arXiv:2411.00855v1 [cs.LG] 30 Oct 2024

Vision-Language Models Can Self-Improve Reasoning via Reflection

Kanzhi Cheng[1] † **Yantao Li**[1] † **Fangzhi Xu**[2] **Jianbing Zhang**[1]
Hao Zhou[2 3] **Yang Liu**[2 3]
[1]National Key Laboratory for Novel Software Technology, Nanjing University
[2]Shanghai AI Lab [3]Institute for AI Industry Research (AIR), Tsinghua University
`{chengkz,li_yantao}@smail.nju.edu.cn` `fangzhixu98@gmail.com`
`zjb@nju.edu.cn` `zhouhao@air.tsinghua.edu.cn` `liuyang2011@tsinghua.edu.cn`

Abstract

Chain-of-thought (CoT) has proven to improve the reasoning capability of large language models (LLMs). However, due to the complexity of multimodal scenarios and the difficulty in collecting high-quality CoT data, CoT reasoning in multimodal LLMs has been largely overlooked. To this end, we propose a simple yet effective self-training framework, R^3V, which iteratively enhances the model's **V**ision-language **R**easoning by **R**eflecting on CoT **R**ationales. Our framework consists of two interleaved parts: (1) iteratively bootstrapping positive and negative solutions for reasoning datasets, and (2) reflection on rationale for learning from mistakes. Specifically, we introduce the self-refine and self-select losses, enabling the model to refine flawed rationale and derive the correct answer by comparing rationale candidates. Experiments on a wide range of vision-language tasks show that R^3V consistently improves multimodal LLM reasoning, achieving a relative improvement of 23% to 60% over GPT-distilled baselines. Additionally, our approach supports self-reflection on generated solutions, further boosting performance through test-time computation. *

1 Introduction

Humans often rely on intuitive Chain-of-Thought (CoT) to perform complex reasoning (Ericsson and Simon, 1980). Previous studies have shown that this CoT capacity also emerges in Large Language Models (LLMs) (Wei et al., 2022). Through simple prompting or fine-tuning (Cobbe et al., 2021; Kojima et al., 2022; Hsieh et al., 2023), CoT enhances the reasoning performance of LLMs while providing insights into their decision-making process. Recently, OpenAI o1 further advances reasoning by producing long internal CoT sequences, taking LLMs intelligence to a new level.

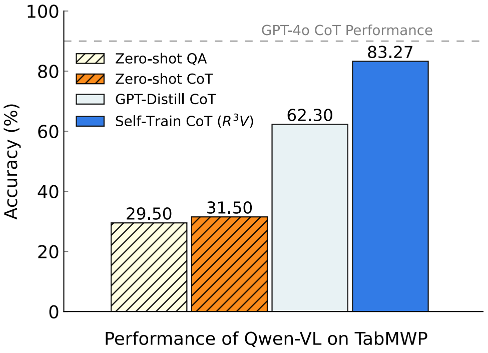


Figure 1: Results of Qwen-VL on TabMWP, a visual mathematical reasoning dataset. Qwen-VL exhibits weak zero-shot CoT reasoning performance, while our R^3V iteratively self-improves, surpassing the GPT-distilled baseline by a large margin.

While CoT reasoning has significantly advanced LLMs in textual domains, extending CoT to multimodal settings remains an open problem. Unlike the abundant, unsupervised text-based CoT in pre-training corpora (Kojima et al., 2022; Wei et al., 2022), multimodal CoT resources are scarce in the text-dominated internet collections (Dai et al., 2023), hindering the full realization of Multimodal LLMs' (MLLMs) reasoning potential.

Recent studies show that open-sourced MLLMs struggle to integrate visual cues into their reasoning process, resulting in weak CoT performance (Zhang et al., 2024a; Shi et al., 2024). Consistent with our observations in Figure 1, CoT prompting provides minimal gains over direct prediction (Chen et al., 2024a) and falls far behind GPT-4o. One potential solution is to construct multimodal CoT annotations for post-training; however, manual annotation is prohibitively expensive and hard to scale. This raises our first research question: *can MLLMs self-improve the reasoning capabilities through bootstrapping on CoT samples?*

Orthogonal to fine-tuning on curated CoT annotations, relying solely on positive samples can lead to suboptimal policy due to insufficient explo-

---

[†] Equal contribution.
[*] Our code is available at `https://github.com/njucckevin/MM-Self-Improve`.

ration of reasoning paths. Inspired by human thinking, another promising direction involves learning from trial-and-errors (Yuan et al., 2024; Song et al., 2024), where mistakes are not failures but key opportunities to enhance reasoning. A few multimodal approaches use corrupted prompts to create negative samples for preference learning, aiming to improve image comprehension (Wang et al., 2023; Deng et al., 2024). However, these methods fail to generate reasoning-aligned positive and negative CoT solutions, making them unsuitable for complex multimodal reasoning tasks. Thus, it remains unaddressed: *how can MLLMs efficiently learn from mistakes to improve their reasoning skills?*

To address the above two questions, this paper proposes R^3V, a self-training framework that enables the model to **R**eflect on bootstrapped CoT **R**ationales, thereby strengthening its **V**ision-Language **R**easoning. Firstly, we leverage MLLM's pre-existing but weak CoT ability to bootstrap both rationales and answers for a given question, enabling the collection of a large number of positive and negative solutions based on answer correctness. Secondly, we introduce a reflection mechanism on negative solutions to help the model learn from mistakes. Specifically, we design self-refine and self-select losses that guide the model to correct flawed rationales and derive the correct answer by comparing rationale candidates, respectively. The above synergistic process can be repeated, with improved samples boosting MLLM's reasoning and the enhanced model further improving rationale generation. Additionally, through self-select training, our model can derive the superior solution from multiple samples, further boosting performance via test-time computation.

We conduct experiments across a wide range of multimodal reasoning benchmarks, including charts, geometry, commonsense, science, mathematics, etc. R^3V progressively enhances the reasoning ability of MLLMs, delivering a 23%-60% relative accuracy improvement compared to GPT distillation, and consistently outperforming the strong self-training baseline, STaR (Zelikman et al., 2022). Moreover, our test-time selection is robust and effective, consistently surpassing Pass@1 and majority voting, even in OOD scenarios.

Our main contributions are as follows:

- We introduce an iterative self-training framework R^3V that leverages CoT bootstrapped by MLLM itself for self-improvement. To our knowledge, this is the first attempt to apply self-training in vision-language reasoning.
- We propose learning from mistakes through self-reflection, with support for test-time computation to further improve reasoning performance.
- We perform extensive evaluations across 6 different multimodal domains to validate the effectiveness of R^3V. Our analysis reveals the key factors driving the success of multimodal self-training.

2 Related Work

Vision-Language Reasoning Beyond the extensively studied unimodal reasoning (Cobbe et al., 2021; Sun et al., 2023), multimodal reasoning has recently attracted significant interest as an essential part of human intelligence (Yue et al., 2024; Lu et al., 2023). Although MLLMs perform well on general vision-language benchmarks (Liu et al., 2024; Chen et al., 2024b), integrating visual cues into the reasoning process poses unique challenges, especially for open-source models (Zhang et al., 2024a; Chen et al., 2024a). Several studies have explored using rationale datasets to fine-tune models and enhance visual-language reasoning capabilities. For example, (Gao et al., 2023; Zhang et al., 2024b) augmented existing mathematical datasets with rationales using GPT distillation, while (Yang et al., 2024) enhanced performance through manually collected CoT annotations. In this work, we advocate for MLLMs to self-improve, reducing reliance on resource-heavy rationale annotations.

Self-Training Methods Self-training helps the model learn from its own generated outputs, reducing the need for labor-intensive human annotations (Yuan et al., 2024; Chen et al., 2024c). Prior works have focused on enhancing the reasoning capacity of LLM. The typical approach involves sampling multiple rationales and filtering positive and negative solutions based on the answers. The LLM is then fine-tuned on the positive samples (Zelikman et al., 2022; Hosseini et al., 2024; Yuan et al., 2023) or improved using preference learning (Wang et al., 2024b; Mitra et al., 2024), such as DPO (Rafailov et al., 2024). Recent advances have also extended self-training to agents (Song et al., 2024) and neural symbolic (Xu et al., 2024) scenarios. In this paper, we pioneer the exploration of self-training in vision-language reasoning, investigate the failure of DPO in multimodal settings, and address these challenges with our R^3V framework.

3 Methodology

Our self-training framework consists of two alternating components: (1) bootstrapping a large number of positive and negative CoT solutions for multimodal questions (Section 3.1); (2) using the above-sampled solutions to reflect on rationales and learn from mistakes (Section 3.2). This iterative process turns the MLLM from weak to strong. The overall framework is illustrated in Figure 2.

3.1 Preliminaries

In visual-language reasoning, given an image I and a question x, a multimodal large language model is required to integrate information from both the image and the question for reasoning, generating a CoT rationale r and then deriving the final answer a. However, due to the difficulty in collecting high-quality rationale data, constructing large-scale (I, x, r, a) pairs presents significant challenges. This hinders the enhancement of MLLM reasoning capacities through fine-tuning. To overcome this limitation, we propose leveraging the MLLM's pre-existing but weak CoT capability to iteratively augment (I, x, r, a) pairs from the widely available vision question answering data (I, x, a), enabling the model to self-improve.

Following STaR (Zelikman et al., 2022), the MLLM self-training process involves iteratively fine-tuning on its self-generated rationale data. In each iteration t, given a question x from training set $\mathcal{D} = \{(I, x, \hat{a})\}$, the MLLM $\mathcal{M}$ first generate a CoT rationale r along with an answer a , formulated as $\{(r_i, a_i)\}_{i=1}^{|\mathcal{D}|}$. These intermediate outputs are then combined with the original training set, resulting in an augmented dataset that includes rationales:

$$\mathcal{D}_r = \{(I_i, x_i, r_i, a_i)\}_{i=1}^{|\mathcal{D}|} \tag{1}$$

Assuming that rationales leading to correct answers are of higher quality compared to those that do not, we can divide D_r into positive and negative sample sets based on the correctness of the answers:

$$\mathcal{D}_r^+ = \{(I_i, x_i, r_i, a_i) \mid a_i = \hat{a}_i\}_{i=1}^{|\mathcal{D}|} \tag{2}$$

$$\mathcal{D}_r^- = \{(I_i, x_i, r_i, a_i) \mid a_i \neq \hat{a}_i\}_{i=1}^{|\mathcal{D}|} \tag{3}$$

We then fine-tune model $\mathcal{M}$ on the filtered positive CoT samples $\mathcal{D}_r^+$ using supervised fine-tuning (SFT) with a negative log-likelihood objective:

$$\mathcal{L}_{SFT} = -\sum_{(I,x,y)\sim\mathcal{D}_t} \log \mathcal{M}(y \mid x, I), \tag{4}$$

where the $y = (r, a)$ is the solution generated by the model. We continue repeating the above process, generating new rationales with the newly fine-tuned model, until performance plateaus.

3.2 *R³V*: Reflection on Rationales

The above self-improvement process strengthens the model using positive solutions, while negative ones are typically discarded. However, negative samples comprise a large portion of the sampled solutions and offer valuable insights for further model enhancement (An et al., 2023; Hosseini et al., 2024). In our preliminary experiments, we found that the noisy nature of CoT in multimodal scenarios leads to suboptimal performance when using DPO (Rafailov et al., 2024). Inspired by the error-driven learning of humans, we introduce reflection on rationales, teaching the model to correct its own mistakes and reflect on multiple reasoning paths to identify the correct solution. Specifically, we propose additional self-refine (Section 3.2.1) and self-select (Section 3.2.2) losses for multitask learning. Our framework harnesses the continuous production of positive and negative samples in self-training, offering a robust and effective solution for learning from mistakes. Appendix E provides examples of different components in *R³V*.

3.2.1 Self-Refine

Upon failing to solve a problem, human students will analyze the errors in their solutions and reflect on how to correct them. Inspired by this, we designed the self-refine mechanism to encourage the model to correct flaws in its generated solutions. Multiple positive and negative solutions sampled during self-training can be viewed as the model's repeated reasoning on the same problem, making them well-suited for self-refine training. Specifically, we construct dataset for self-refine as follows:

$$\mathcal{D}_{REF} = \{(I_i, x_i, y_i^+, y_i^-) \mid \exists\, y_i^+, y_i^-\}_{i=1}^{|\mathcal{D}|}, \tag{5}$$

where y_i^+ and y_i^- are positive and negative samples obtained from preceding iterations. Next, the self-refine loss is employed to guide the model in correcting errors in its self-generated answers:

$$\mathcal{L}_{REF} = -\sum_{(I,x,y^+,y^-)\sim\mathcal{D}_{REF}} \log \mathcal{M}(y^+ \mid y^-, x, I) \tag{6}$$

Throughout the self-training iterations, samples for self-refine are continuously updated to incorporate higher-quality positive solutions and harder negative solutions.

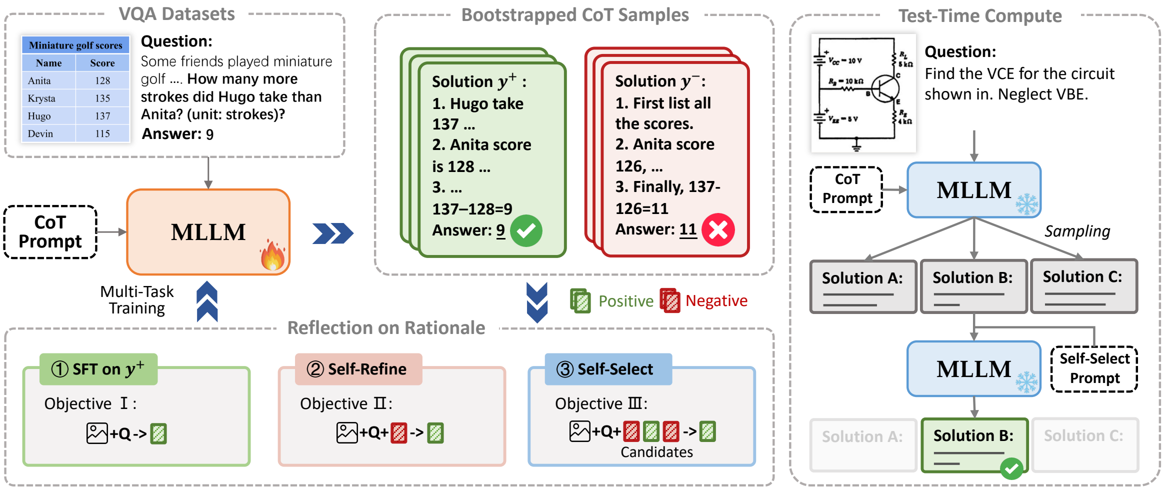


Figure 2: Overview of our multimodal self-training framework of R^3V. It boosts vision-language reasoning by iteratively reflecting on bootstrapped CoT rationales and enables self-reflection through test-time computing.

3.2.2 Self-Select

Our early explorations reveal a key challenge in MLLM reasoning: current MLLMs frequently make simple errors such as misreading chart numbers or calculation mistakes, however, the autoregressive model has no mechanism to correct them, leading to suboptimal performance. In contrast, human reasoners implicitly simulate multiple reasoning paths, check for errors, and select the best one. Inspired by this, we introduce the self-selection mechanism, guiding MLLMs to derive the correct answer from multiple candidate solutions.

Given a set of sampled rationales, the model is required to analyze their differences and finally select the correct answer. Specifically, we construct the self-select dataset as follows:

$$\mathcal{D}_{SEL} = \{(I_i, x_i, \hat{a}_i, \mathcal{C}_i) \mid \exists \mathcal{C}_i\}_{i=1}^{|\mathcal{D}|}, \qquad (7)$$

where $\hat{a}$ is the ground truth and $\mathcal{C}_i = (y_i^1, y_i^2, ..., y_i^N)$ is a set of N sampled rationale-answer pair. In our experiments, N is set to 3 by default. We ensure that the candidate set $\mathcal{C}$ contains at least one positive solution y^+, allowing the model to select the final correct answer. Then, the self-select loss is defined as:

$$\mathcal{L}_{SEL} = -\sum_{(I,x,\hat{a},\mathcal{C})\sim\mathcal{D}_{SEL}} \log \mathcal{M}(\hat{a}|x, I, \mathcal{C}) \qquad (8)$$

Finally, our framework combines three loss functions in a multi-task training setup to enhance MLLM reasoning (see algorithm in Appendix D):

$$\mathcal{L}_{R^3V} = \mathcal{L}_{SFT} + \mathcal{L}_{REF} + \mathcal{L}_{SEL} \qquad (9)$$

From another perspective, we argue that this multi-task training enables MLLMs to learn reasoning from easy to hard: selecting the correct solution from multiple candidates, refining existing rationales, and eventually generating solutions directly.

3.2.3 Test-Time Selection

Through self-select training, our framework enables MLLMs to reflect on their self-generated solutions and select the final answer from multiple reasoning paths. During inference, given a question x and corresponding image I, we first sample multiple reasoning solutions to form the candidate set $\mathcal{C}$. Next, the MLLM is prompted to select the best answer from these candidate solutions: $a = \mathcal{M}(x, I, \mathcal{C})$.

Test-time selection offers a novel approach for MLLMs to tackle complex multimodal reasoning. Instead of directly generating an answer, the model applies an elimination method by comparing different reasoning paths and checking for errors (e.g., visual recognition, calculation, or reasoning mistakes) to identify the most likely correct solution. In this way, our approach further boosts reasoning performance through test-time computation.

4 Experiments

In our experiments, we focus on a diverse and comprehensive set of vision-language reasoning tasks to demonstrate the effectiveness of R^3V. We begin by outlining the benchmarks (Section 4.1) and experimental setup (Section 4.2), followed by the main results of R^3V on six widely used datasets (Section 4.3). We also evaluated the improvements

Methods	Is CoT?	Logical and Numerical reasoning			Agentic	Geometry	Multi-Domain	Avg
		TabMWP	ChartQA	CLEVR-Math	MiniWob	GeoQA	M^{3}CoT	
GPT-4o	✓	94.47	67.00	70.60	98.50	55.17	65.85	70.62
Qwen-VL								
Zero-shot QA	✗	29.50	38.56	17.32	-	15.14	31.28	26.36
Zero-shot CoT	✓	31.50	37.59	12.61	-	16.58	30.73	25.80
SFT Based								
QA	✗	66.00	46.64	65.20	-	33.03	48.96	51.97
GPT Distill	✓	62.30	46.72	51.83	51.11	31.43	47.41	48.47
Self-Train Based								
STaR	✓	77.84	53.60	61.45	78.22	34.08	50.47	59.28
R^3V	✓	**83.27**	**57.36**	**68.81**	**82.89**	**39.25**	**54.66**	**64.37**
LLaVA-1.5								
Zero-shot QA	✗	17.66	13.04	19.04	-	26.92	36.63	22.66
Zero-shot CoT	✓	15.33	8.39	13.87	-	23.47	35.81	19.37
SFT Based								
QA	✗	48.06	27.20	75.08	-	42.17	52.63	49.03
GPT Distill	✓	44.63	28.48	56.52	60.44	33.81	47.54	45.24
Self-Train Based								
STaR	✓	56.67	33.44	73.46	76.00	41.25	54.06	55.81
R^3V	✓	**59.30**	**33.92**	**79.01**	**80.11**	**45.76**	**56.08**	**59.03**

Table 1: Main results on six vision-language reasoning benchmarks. *Is CoT?* column indicates whether a CoT or a direct answer was generated. *Avg.* column reports the average performance across all tasks (- indicates MiniWob is not applicable to this setting and is excluded from the average). R^3V significantly improves upon the GPT-distilled baseline without additional annotation costs, and surpasses the strong baseline STaR by a large margin.

achieved by our framework in out-of-distribution (OOD) scenarios (Section 4.4).

4.1 Datasets

We validate our framework's self-improvement on six vision-language reasoning benchmarks, which require integrating visual information into complex, multi-step reasoning. Refer to Appendix B for detailed information of these benchmarks.

TabMWP (Lu et al., 2022): A dataset for table-based math word problems requiring reasoning and numerical calculation.

ChartQA (Masry et al., 2022): Focuses on reasoning and calculations within real-world charts.

CLEVR-Math (Lindström and Abraham, 2022): Compositional reasoning over abstract figures.

MiniWob (Shi et al., 2017): A widely-used multimodal web navigation benchmark requiring models to generate multi-step actions.

GeoQA (Chen et al., 2021): A geometry problem benchmark requiring complex reasoning.

M^{3}CoT (Chen et al., 2024a): A recently introduced dataset featuring multi-domain, multi-step multimodal reasoning problems.

4.2 Experimental Setings

We primarily compare our framework with three categories of methods to comprehensively assess its effectiveness. All experiments are conducted under the same parameters to ensure a fair comparison.

Zero-shot Methods. We evaluated the MLLMs' zero-shot performance under the direct prompt (where the model tends to provide an immediate answer (Liu et al., 2024)) and the CoT prompt using "Let's think step by step." GPT-4o was also chosen as a strong baseline for comparison.

Supervised Fine-tuning Baselines. Since the self-training requires existing (I, x, a) datasets, we provide the results of fine-tuning MLLMs using direct prompts on these question-answer pairs. We also include a GPT distillation baseline, where GPT-4o annotates CoT rationales for a small subset of each dataset, and then the open-source MLLMs are fine-tuned for CoT reasoning.

Self-Training Methods. We employ the aforementioned GPT-distilled, warmed-up MLLM as the starting point for self-training, iteratively sampling positive and negative rationales from training samples for continuous self-improvement. We then compare R^3V with the well-known self-training baselines STaR (Zelikman et al., 2022), which iteratively fine-tunes on self-generated positive solutions for model improvement.

We use two established MLLMs, Qwen-VL (Bai et al., 2023) and LLaVA-1.5 (Liu et al., 2024), as

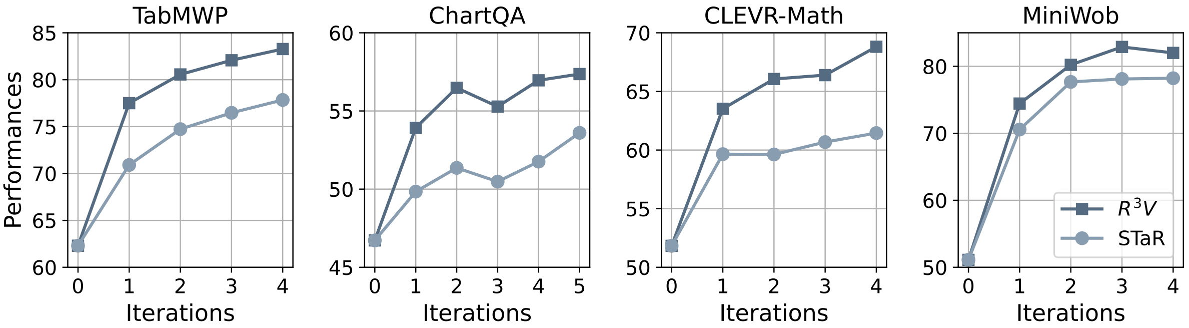


Figure 3: Comparison of the iterative self-training process between *R³V* and STaR on Qwen-VL across four benchmarks. Full results are provided in Appendix F. *R³V* demonstrates higher efficiency in evolution and superior final performance.

base models for self-training. We sample three solutions per sample in each iteration by default. The total number of iterations for these tasks is set to 4-5, depending on the convergence speed. Further details can be found in Appendix C.

4.3 Main Results

Table 1 presents the evaluation results of *R³V* on various multimodal reasoning tasks, including logical and numerical reasoning, agentic tasks, geometry, and multi-domain scenarios. The evolution progress of self-training is illustrated in Figure 3.

Self-training effectively converts MLLMs from weak to strong. Open-source MLLMs struggle with complex vision-language reasoning tasks. CoT reasoning with the "Let's think step by step" prompt (Zero-shot CoT) proves ineffective, with performance even worse than direct prompting (Zero-shot QA). In this situation, the self-training method leverages MLLMs' pre-existing but weak CoT capabilities to bootstrap multimodal CoT data for self-improvement. This process progressively elevates MLLMs' CoT reasoning, as shown in Figure 3, taking it to the next level on top of the GPT-distilled baseline. As an example with Qwen-VL, our self-training framework *R³V* delivers an average 32.8% relative performance improvement over the GPT-distilled baseline (48.47 → 64.37). This result highlights the remarkable potential of MLLMs to enhance their reasoning capabilities through self-training on synthetic data.

Methods	**MMMU**	**MathVista**	**VCR**
Qwen-VL	30.44	29.1	34.02
+ GPT-distilled	33.67	32.7	45.39
+ Ours	35.63	35.10	50.23
+ Ours (TTS)	**38.48**	**35.80**	**51.78**

Table 2: Evaluation results on OOD benchmarks. *Ours (TTS)* denotes Test-time Selection, a new feature introduced by our framework. The self-generated CoT data in R3V contributes to improving performance in more challenging scenarios. Test-time selection is also capable of generalizing to OOD settings.

***R³V* further enhances self-training efficiency by learning from mistakes.** Instead of discarding valuable negative samples, our *R³V* framework leverages carefully designed self-refine and self-select mechanisms to learn from negative solutions, surpassing the strong self-training baseline STaR by a large margin (average 59.28 → 64.37 on Qwen-VL). As shown in Figure 3, *R³V* demonstrates swift adaptation across different multimodal scenarios, achieving notably higher gains in the first iteration compared to the STaR baseline, highlighting the efficiency of our method. These results underscore the value of learning from mistakes in multimodal reasoning and demonstrate the effectiveness of our reflection-based methodology.

4.4 Out-of-Distribution Evaluation

Beyond the success of the *R³V* framework on in-domain benchmarks, we are curious whether its reasoning improvements can generalize to out-of-distribution (OOD) and more difficult vision-language tasks. To this end, we aggregated the CoT rationales self-generated by *R³V* across in-domain benchmarks and constructed positive and negative pairs for continual training on Qwen-VL. For a fair comparison, we also included a baseline that uses only GPT-distilled positive CoT annotations. We conducted evaluations on three challenging benchmarks: (1) **MMMU** (Yue et al., 2024), a multi-discipline dataset designed to evaluate various aspects of multimodal reasoning; (2) **MathVista** (Lu et al., 2023), which focuses specifically on mathematical reasoning in multimodal contexts; (3) **VCR**

Methods	Logical and Numerical reasoning			Agentic	Geometry	Multi-Domain	Avg
	TabMWP	ChartQA	CLEVR-Math	MiniWob	GeoQA	M^3CoT	
R^3V	**83.27**	**57.36**	**68.81**	**82.89**	**39.25**	**54.66**	**64.37**
w/o self-refine	80.87	56.32	64.51	80.67	38.33	54.31	62.50
w/o self-select	79.72	55.36	64.00	79.11	35.81	50.69	60.78
w/o iteration	78.53	54.72	64.56	76.87	36.07	53.11	60.64

Table 3: Ablation study of key components. *w/o iteration* refers to the ablation of iterative training, where we sample $num\_sample\_per\_iter * num\_iter$ samples in a single pass.

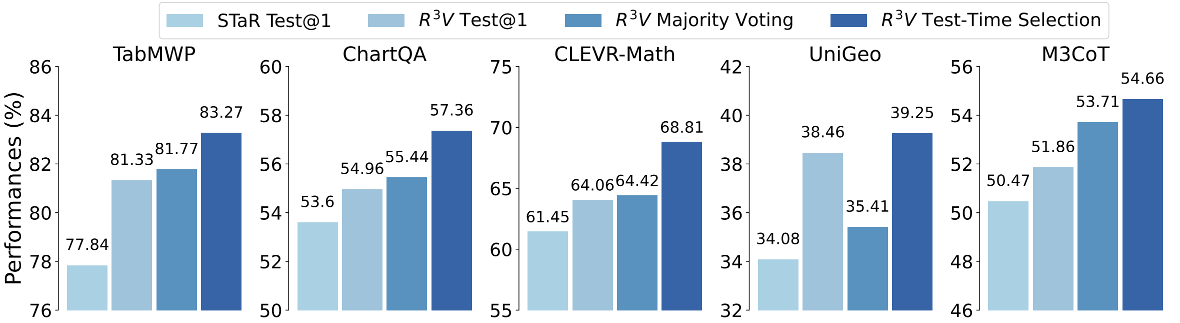


Figure 4: Performance comparison of different test-time methods. Our test-time selection is robust and effective, consistently outperforming Test@1 and majority voting.

(Zellers et al., 2019), a cognition-level visual understanding benchmark that requires reasoning based on common sense and visual content.

***R^3V* also strengthens multimodal reasoning in OOD scenarios.** As shown in Table 2, after incorporating *R^3V*'s self-generated CoT reasoning data, Qwen-VL significantly outperforms both the zero-shot and GPT-distilled baselines. This demonstrates that the CoT annotations synthesized by our framework not only enhance MLLM in-domain reasoning but also generalize to OOD and more challenging vision-language tasks.

Test-time selection generalizes to unseen tasks. Somewhat surprisingly, we found that the test-time selection ability does generalize to unseen tasks. For example, on MMMU, sampling three times during inference combined with our self-select mechanism (see Section 3.2.3) led to further improvement (35.63 → 38.48). This suggests that through our self-select training, the MLLM has learned to compare multiple reasoning paths, identify errors (e.g., recognition or calculation mistakes), and eliminate incorrect options to arrive at the correct answer.

5 Analysis

This section analyzes the key factors behind the success of the *R^3V*, as well as the potential challenges of self-training in multimodal reasoning tasks.

5.1 Ablation Studies

Reflection on self-generated CoT facilitates learning from mistakes. To validate the effectiveness of each part of our framework, we independently ablated the self-refine and self-select losses, denoted as *w/o self-refine* and *w/o self-select*. As shown in Table 3, both self-refine and self-select play a crucial role in improving performance. This highlights the value of negative samples, while our *R^3V* framework's reflection mechanism (i.e., self-refine and self-select losses) serves as an effective method for learning from mistakes.

Iterative training process is crucial for self-improvement. Next, we ablated iterative training as *w/o iteration*: instead of iteratively sampling and training, we sampled a large batch at once. For example, iterative self-training samples three times per round over four rounds, while *w/o iteration* samples $3 \times 4 = 12$ times in a single pass. This approach is similar to Rejection Sampling Fine-tuning (RFT; Yuan et al. (2023)), but includes our self-refine and self-select losses. The results in Table 3 demonstrate the importance of iteratation. Although *w/o iteration* produces a large number of positive and negative samples (comparable to *R^3V* by our statistics), the progressive training process yields higher-quality, more diverse samples, which boosts self-training performance.

Methods	Logical and Numerical reasoning			Agentic	Geometry	Multi-Domain	Avg
	TabMWP	ChartQA	CLEVR-Math	MiniWob	GeoQA	M^3CoT	
STaR	56.67	33.44	73.46	76.0	41.25	54.06	55.81
STaR+DPO	57.61	32.64	73.27	75.33	44.03	52.98	55.90
R^3V	**59.30**	**33.92**	**79.01**	**80.11**	**45.76**	**56.08**	**59.03**

Table 4: Comparison between R^3V and the reinforced baseline (DPO). Due to the noisy nature of CoT in multimodal scenarios, the DPO method struggles to efficiently learn from mistakes and improve performance.

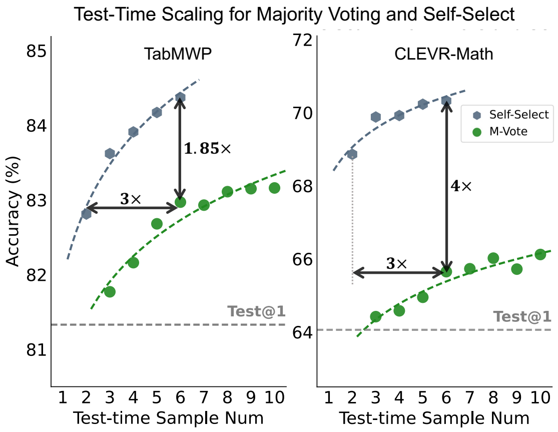


Figure 5: Comparison of scalability between test-time selection and majority voting.

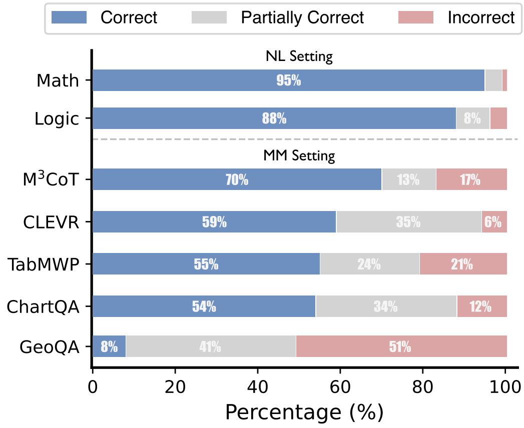


Figure 6: Proportion of correct rationales in solutions with correct answers. Multimodal CoT contains substantially more noise than text-based CoT.

5.2 Test-time Compute

Test-time self-selection boosts performance through sampling. One key advantage of R^3V framework lies in its capacity to enhance performance by scaling test-time computation: during inference, we sample multiple candidate solutions and apply self-select to choose the answer. Figure 4 compares self-selection with Test@1 and majority voting with a sample size of 3. Our self-selection method consistently outperforms Test@1 and majority voting across all tasks. While majority voting reduces noise by aggregating results, self-selection goes further by deeply comparing reasoning paths, eliminating incorrect options, and ultimately analyzing to reach the correct answer.

R^3V consistently benefits from the scaling of sampling size. An open question is the scalability of our test-time selection. We conducted experiments with Qwen-VL on the TabMWP and CLEVR-Math benchmarks, comparing the performance of self-select and majority voting as the sample size increases. As shown in Figure 5, scaling the sample size consistently improves the performance of test-time selection, achieving both higher efficiency and accuracy compared to majority voting. Due to limitations in input length and capability of current MLLMs, performance plateaus with excessive sample size, which we believe stronger base models could address.

Our self-training framework requires no manual annotation, instead synthesizing large-scale positive and negative CoT rationales through sampling, equipping the model with the capacity for self-reflection during reasoning. It also opens up new opportunities for boosting MLLM reasoning performance by scaling test-time computation.

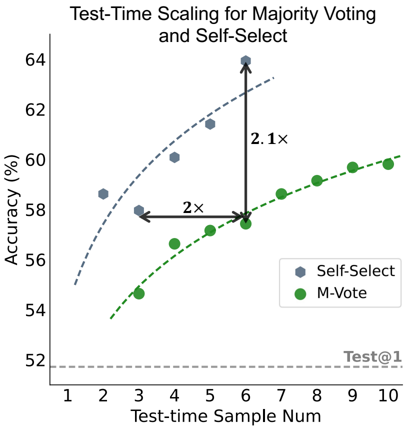
5.3 The Noisy Nature of Multimodal CoT

In our preliminary study, we found that the widely-used preference learning method DPO (Rafailov et al., 2024) struggles to leverage positive and negative solutions for further improvement in multimodal settings. As shown in Table 4, equipping STaR with DPO training yields minimal improvement and falls short of our R^3V.

To investigate DPO's failure, we closely examined the positive and negative samples self-generated by the MLLM (see details and case study in Appendix G). For each task, we randomly selected 100 positive solutions based on answer correctness and manually categorized their CoT fidelity as correct, partially correct, or incorrect. As shown in Figure 6, unlike natural language reasoning tasks (e.g., Logic, Math), multimodal CoT

Methods	**Score**
Zero-shot CoT	17.11
Self-Train R^3V	51.72
+ Test-Time Selection (N=3)	**57.96**

(a) The effectiveness of our self-training framework R^3V.



(b) Scalability between test-time selection and majority voting.

Figure 7: Evaluation result of Qwen2-VL on GeoQA. (a) shows that our self-training approach significantly enhances performance without GPT-distilled warmup. (b) demonstrates the superior scalability of test-time selection, which boosts performance through sampling.

contains significant noise, with the proportion of fully correct CoT ranging from 8% to 70%. This stems from MLLM's limited recognition capabilities, leading to flawed CoT despite correct answers, such as OCR errors. As a result, faulty reasoning in noisy CoT is often misjudged as better solutions, making it challenging for DPO to distinguish between correct and incorrect reasoning paths and ultimately reducing performance (Chowdhury et al., 2024). In contrast, our reflection method avoids encouraging the generation of faulty solutions, instead guiding the model to select the correct answer through elimination, demonstrating greater efficiency in noisy multimodal CoT scenarios.

5.4 Generalization to Stronger Backbone

To demonstrate generalizability, we applied R^3V to the latest advanced MLLM, Qwen2-VL (Wang et al., 2024a) , evaluating its ability to self-improve in solving geometric problems (Chen et al., 2021). As shown in Figure 7a, even without GPT-distilled warmup, R^3V achieves significant self-improvement by leveraging the model's pre-existing CoT abilities, demonstrating the R^3V's generalizability across backbones. More impressively, we found that test-time selection demonstrates superior scalability on Qwen2-VL, markedly outperforming majority voting, as illustrated in Figure 7b. We hypothesize that the enhanced general capabilities of the base model further amplify the effectiveness of self-select, which we leave for future exploration.

6 Conclusion

The scarcity of multimodal CoT data limits the reasoning capabilities of current MLLMs. In this paper, we take the first step toward enabling MLLMs to self-improve for better vision-language reasoning. We propose an iterative self-training framework, R^3V, which continuously bootstraps positive and negative solutions and improves reasoning through reflection on self-generated CoT rationales. Meanwhile, R^3V enables MLLMs to self-reflect on their generated solutions, offering new opportunities for boosting performance through test-time computation. Extensive experiments and analyses demonstrate the effectiveness of our framework and the key factors behind its success.

Limitations

As discussed in Section 5.3, due to the limitations of current MLLMs, the CoT annotations generated by R^3V often contain noise. While our framework can self-improve performance on noisy multimodal CoT, we believe that higher-quality CoT will further enhance reasoning ability. Due to computational constraints, our main experiments were conducted on two well-known MLLMs, LLaVA and Qwen-VL. Expanding to larger and more advanced MLLMs could yield interesting results, which we plan to explore in future work.

References

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

Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. 2023. Learning from mistakes makes llm better reasoner. *arXiv preprint arXiv:2310.20689*.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

Jiaqi Chen, Tong Li, Jinghui Qin, Pan Lu, Liang Lin, Chongyu Chen, and Xiaodan Liang. 2022. UniGeo: Unifying geometry logical reasoning via reformulating mathematical expression. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3313–3323, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric P Xing, and Liang Lin. 2021. Geoqa: A geometric question answering benchmark towards multimodal numerical reasoning. *arXiv preprint arXiv:2105.14517*.

Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. 2023. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*.

Qiguang Chen, Libo Qin, Jin Zhang, Zhi Chen, Xiao Xu, and Wanxiang Che. 2024a. M^3CoT: A novel benchmark for multi-domain multi-step multi-modal chain-of-thought. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Bangkok, Thailand.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. 2024b. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. 2024c. Self-play fine-tuning converts weak language models to strong language models. *arXiv preprint arXiv:2401.01335*.

Kanzhi Cheng, Wenpo Song, Zheng Ma, Wenhao Zhu, Zixuan Zhu, and Jianbing Zhang. 2023. Beyond generic: Enhancing image captioning with real-world knowledge using vision-language pre-training model. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 5038–5047.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. 2024. Seeclick: Harnessing gui grounding for advanced visual gui agents. *arXiv preprint arXiv:2401.10935*.

Sayak Ray Chowdhury, Anush Kini, and Nagarajan Natarajan. 2024. Provably robust dpo: Aligning language models with noisy feedback. *arXiv preprint arXiv:2403.00409*.

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

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Preprint*, arXiv:2305.06500.

Yihe Deng, Pan Lu, Fan Yin, Ziniu Hu, Sheng Shen, James Zou, Kai-Wei Chang, and Wei Wang. 2024. Enhancing large vision language models with self-training on image comprehension. *arXiv preprint arXiv:2405.19716*.

K Anders Ericsson and Herbert A Simon. 1980. Verbal reports as data. *Psychological review*, 87(3):215.

Jiahui Gao, Renjie Pi, Jipeng Zhang, Jiacheng Ye, Wanjun Zhong, Yufei Wang, Lanqing Hong, Jianhua Han, Hang Xu, Zhenguo Li, et al. 2023. G-llava: Solving geometric problem with multi-modal large language model. *arXiv preprint arXiv:2312.11370*.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, et al. 2023. Reinforced self-training (rest) for language modeling. *arXiv preprint arXiv:2308.08998*.

Anisha Gunjal, Jihan Yin, and Erhan Bas. 2024. Detecting and preventing hallucinations in large vision language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Yucheng Han, Chi Zhang, Xin Chen, Xu Yang, Zhibin Wang, Gang Yu, Bin Fu, and Hanwang Zhang. 2023. Chartllama: A multimodal llm for chart understanding and generation. *arXiv preprint arXiv:2311.16483*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Arian Hosseini, Xingdi Yuan, Nikolay Malkin, Aaron Courville, Alessandro Sordoni, and Rishabh Agarwal. 2024. V-star: Training verifiers for self-taught reasoners. *arXiv preprint arXiv:2402.06457*.

Cheng-yu Hsieh, Chun-liang Li, Chih-kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Jiaxin Huang, Shixiang Shane Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2022. Large language models can self-improve. *arXiv preprint arXiv:2210.11610*.

Muye Huang, Lai Han, Xinyu Zhang, Wenjun Wu, Jie Ma, Lingling Zhang, and Jun Liu. 2024. Evochart: A benchmark and a self-training approach towards real-world chart understanding. *arXiv preprint arXiv:2409.01577*.

Mengzhao Jia, Zhihan Zhang, Wenhao Yu, Fangkai Jiao, and Meng Jiang. 2024. Describe-then-reason: Improving multimodal mathematical reasoning through visual comprehension training. *arXiv preprint arXiv:2404.14604*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR.

Adam Dahlgren Lindström and Savitha Sam Abraham. 2022. Clevr-math: A dataset for compositional language, visual and mathematical reasoning. *arXiv preprint arXiv:2208.05358*.

Fangyu Liu, Guy Emerson, and Nigel Collier. 2023. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2020. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. *arXiv preprint arXiv:2007.08124*.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2023. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*.

Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. 2022. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. *arXiv preprint arXiv:2209.14610*.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. *arXiv preprint arXiv:2203.10244*.

Arindam Mitra, Hamed Khanpour, Corby Rosset, and Ahmed Awadallah. 2024. Orca-math: Unlocking the potential of slms in grade school math. *arXiv preprint arXiv:2402.14830*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Tianlin Shi, Andrej Karpathy, Linxi Fan, Jonathan Hernandez, and Percy Liang. 2017. World of bits: An open-domain platform for web-based agents. In *International Conference on Machine Learning*, pages 3135–3144. PMLR.

Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See-Kiong Ng, Lidong Bing, and Roy Ka-Wei Lee. 2024. Math-llava: Bootstrapping mathematical reasoning for multimodal large language models. *arXiv preprint arXiv:2406.17294*.

Yifan Song, Da Yin, Xiang Yue, Jie Huang, Sujian Li, and Bill Yuchen Lin. 2024. Trial and error: Exploration-based trajectory optimization for llm agents. *arXiv preprint arXiv:2403.02502*.

Qiushi Sun, Zhirui Chen, Fangzhi Xu, Kanzhi Cheng, Chang Ma, Zhangyue Yin, Jianing Wang, Chengcheng Han, Renyu Zhu, Shuai Yuan, et al. 2024. A survey of neural code intelligence: Paradigms, advances and beyond. *arXiv preprint arXiv:2403.14734*.

Qiushi Sun, Zhangyue Yin, Xiang Li, Zhiyong Wu, Xipeng Qiu, and Lingpeng Kong. 2023. Corex: Pushing the boundaries of complex reasoning through multi-model collaboration. *arXiv preprint arXiv:2310.00280*.

Junyang Wang, Yiyang Zhou, Guohai Xu, Pengcheng Shi, Chenlin Zhao, Haiyang Xu, Qinghao Ye, Ming Yan, Ji Zhang, Jihua Zhu, et al. 2023. Evaluation and analysis of hallucination in large vision-language models. *arXiv preprint arXiv:2308.15126*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. 2024a. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Tianduo Wang, Shichen Li, and Wei Lu. 2024b. Self-training with direct preference optimization improves chain-of-thought reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11917–11928.

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

Fangzhi Xu, Qiushi Sun, Kanzhi Cheng, Jun Liu, Yu Qiao, and Zhiyong Wu. 2024. Interactive evolution: A neural-symbolic self-training framework for large language models. *arXiv preprint arXiv:2406.11736*.

Zhen Yang, Jinhao Chen, Zhengxiao Du, Wenmeng Yu, Weihan Wang, Wenyi Hong, Zhihuan Jiang, Bin Xu, Yuxiao Dong, and Jie Tang. 2024. Mathglm-vision: Solving mathematical problems with multi-modal large language model. *arXiv preprint arXiv:2409.13729*.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2020. Reclor: A reading comprehension dataset requiring logical reasoning. *arXiv preprint arXiv:2002.04326*.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Keming Lu, Chuanqi Tan, Chang Zhou, and Jingren Zhou. 2023. Scaling relationship on learning mathematical reasoning with large language models. *arXiv preprint arXiv:2308.01825*.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. 2024. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488.

Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6720–6731.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Peng Gao, et al. 2024a. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? *arXiv preprint arXiv:2403.14624*.

Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Yichi Zhang, Ziyu Guo, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Bin Wei, Shanghang Zhang, et al. 2024b. Mavis: Mathematical visual instruction tuning. *arXiv preprint arXiv:2407.08739*.

Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. 2023. Multi-modal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*.

Zhiyuan Zhao, Bin Wang, Linke Ouyang, Xiaoyi Dong, Jiaqi Wang, and Conghui He. 2023. Beyond hallucinations: Enhancing lvlms through hallucination-aware direct preference optimization. *arXiv preprint arXiv:2311.16839*.

Yiyang Zhou, Zhiyuan Fan, Dongjie Cheng, Sihan Yang, Zhaorun Chen, Chenhang Cui, Xiyao Wang, Yun Li, Linjun Zhang, and Huaxiu Yao. 2024. Calibrated self-rewarding vision language models. *arXiv preprint arXiv:2405.14622*.

A Additional Related Work

Multimodal Large Language Models and Multimodal Reasoning Driven by the advancement of Large Language Models (LLMs), the multimodal research community has recently witnessed a domain shift from Vision-Language Models (VLMs) (Radford et al., 2021; Li et al., 2022; Cheng et al., 2023) to Multimodal Large Language Models (MLLMs) (Achiam et al., 2023; Liu et al., 2023; Chen et al., 2023; Cheng et al., 2024). Unimodal reasoning has a strong research foundation, such as in mathematics (Hendrycks et al., 2021) and code generation (Sun et al., 2024). Multimodal reasoning requires models to integrate visual cues into the reasoning process (Zhang et al., 2023), presenting new challenges. Recent studies have explored synthesizing table or chart data and leveraging GPT to annotate CoT, aiming to enhance MLLM reasoning capabilities (Han et al., 2023; Jia et al., 2024). For instance, Huang et al. (2024) utilizes GPT to generate chart code and render it to obtain diverse chart reasoning samples. In this work, we do not rely on stronger models to synthesize new reasoning samples; instead, we enable MLLMs to achieve self-improvement from self-generated CoT data.

Self-Training Methods Self-training, especially integrated with reinforcement learning from its own outputs, offers a promising avenue for model self-improvement (Huang et al., 2022; Gulcehre et al., 2023). Recent studies have applied self-training to MLLMs with the goal of enhancing image comprehension, particularly in mitigating hallucinations (Zhou et al., 2024; Gunjal et al., 2024; Zhao et al., 2023). Deng et al. (2024) proposes constructing positive and negative sample pairs by perturbing images and prompts, and enhances alignment through DPO training. In contrast, this work focuses on complex reasoning in multimodal scenarios, which requires integrating visual cues to generate step-by-step reasoning CoT. To our knowledge, we are the first to explore self-training in the context of vision-language reasoning.

B Vision-Language Reasoning Benchmarks

TabMWP (Lu et al., 2022) Tabular Math Word Problems (TabMWP) is benchmark containing open-domain grade-level problems that require mathematical reasoning and calculation on table figures. We use the standard train/test split provided by the author.

ChartQA (Masry et al., 2022) We used the human-written version as the self-train benchmark, which contains more reasoning-intensive questions compared to the augmented split. This subset contains 7,398 chart figures and question pairs, comprising both free-text and multiple-choice questions.

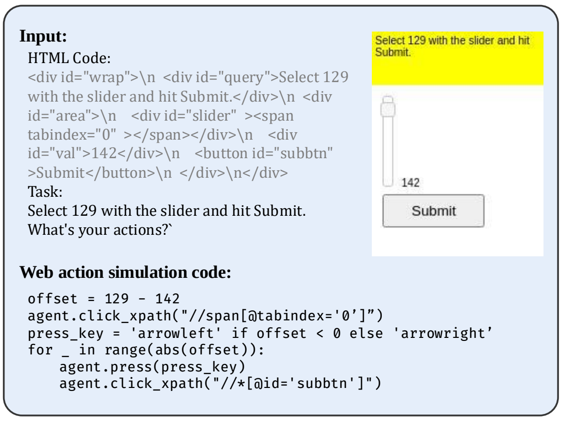
CLEVR-Math (Lindström and Abraham, 2022) The CLEVR-Math dataset consists of multimodal math word problems that combine text and images, where questions are posed about the state of the scene after a sequence of actions (like addition or subtraction of objects) have been applied. We randomly sampled 10000 instances for training and used the original test set.

MiniWob (Shi et al., 2017) MiniWob asks MLLM to interact with a simulated Web environment. As shown in Figure 8, the model is provided with an image of the web interface along with the html as input. It is then asked to generate Python code to simulate keyboard and mouse actions and complete the given task.

GeoQA (Chen et al., 2021) GeoQA contains 4,998 multiple-choice geometric problems from Chinese middle school exams and annotated with solving programs. We use human translated English version provided by UniGeo (Chen et al., 2022).

M^{3}CoT (Chen et al., 2024a) M^{3}CoT is a manually verified multimodal, multi-domain, multi-step visual-language reasoning dataset. We use the official train/test splits in our R^3V self-training process.

Evaluation For structured outputs like GPT-distilled and self-train methods, we use the benchmark's default evaluation script to calculate metrics. For free-form outputs like the zero-shot CoT baseline, we employ GPT-4o-mini as the evaluator to assess accuracy. For MiniWob, the simulated web environment provides an automatic reward of



Input:

HTML Code:

<div id="wrap">\n <div id="query">Select 129 with the slider and hit Submit.</div>\n <div id="area">\n <div id="slider" ><span tabindex="0" ></span></div>\n <div id="val">142</div>\n <button id="subbtn" >Submit</button>\n </div>\n</div>

Task:

Select 129 with the slider and hit Submit.

What's your actions?`

Web action simulation code:

```
offset = 129 - 142
agent.click_xpath("//span[@tabindex='0’]”)
press_key = 'arrowleft' if offset < 0 else 'arrowright’
for _ in range(abs(offset)):
    agent.press(press_key)
    agent.click_xpath("//*[@id='subbtn']")
```

Figure 8: Example for MiniWob.

Dataset	Train/Test	GPT Anno.	Iter#
TabMWP	23059 / 7686	1000	4
CLEVR	10000 / 7955	1000	4
ChartQA	7398 / 1250	800	5
MiniWob	–	550	4
M^{3}CoT	7973 / 2359	936	4
GeoQA	3499 / 754	536	5

Table 5: Dataset Statistics

Hyperparameter	Qwen-VL	LLaVA-1.5
Batch Size	64	
LR	3e-5	
Epochs	3	
LR Schedule	Constant with warmup	
LR Warmup Ratio	0.1	
Weight Decay	0	
LoRA Rank	64	128
LoRA alpha	16	256
LoRA Dropout	0.05	
Optimizer	AdamW	

Table 6: Hyperparameter Settings

0 or 1, which we use to determine the solution's correctness.

C Training Details

We use the Qwen-VL and LLaVA-1.5 as the base model and conducted experiments on an MLLM training infrastructure [*]. We show the number of training and testing samples for each dataset, along with the amount of GPT annotations in Table 5. Our self-training process begins with the GPT-distilled warmup, where we fine-tune the base model using the training dataset augmented with GPT-4o CoT annotations. After this warm-up, the fine-tuned MLLM is employed to sample from the training dataset to build SFT, Self-Refine, and Self-Select data for our training in the R^3V framework. We performed self-training in either four or five iterations, depending on performance saturates.

The same training hyperparameters are used across all experiments, as detailed in Table 6. We employ DeepSpeed to train MLLM using the Zero2 strategy, maintaining a global batch size of 64.

D Algorithms

Algorithm 1 describes the overall process of R^3V. The inner for-loop describes how we sample instances to build the proposed dataset, where we always select the most recent data. It is important to note that the sampled instances must be formatted as the data examples shown in Appendix E later.

E Examples of R^3V Multi-task Learning

We illustrate examples of SFT, Self-Refine, and Self-Select in Figure 10. The input part in each subfigure shows the contexts such as question-choices pair and prompts used to guide the MLLM, and the output part shows the expected response from the MLLM. CoT prompt "Let's think step by step." will always be appended to the question-choices pairs. The prompts for self-refine and self-select vary slightly between multiple-choice and short-answer tasks. Note that only the self-select prompt will be used for test-time reflection.

We add the sample generated by MLLM into both self-refine and self-select contexts using "Model Prediction" to divide with the question-choices part. As illustrated in the figure, we use a green checkmark to indicate the positive solutions and a red cross to mark the negative ones. It highlights that R^3V successfully builds negative-positive rationales pairs, from which the model can learn from mistakes in negative demonstrations. Additionally, R^3V also builds diverse reasoning paths, ranging from completely wrong to correct rationales for the MLLM learning to choose from like human.

F Evolution Progress

Figure 11 shows the evolution progress of our R^3V framework.

G Noisy Nature of Multimodal CoT

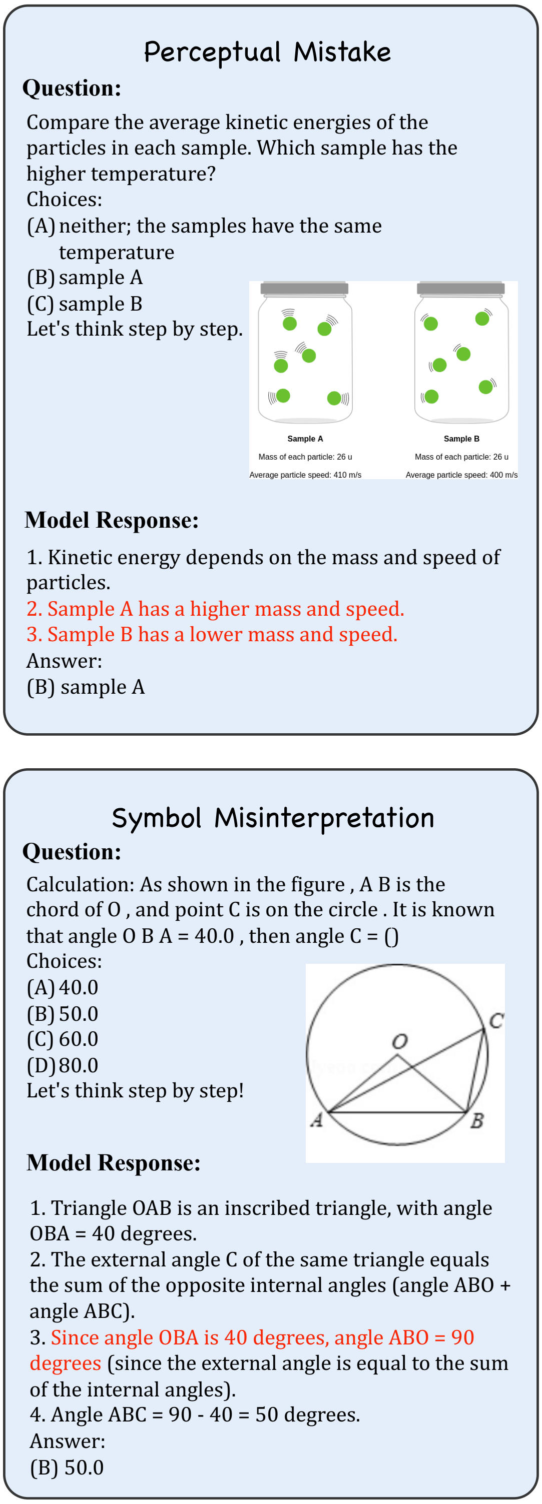
We manually reviewed the positive solutions generated by the Qwen-VL in our self-training process and evaluated the quality of its CoT reasoning. The CoT error in multimodal setting is significantly higher than samples from logical reasoning datasets (Liu et al., 2020; Yu et al., 2020) and math datasets(Cobbe et al., 2021; Hendrycks et al., 2021) in natural language setting. Multimodal CoT has considerable noise, such as visual perception error and symbol misinterpretation. We highlight this issue with case studies on the M^{3}CoT and GeoQA dataset in Figure 9.

---

[*] `https://github.com/TideDra/VL-RLHF`

Algorithm 1 R^3V

```
Require: Training QA datasets $\mathcal{D}_{\text{QA}}$, subset with GPT-distilled CoT annotations $\mathcal{D}^{w}_{\text{SFT}}$, model $\mathcal{M}$, number of iterations $T$
Initialize $\mathcal{D}_0 = \mathcal{D}_{\text{QA}} \cup \mathcal{D}^{w}_{\text{SFT}}$ , $\mathcal{D}_{\text{pos}} = \emptyset$, $\mathcal{D}_{\text{neg}} = \emptyset$
for each iteration $t = 1, 2, \dots, T$ do
    $\mathcal{D}_{\text{SFT}} = \mathcal{D}_{\text{REF}} = \mathcal{D}_{\text{SEL}} = \emptyset$
    $\mathcal{M}_t \leftarrow \text{SFT}(\mathcal{M}, \mathcal{D}_{t-1})$
    # sample 3 times on training set
    $\mathcal{S} \leftarrow \text{sample}(\mathcal{M}_t, \mathcal{D}_{\text{QA}}, n = 3)$
    $\mathcal{D}_{\text{pos}} \leftarrow \mathcal{D}_{\text{pos}} \cup \text{eval\_pos}(\mathcal{S}, \mathcal{D}_{\text{QA}})$
    $\mathcal{D}_{\text{neg}} \leftarrow \mathcal{D}_{\text{neg}} \cup \text{eval\_neg}(\mathcal{S}, \mathcal{D}_{\text{QA}})$
    for item index $i = 1, 2, \dots, |\mathcal{D}_{\text{QA}}|$ do
        # all solutions for sample $i$
        $\mathbf{d}^i_+ = \{s^i_1, s^i_2, \dots, s^i_k | s^i_k \in \mathcal{D}^i_{\text{pos}}\}$
        $\mathbf{d}^i_- = \{s^i_1, s^i_2, \dots, s^i_k | s^i_k \in \mathcal{D}^i_{\text{neg}}\}$
        # the $r$-th latest pos/neg solution
        $s^{+/-}_r \leftarrow \mathbf{d}^i_{\text{+/-}}[-r]$
        # SFT
        if $\exists\ s^+_1$ then
            $\mathcal{D}_{\text{SFT}} \leftarrow \mathcal{D}_{\text{SFT}} + s^+_1$
        end if
        # self-refine
        if $\exists\ (s^+_1, s^-_1)$ then
            $\mathcal{D}_{\text{REF}} \leftarrow \mathcal{D}_{\text{REF}} + (s^-_1, s^+_1)$
        end if
        # self-select
        if $\exists\ (s^+_1, s^-_1, s^-_2)$ then
            $\mathcal{D}_{\text{SEL}} \leftarrow \mathcal{D}_{\text{SEL}} + (s^+_1, s^-_1, s^-_2)$
        end if
        if $\exists\ (s^+_1, s^+_2, s^-_1)$ then
            $\mathcal{D}_{\text{SEL}} \leftarrow \mathcal{D}_{\text{SEL}} + (s^+_1, s^+_2, s^-_1)$
        end if
    end for
    $\mathcal{D}_t \leftarrow \mathcal{D}_0 \cup \mathcal{D}_{\text{SFT}} \cup \mathcal{D}_{\text{REF}} \cup \mathcal{D}_{\text{SEL}}$
end for
```



Perceptual Mistake

Question:

Compare the average kinetic energies of the particles in each sample. Which sample has the higher temperature?

Choices:

(A) neither; the samples have the same temperature

(B) sample A

(C) sample B

Let's think step by step.

Sample A
Mass of each particle: 26 u
Average particle speed: 410 m/s
Sample B
Mass of each particle: 26 u
Average particle speed: 400 m/s

Model Response:

1. Kinetic energy depends on the mass and speed of particles.
2. Sample A has a higher mass and speed.
3. Sample B has a lower mass and speed.

Answer:

(B) sample A

Symbol Misinterpretation

Question:

Calculation: As shown in the figure , A B is the chord of O , and point C is on the circle . It is known that angle O B A = 40.0 , then angle C = ()

Choices:

(A) 40.0

(B) 50.0

(C) 60.0

(D) 80.0

Let's think step by step!

Model Response:

1. Triangle OAB is an inscribed triangle, with angle OBA = 40 degrees.
2. The external angle C of the same triangle equals the sum of the opposite internal angles (angle ABO + angle ABC).
3. Since angle OBA is 40 degrees, angle ABO = 90 degrees (since the external angle is equal to the sum of the internal angles).
4. Angle ABC = 90 - 40 = 50 degrees.

Answer:

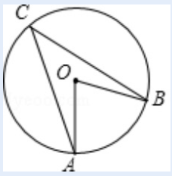
(B) 50.0

Figure 9: The noisy nature of multimodal CoT, where positive solutions often contain flawed CoT steps. The perceptual error and symbol misinterpretation are marked in red.

CoT for self-train

Input:

Calculation: As shown in the figure , it is known that angle AOB is the central angle of circle O , angle AOB = 50.0 , then the degree of angle of circumference angle ACB is ()



Choices:
(A) 50.0
(B) 25.0
(C) 100.0
(D) 30.0

Let's think step by step.

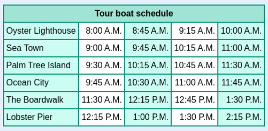
Output:

1. Angle AOB is the central angle, and angle ACB is an inscribed angle subtended by the same arc AB.
2. The measure of an inscribed angle is half the measure of the central angle.
3. Therefore, angle ACB = 50 / 2 = 25 degrees.
Answer: (B) 25.0 ✓

Self-Refine for self-train

Input:

Look at the following schedule. Hayley just missed the 9.45 AM boat at Sea Town. How long does she have to wait until the next boat?



Tour boat schedule

Oyster Lighthouse	8:00 A.M.	8:45 A.M.	9:15 A.M.	10:00 A.M.
Sea Town	9:00 A.M.	9:45 A.M.	10:15 A.M.	11:00 A.M.
Palm Tree Island	9:30 A.M.	10:15 A.M.	10:45 A.M.	11:30 A.M.
Ocean City	9:45 A.M.	10:30 A.M.	11:00 A.M.	11:45 A.M.
The Boardwalk	11:30 A.M.	12:15 P.M.	12:45 P.M.	1:30 P.M.
Lobster Pier	12:15 P.M.	1:00 P.M.	1:30 P.M.	2:15 P.M.

Choices:
(A) 1 hour
(B) 15 minutes
(C) 30 minutes
(D) 45 minutes

Let's think step by step.

Model Prediction:
1. Hayley missed the 9:45 AM boat at Sea Town.
2. The next boat at Sea Town is scheduled for 10:15 AM.
3. The difference between missed boat and next boat is 15 minutes.
Answer: B ✗

Judge the correctness of the model's prediction and refine it.

Output:

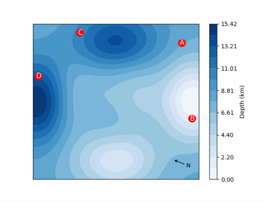
1. The next available boat at Sea Town is at 10:15 A.M.
2. The difference in time is 10:15 A.M. - 9:45 A.M. = 30 minutes.
3. The waiting time is 30 minutes.
Answer: C ✓

Self-Select for self-train

Input:

In the picture, pinpoint the spot with the shallowest depth.
Choices:
(A) Point C in the image
(B) Point A in the image
(C) Point D in the image
(D) Point B in the image

Let's think step by step.



Model Prediction 1:
1. The color scale indicates the depth, with darker colors representing shallower depths.
2. Point C is in the darkest blue area, which corresponds to the shallowest depth.
3. The other options are not as dark as Point C.
Answer: C ✗

Model Prediction 2
1. The color scale indicates depth, with darker blue representing deeper water.
2. Point C is at the darkest blue, indicating the shallowest depth.
3. The other points have similar depths but are not as dark as point C.
Answer: C ✗

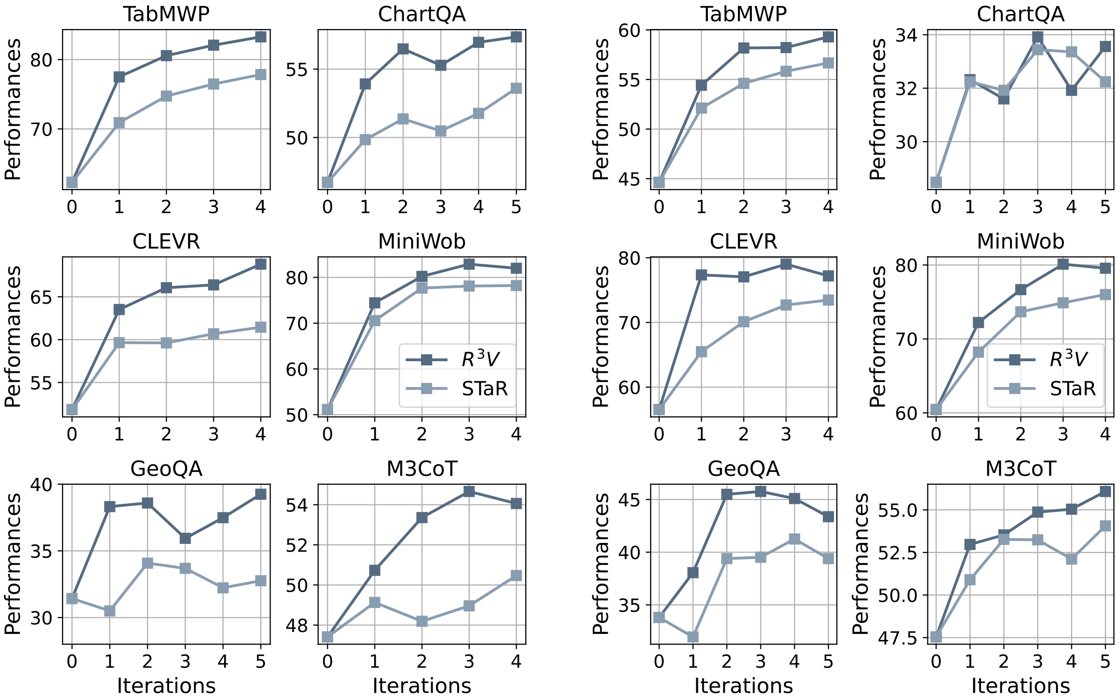
Model Prediction 3:
1. The color scale indicates that deeper colors represent shallower depths.
2. Point B is located in a lighter blue area, indicating shallow depth.
3. The other points (C, A, D) are in deeper blue regions.
Answer: D ✓

Which prediction is correct? Give the final answer for the beginning question by selecting the correct option.

Output:

Answer: D ✓

Figure 10: Examples for generated data. The image displays examples of generated data for SFT, Self-Refine, and Self-Select tasks. Positive samples are marked with a green checkmark, while negative samples are marked with a red cross.



(a) Qwen-VL

(b) LLaVA-1.5

Figure 11: The evolution progress of R^3V.