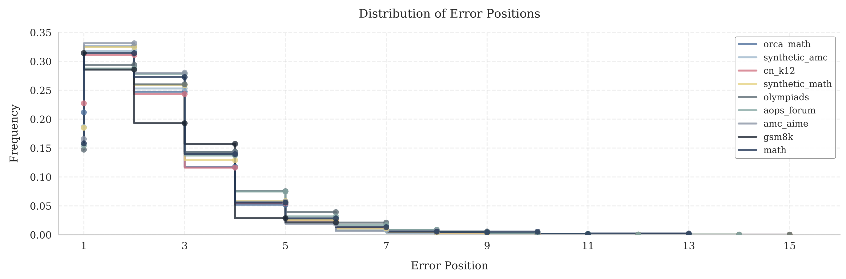
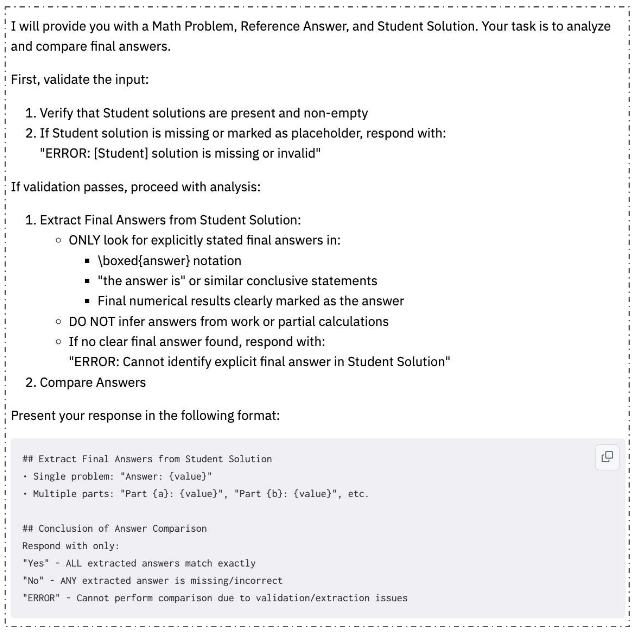


Figure 16: Distribution of first error positions identified by our self-critic across different mathematical domains.

H Distribution of First Error Step identified by Self-Critic

I Classify Solutions into Correct and Incorrect

Again we use Qwen2.5-72B-Instruct itself to classify solutions into correct and incorrect ones. We present the system prompt in the following Figure 17:



I will provide you with a Math Problem, Reference Answer, and Student Solution. Your task is to analyze and compare final answers.

First, validate the input:

1. Verify that Student solutions are present and non-empty
2. If Student solution is missing or marked as placeholder, respond with:
   "ERROR: [Student] solution is missing or invalid"

If validation passes, proceed with analysis:

1. Extract Final Answers from Student Solution:
   - ONLY look for explicitly stated final answers in:
     - \boxed{answer} notation
     - "the answer is" or similar conclusive statements
     - Final numerical results clearly marked as the answer
   - DO NOT infer answers from work or partial calculations
   - If no clear final answer found, respond with:
     "ERROR: Cannot identify explicit final answer in Student Solution"
2. Compare Answers

Present your response in the following format:

```
## Extract Final Answers from Student Solution
• Single problem: "Answer: {value}"
• Multiple parts: "Part {a}: {value}", "Part {b}: {value}", etc.

## Conclusion of Answer Comparison
Respond with only:
"Yes" - ALL extracted answers match exactly
"No" - ANY extracted answer is missing/incorrect
"ERROR" - Cannot perform comparison due to validation/extraction issues
```

Figure 17: System Prompt to classify solutions into correct and incorrect ones.

J Self-Training Details

Here we present the detailed configuration for self-training of Qwen2.5-72B-Instruct. We utilize open-instruct (Wang et al., 2023) for our continued supervised fine-tuning implementation. The training was conducted on 4 servers, each equipped with 8 NVIDIA A100 GPUs (32 GPUs in total), with a total training time of several hours[2].

The key hyper-parameters for training are as follows:

- Batch size: 256

[2]The exact training time may vary depending on the specific hardware configuration and system load.

- Learning rate: 5e-6
- Number of training epochs: 1
- Warmup ratio: 0.03
- Model parallel size: 8
- Total GPUs: 32 (4 servers × 8 A100 GPUs)

For reproducibility, we use gradient checkpointing and mixed-precision training (FP16) to optimize memory usage. The training was performed using DeepSpeed ZeRO-3 for efficient distributed training.