Towards Medical Complex Reasoning with LLMs through Medical Verifiable Problems

Junying Chen[1], **Zhenyang Cai**[1], **Ke Ji**[1], **Xidong Wang**[1], **Wanlong Liu**[1], **Rongsheng Wang**[1], **Benyou Wang**[1,2*]

[1] The Chinese University of Hong Kong, Shenzhen

[2] Shenzhen Research Institute of Big Data

wangbenyou@cuhk.edu.cn

Abstract

The breakthrough of OpenAI o1 highlights the potential of enhancing reasoning to improve LLM. Yet, most research in reasoning has focused on mathematical tasks, leaving domains like medicine underexplored. The medical domain, though distinct from mathematics, also demands robust reasoning to provide reliable answers, given the high standards of healthcare. However, verifying medical reasoning is challenging, unlike those in mathematics. To address this, we propose **Medical Verifiable Problems** with a medical verifier to check the correctness of model outputs. This verifiable nature enables advancements in medical reasoning through **a two-stage approach**: (1) using the verifier to guide the search for a complex reasoning trajectory for fine-tuning LLMs, (2) applying reinforcement learning (RL) with verifier-based rewards to enhance complex reasoning further. Finally, we introduce **HuatuoGPT-o1**, a medical LLM capable of complex reasoning, which outperforms general and medical-specific baselines using only 40K verifiable problems. Experiments show complex reasoning improves medical problem-solving and benefits more from RL. We hope our approach inspires advancements in reasoning across medical and other specialized domains. Code, datasets, and models are publicly available at https://github.com/FreedomIntelligence/HuatuoGPT-o1.

1 Introduction

The release of OpenAI o1 has marked a significant milestone in large language model (LLM) development, showcasing impressive capabilities (Guan et al., 2024; Xie et al., 2024; Zhong et al., 2024). This breakthrough highlights the potential of **scaling Chain-of-Thought (CoT)** and **reinforcement learning** to enhance LLM performance (Qin et al., 2024; Zeng et al., 2024; Wang et al., 2024a). While subsequent research efforts attempt to replicate these advancements, they often remain limited to mathematical reasoning tasks (Team, 2024b; Luong et al., 2024; Zhang et al., 2024a; Wang et al., 2024a). The application of o1-like methods to specialized fields, such as medicine, remains largely underexplored.

Medical tasks often involve deeper reasoning (Saab et al., 2024; Patel et al., 2005; Chen et al., 2024a). In real-world medical diagnoses or decisions, doctors often deliberate carefully. Such a life-critical field necessitates meticulous thinking to ensure more reliable answers (Xu et al., 2024b; Temsah et al., 2024). Thus, enabling LLMs to perform extended reasoning and reflection to provide more reliable medical responses holds significant value for the future applications of LLMs in healthcare. We term this *extended and reflective thinking process* as **complex reasoning** (Jaech et al., 2024). Moreover, medical reasoning closely resembles real-world applications in domains like finance, law and education, making advancements in this area readily transferable (Cheng et al., 2023).

However, a key challenge for medical reasoning is verifying the thought process, which often lacks clear steps. Inspired by mathematical problems that allow verification through their outcomes, we construct 40K **Medical Verifiable Problems** reformatted from challenging, closed-set medical exam questions. These verifiable problems are characterized as *open-ended* with unique, objective *ground-truth answers* that allow an LLM verifier to check solution correctness. Such verifiability enable a two-stage method for medical complex reasoning:

Stage 1: Learning from Verified Reasoning Trajectories The verifier feedback can guide LLMs to search for a long CoT with reflection. The LLM first initializes a CoT and an answer. When the verifier rejects the answer, the LLM extends by applying a strategy sampled from *Backtracking*,

*Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2025, pages 14552–14573
July 27 - August 1, 2025

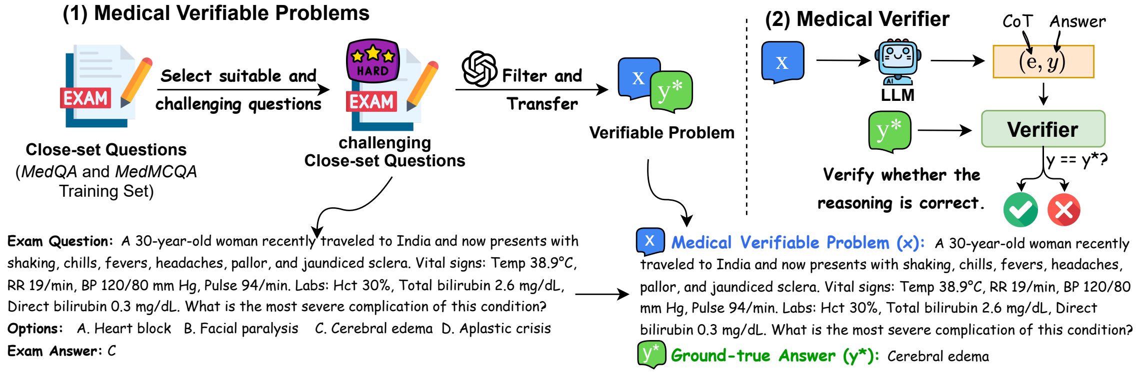


Figure 1: **Left:** Constructing Medical Verifiable Problems using challenging close-set exam questions. **Right:** The verifier checks the model's answer against the *ground-truth answer*.

Exploring New Paths, *Verification*, and *Correction* to refine its answer until it reaches a correct answer. Successful trajectories, involving iterative refinements, are then used to fine-tune the LLM with complex reasoning skills.

Stage 2: Reinforcement Learning with Verification Rewards After acquiring complex reasoning skills, reinforcement learning (RL) further enhances this ability using verification-based rewards. The verification feedback enables the model to spontaneously explore optimal long CoT strategies without human preset guidance.

Using this approach, we present **HuatuoGPT-o1**, a medical LLM that performs extended and reflective thinking before answering. Experiments demonstrate that our method (using only 40K data points) yields an 8-point improvement on medical benchmarks with an 8B model. Furthermore, our 70B model outperforms other general and medical-specific LLMs of similar parameter scale on medical benchmarks. The experiments further reveal the effectiveness and domain compatibility of our method.

Our contributions are as follows:

- To the best of our knowledge, this is the first work to build o1-like LLMs in the medical domain using Medical Verifiable Problems.
- We propose a two-stage training framework based on Medical Verifiable Problems.
- We introduce HuatuoGPT-o1, a medical LLM capable of complex reasoning, which outperforms other baselines on medical benchmarks.
- Our experiments reveal that complex reasoning is effective for medical problem-solving and benefits more from RL enhancements. We also validate the effectiveness of our approach across different languages (e.g., Chinese) and domains (e.g., chemistry).

2 Medical Verifiable Problems

2.1 Philosophy of Verifiability

Solving complex problems often requires long reasoning trajectories. Many approaches (Muennighoff et al., 2025; Guo et al., 2025) integrate pre-defined, high-quality trajectories from expert examples or distilled models into training. While beneficial, these fixed paths can introduce biases from either humans or LLMs, thereby limiting reasoning diversity. To address this, AlphaGo Zero (Silver et al., 2017) uses *result verification* (e.g., win/loss) instead of human game records, reducing path dependency and enabling the potential to surpass human-level performance. More recently, DeepSeek R1 (Guo et al., 2025) leveraged the inherent verifiability of mathematics and code to facilitate advanced mathematical reasoning. This underscores the pivotal role of *verifiability* in incentivizing LLMs toward stronger reasoning.

Inspired by this, we introduce **Medical Verifiable Problems**, which are *open-ended* yet have with unique, objective *ground-truth answers*, as illustrated in Figure 1. This brings verifiability to the medical domain, akin to mathematics, enabling a result-driven verification process.

2.2 Constructing Medical Verifiable Problems

Sourcing from Medical Exam Questions To achieve this, we utilize closed-set real-world exam questions for two key reasons: 1) a large number of medical exam questions are available; and 2) these real-world exam questions are typically objective and accurate. Specifically, we collected 192K medical multiple-choice exam questions from the train-

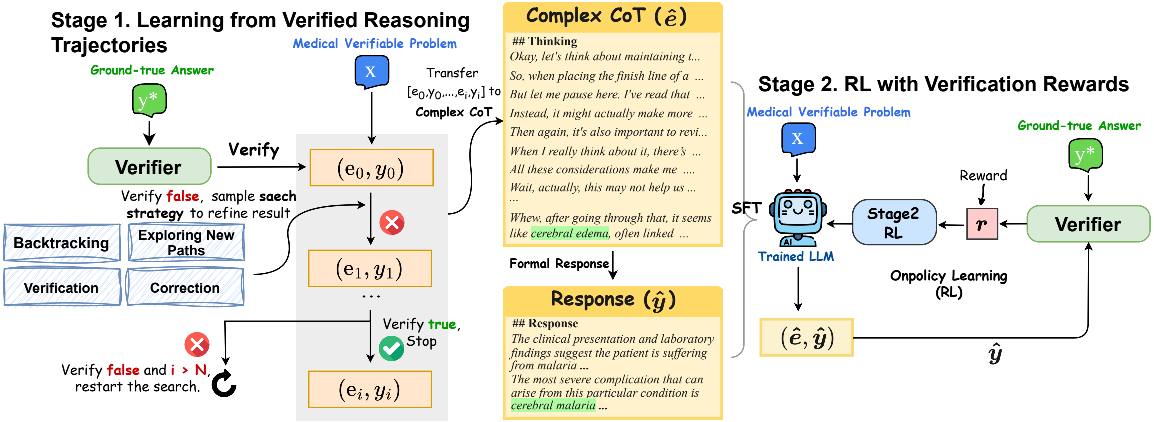


Figure 2: Demonstration of training LLMs for medical complex reasoning with Medical Verifiable Problems. **Left (Stage 1):** Searching for verified reasoning trajectories to fine-tune LLMs. **Right (Stage 2):** Using the verifier to enhance complex reasoning via reinforcement learning.

ing sets of *MedQA-USMLE* (Jin et al., 2021) and *MedMcQA* (Pal et al., 2022a).

Constructing Verifiable Problems However, these medical questions are closed-set (multiple-choice), making it easy for models to guess the correct option. Additionally, some questions are not suitable due to they may lack a unique correct answer for verification or are too simple to require reasoning. We address this by selecting and processing questions as follows:

1. **Selecting Challenging Questions** We removed questions that three small LLMs (Gemma2-9B (Team et al., 2024), LLaMA-3.1-8B (Dubey et al., 2024), Qwen2.5-7B (Team, 2024a)) all answered correctly and discarded short questions to retain those requiring more reasoning.
2. **Ensure Unique Answers:** We excluded questions asking for "incorrect options" or with multiple correct answers. A LLM (GPT-4o) is further employed to remove questions where the correct answer might not be unique or could be ambiguous.
3. **Reformatting to Open-Ended Formal:** Using LLMs (GPT-4o), We reformatted each closed-set question into open-ended problem an open-ended problem x and a ground-truth answer y^*, as shown in Figure 1.

The prompt used for filtering and processing can be found in Appendix L. After this filtering and processing, we ultimately constructed a dataset of 40K Medical Verifiable Problems denoted as $\mathcal{D} = \{(x, y^*)\}$, where x is a verifiable problem and y^* the ground-truth answer.

Medical Verifier With these verifiable problems, we propose a verifier to assess the correctness of model outputs. Given a medical verifiable problem x, the model generates a Chain-of-Thought (CoT) e and a result y. The verifier checks y against the ground-truth answer y^* and provides binary feedback as:

$$\text{Verifier}(y, y^*) \in \{\text{True}, \text{False}\}$$

Unlike mathematical problems, we use LLMs (GPT-4o) as the verifier, prompting it to perform verification with the detailed prompt provided in Appendix M. Due to the prevalence of aliases in the medical domain, exact match methods (Luong et al., 2024; Gandhi et al., 2024), which are commonly applied in mathematics, are not feasible. Experiments in Section 4.2 confirm this and demonstrate the reliability of the LLM-based verifier.

3 Methodology

In this section, we describe the method for training LLMs to perform medical complex reasoning. Complex reasoning refers to longer Chains-of-Thought (CoT) coupled with reflective behaviors. The formal definition of complex reasoning is provided in Appendix I. As shown in Figure 1, the method consists of two stages based on the Medical Verifiable Problems.

3.1 Stage 1: Learning from Verified Reasoning Trajectories

Searching for Verified Trajectories Given a verifiable medical problem as a tuple (x, y^*), i.e. (question, ground-true answer), the LLM (GPT-4o)

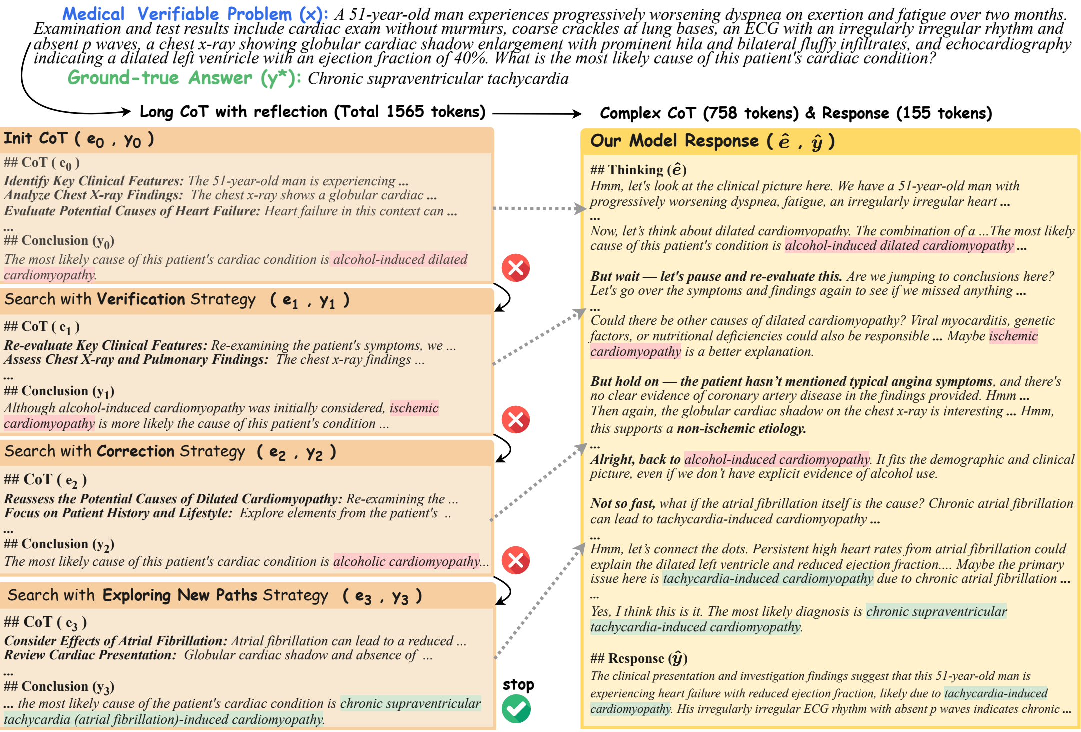


Figure 3: Example of data synthesis. **Left:** Strategy search on medical verifiable problems until the answer is verified. **Right:** Transfer the entire search process into complex CoTs to mimic human complex thinking. The complex CoTs and responses are used to train the model to adopt *thinks-before-it-answers* behavior akin to o1.

generates an initial CoT e_0 and answer y_0:

$$e_0, y_0 = \mathsf{LLM}_{\text{init}}(x)$$

The verifier checks if y_0 matches y^*. If incorrect, the model iteratively refines the answer by applying a randomly selected search strategy $k \in \mathcal{K}$ on prior thoughts $[e_0, y_0, \ldots, e_{i-1}, y_{i-1}]$, producing new reasoning e_i and new answer y_i:

$$e_i, y_i = \mathsf{LLM}_{k_i}(x, [e_0, y_0, \ldots, e_{i-1}, y_{i-1}])$$

where i denotes the i-th iteration. We define four search strategies $\mathcal{K}$ to guide the refinement process:

- **Exploring New Paths** The LLM explores a new approach e_i , distinct from prior $e_0, \ldots, e_{i-1}$, to derive a new answer y_i.
- **Backtracking** The LLM revisits a previous reasoning process e_j, y_j, where $j < i-1$, and continues reasoning from there. Note that Backtracking is sampled only if $i \geq 2$.
- **Verification** The LLM evaluates the current reasoning $e_{i-1}$and result y_{i-1}, providing a validation process e_i and the verified result y_i.
- **Corrections** The LLM critiques and corrects the current reasoning e_{i-1}, yielding a revised reasoning e_j and answer y_i.

The process iterates until y_i is verified as correct. If the maximum iteration count $N = 3$ are reached, the search restarts. Each data point (x, y^*) is given up to $T = 3$ attempts; if all fail, the data point is discarded. The prompts for search reasoning trajectories and search statistics can be found in Appendix N and Appendix D.

Constructing SFT Training Data When a successful trajectory $[e_0, y_0, \ldots, e_i, y_i]$ is found, it is reformatted into a coherent, natural language reasoning process $\hat{e}$ (*Complex CoT*):

$$\hat{e} = \mathsf{LLM}_{\text{Reformat}}([e_0, e_1, \ldots, e_i, y_i])$$

As shown in Figure 3, this reformatting avoids rigid structures to reduce token usage and employs smooth transitions (e.g., "hmm," "also," "wait") to mimic human reasoning processes. Additionally, $\hat{e}$ preserves the entire self-reflective thinking process of $[e_0, e_1, \ldots, e_i, y_i]$. This Complex CoT ($\hat{e}$) indicates the thinking process of complex reasoning. The model then generates a formal response $\hat{y}$ for question x using the conclusion of $\hat{e}$:

$$\hat{y} = \mathsf{LLM}_{\text{Response}}(x, \hat{e})$$

The prompt used for constructing SFT data can be found in Appendix O.

Algorithm 1: Training LLMs for Medical Complex Reasoning

```
Require: Medical Verifiable Problems
  D = {(x, y*)}, a Verifier, an LLM (GPT-4o) for
  synthesizing reasoning trajectories, search strategies
  K, max search depth N, max search attempts T,
  and initial policy π_θ.
D_Search, D_RL ← Split(D)
D_SFT ← ∅
// Stage 1: Learning Complex Reasoning
for (x, y*) ∈ D_Search do
    for j ← 1 to T do
        e_0, y_0 ← LLM_init(x)
        i ← 0
        if not Verifier(y_0, y*) then
            for i ← 1 to N do
                k_i ∼ K
                e_i, y_i ←
                  LLM_{k_i}(x, [e_0, y_0, ..., e_{i-1}, y_{i-1}])
                if Verifier(y_i, y*) then
                    break
        if Verifier(y_i, y*) then
            ê ← LLM_Reformat([e_0, y_0, ..., e_i, y_i])
            ŷ ← LLM_Response(ê)
            D_SFT ← D_SFT ∪ {(x, ê, ŷ)}
            break
// SFT
for (x, ê, ŷ) ∈ D_SFT do
    L_SFT(θ) ← − log π_θ(ê, ŷ | x)
    θ ← UpdateParameters(L_SFT(θ), θ)
// Stage 2: Enhance Reasoning with RL
π_ref ← π_θ
for (x, y*) ∈ D_RL do
    ê, ŷ ∼ π_θ(x)
    // Reward
    r ← Rule (Verifier (ŷ, y*)) −
      βKL (π_θ (· | x) || π_ref (· | x))
    θ ←
     UpdateParameters (L_RL (x, ê, ŷ, r, π_ref, π_θ) , θ)
return π_θ
```

Supervised Fine-Tuning (SFT) Finally, we synthesize 20K SFT data $D_{\text{SFT}} = \{(x, \hat{e}, \hat{y})\}$ from the Medical Verifiable Problems $\mathcal{D} = \{(x, y^*)\}$ using GPT-4o. D_{SFT} is used to fine-tune LLMs to generate a complex CoT $\hat{e}$ followed by a formal response $\hat{y}$, behaving similarly to OpenAI-o1 and DeepSeek-R1. This fine-tuning serves as a warm-up to train the model to perform complex reasoning.

3.2 Stage 2: RL with Verification Rewards

In this stage, we further enhance the complex reasoning skills using reinforcement learning (RL). While the LLM learns successful reasoning trajectories in Stage 1, these paths, derived via search, may not be optimal. On-policy learning in Stage 2 refines complex reasoning through verification feedback.

Verification Rewards The verification of Medical Verifiable Problems provides an important reward for LLMs to optimize their reasoning trajectories. Given a verifiable problem x and the generated response $(\hat{e}, \hat{y})$, the reward is assigned as follows:

$$r'(x, \hat{y}, y^*) = \begin{cases} 1 & \text{if verifier}(\hat{y}, y^*) = \text{True} \\ 0.1 & \text{if verifier}(\hat{y}, y^*) = \text{False} \\ 0 & \text{if } \hat{y} = \text{null} \end{cases}$$

Following (Riedmiller et al., 2018; Trott et al., 2019; Luong et al., 2024), correct answers receive a reward of 1, incorrect answers receive 0.1, and responses that lack *think-before-answering* behavior receive 0. Additionally, following related works, the total reward combines this function score with the Kullback-Leibler (KL) divergence between the learned RL policy π_θ and the initial policy π_{ref}, scaled by a coefficient β:

$$r(x, \hat{y}, y^*) = r'(x, \hat{y}, y^*) - \beta \text{KL}(\theta)$$

to stabilize training with sparse rewards (Luong et al., 2024).

Reinforcement Learning For RL, We use the Proximal Policy Optimization (PPO) (Schulman et al., 2017) algorithm with a clipped objective. The fine-tuned model serves as the policy model π_θ. Training is conducted on the remaining Medical Verifiable Problems $\mathcal{D}_{\text{RL}} = \{(x, y^*)\}$. The policy samples responses $(\hat{e}, \hat{y})$ for input x, computes the reward, and updates parameters θ.

The full training process for both stages is summarized in Algorithm 1.

4 Experiments

4.1 Experimental Setup

Training Data We constructed a 40K Medical Verifiable Problems $\mathcal{D} = \{(x, y^*)\}$ from the training sets of *MedQA-USMLE* (Jin et al., 2021) and *MedMCQA* (Pal et al., 2022b). Of this, 20K is used for SFT in stage 1 and 20K for RL in stage 2. Additionally, 4K unconverted data (close-set questions with option answers) from $\mathcal{D}$ are included to enhance generalization. All data were strictly screened to avoid contamination with the evaluation data using the filtering method of Med-PaLM2 (Singhal et al., 2023b) (filtering overlaps of 64 consecutive characters).

	MedQA	MedMCQA	PubMedQA	MMLU-Pro		GPQA		Avg.
				Health	Biology	Genetics	Molecular Biology	
∼ 8B Large Language Models								
BioMistral-7B	45.0	40.2	66.9	27.4	49.2	28.6	38.5	42.3
OpenBioLLM-8B	57.7	54.1	74.1	38.4	52.4	43.7	39.6	51.4
UltraMedical-8B	<u>71.1</u>	<u>58.3</u>	<u>77.4</u>	<u>55.1</u>	66.7	41.2	48.4	59.7
Mistral-7B-Instruct	48.2	44.6	59.5	33.7	53.6	30.0	46.1	45.1
Yi-1.5-9B-Chat	50.8	48.7	69.8	43.4	65.6	42.5	48.1	52.7
LLaMA-3.1-8B-Instruct	58.7	56.0	75.2	52.7	64.6	33.8	46.8	55.4
GLM-4-9B-Chat	58.9	49.8	73.5	45.5	65.4	**53.8**	41.6	55.5
DeepSeek-R1-Distill-Llama-8B	54.4	49.5	74.4	45.2	66.4	41.2	59.0	55.8
Qwen2.5-7B-Instruct	57.0	55.6	72.7	50.6	<u>70.2</u>	36.2	49.7	56.0
Gemma2-9B	61.8	55.9	63.3	<u>55.1</u>	**74.9**	35.0	<u>57.4</u>	57.6
HuatuoGPT-o1-8B	**73.5**	**61.6**	**79.5**	**59.5**	69.6	<u>48.1</u>	**59.0**	**64.4**
w/o Stage2 (RL)	69.2	57.7	77.6	53.7	65.8	41.0	53.6	<u>59.8</u>
10B to 100B Large Language Models								
UltraMedical-70B	82.2	71.8	78.4	64.8	71.1	33.8	62.9	66.4
OpenBioLLM-70B	76.1	<u>74.7</u>	79.2	68.8	76.7	38.8	54.8	67.0
DeepSeek-67B-Chat	57.1	51.7	76.1	46.9	66.2	40.0	51.0	55.6
Yi-1.5-34B-Chat	59.5	56.7	74.3	52.8	71.0	32.5	56.8	57.7
Gemma2-27B	65.4	60.2	72.6	61.1	76.2	32.5	61.6	61.4
Qwen2.5-72B-Instruct	72.7	66.2	71.7	65.3	78.8	41.2	56.8	64.7
QwQ-32B-Preview	72.3	65.6	73.7	62.0	78.1	37.5	64.5	64.8
Llama-3.1-70B-Instruct	78.4	72.5	78.5	68.2	<u>80.8</u>	52.5	61.6	70.3
DeepSeek-R1-Distill-Llama-70B	<u>85.6</u>	74.3	<u>80.0</u>	<u>70.7</u>	80.6	43.8	<u>65.2</u>	71.4
HuatuoGPT-o1-70B	**87.9**	**77.7**	**80.4**	**70.9**	**82.7**	**56.0**	**66.4**	**74.6**
w/o Stage2 (RL)	83.9	73.1	78.7	70.3	79.9	<u>54.1</u>	64.1	<u>72.0</u>

Table 1: Main Results on Medical Benchmarks. LLMs with are specifically trained for the medical domain, and indicates LLMs training for long chain-of-thought reasoning. "w/o" means "without". Within each segment, **bold** highlights the best scores, and <u>underlines</u> indicate the second-best.

Model Training Using the proposed method, we train our models **HuatuoGPT-o1-8B** and **HuatuoGPT-o1-70B** based on *LLaMA-3.1-8B-Instruct* and *LLaMA-3.1-70B-Instruct*, respectively. In Stage 1, the models are fine-tuned on the $\mathcal{D}_{\mathrm{SFT}}$ for 3 epochs with a learning rate of 5e-6 and a batch size of 128. In Stage 2, we employ PPO for RL with a learning rate of 5e-7, a batch size of 128, and β set to 0.03. The PPO parameters are set as: 3 PPO epochs, a discount factor 1.0, a value coefficient 1.0, and a clip range 0.2.

Baselines We compare our models with three types of open-source LLMs: **1) General LLMs:** Qwen-2.5 (Yang et al., 2024), LLaMA-3.1 (Dubey et al., 2024), Gemma 2 (Team et al., 2024), Yi (Young et al., 2024), Mistral (Jiang et al., 2023), GLM-4 (Zeng et al., 2023); **2) o1-like LLMs:** DeepSeek-R1-Distill (Guo et al., 2025) and QwQ (Team, 2024b); and **3) Medical-Specific LLMs:** UltraMedical (Zhang et al., 2024b), OpenBioLLM (Pal and Sankarasubbu, 2024), and BioMistral (Labrak et al., 2024).

Benchmarks We evaluate on standard medical benchmarks: *MedQA* (USMLE test set) (Jin et al., 2021), *MedMCQA* (validation set) (Pal et al., 2022a), and *PubMedQA* (test set) (Jin et al., 2019). Aditionally, we evaluated the medical sections of some challenging LLM benchmarks, including the health and biology tracks of *MMLU-Pro* (Wang et al., 2024c), and the genetics and molecular biology tracks of *GPQA* (Rein et al., 2023), using the main set with the multiple-choice setting. Due to the limited number of *GPQA* questions, we ran this evaluation 5 times and averaged the results.

4.2 Experimental Results

Main Results We evaluated LLMs with similar parameter sizes on medical benchmarks, as shown in Table 1. The results indicate that prior medical-specific LLMs, like UltraMedical, excel on traditional medical benchmarks (MedQA, MedMCQA, PubMedQA) but perform moderately on the more challenging datasets (GPQA and MMLU-Pro). Furthermore, o1-like models demonstrate superior performance (e.g., Deepseek-R1-Distill-70B outperforms LLaMA-70B, and QwQ-32B surpasses Qwen-72B), suggesting that enhancing reasoning capabilities improves medical capabilities.

Overall, our model, HuatuoGPT-o1, excels

	MedQA	MedMCQA	PubMedQA	MMLU-Pro (Med✚)	GPQA (Med✚)
			Baseline LLMs		
LLaMA-3.1-8B-Instruct	58.7	56.0	75.2	58.2	44.1
			Fine-Tuned Baseline		
SFT w/ Original Exam Data of $\mathcal{D}$	60.0	55.5	74.1	54.3	46.9
			Effectiveness of Complex Chain-of-Thought (CoT)		
SFT w/o ~~CoT~~ (only $\hat{y}$)	65.2	58.1	75.4	58.5	48.7
SFT w/ Simple CoT (x_0, y_0)	66.6	**59.2**	75.4	57.0	46.7
SFT w/ Complex CoT $(\hat{x}, \hat{y})$	**69.2**	57.7	**77.6**	**59.2**	**51.1**
			Effectiveness of RL		
SFT w/o ~~CoT~~ + RL w/ PPO	66.4	58.6	76.3	60.1	49.8
SFT w/ Simple CoT + RL w/ PPO	68.7	58.4	77.5	60.2	53.1
SFT w/ Complex CoT + RL w/ PPO	**73.5**	**61.6**	**79.5**	**64.2**	**56.8**
			Comparison of Different RL Algorithms		
SFT w/ Complex CoT + RL w/ DPO	72.0	58.6	77.0	60.5	52.7
SFT w/ Complex CoT + RL w/ RLOO	71.3	60.2	78.0	60.9	**58.4**
SFT w/ Complex CoT + RL w/ PPO	**73.5**	**61.6**	**79.5**	**64.2**	56.8

Table 2: The results of ablation experiments on *HuatuoHPT-o1-8B*. (Med✚) indicates that only the medical-related parts are evaluated. "w/o" and "w/" denote "without" and "with". "Original Exam Data" refers to original multiple-choice questions used for medical verifiable problems D. **Bold** highlights the best scores in each segment.

across all datasets. The 8B version outperforms the base model (LLaMA-3.1-8B-Instruct) by 8 points in overall evaluation. Moreover, our 70B model surpasses other compared LLMs, clearly demonstrating the effectiveness of our approach. This emphasizes the value of domain-specific optimization over basic distillation methods, as seen in models like Deepseek-R1-Distill.

Ablation Study We conducted an ablation study on the 8B model to analyze the impact of Complex-CoT and RL. The results, shown in Table 2, reveal the following insights:

1. Fine-tuning with Complex CoT Helps We examined the impact of different types of Chain-of-Thought (CoT). The results show that direct learning of the response ($\hat{y}$) performs the worst, while simple CoT (y_0, e_0) provides only minimal benefits. In contrast, Complex CoT $(\hat{e}, \hat{y})$ significantly improves performance by an average of 4.3 points. This highlights the importance of teaching models long, reflective reasoning processes.

2. LLMs with Complex Reasoning Benefit More Than Vanilla LLMs We compared the RL improvements under different CoT strategies, as shown in Table 3. The results reveal that Complex CoT, which involves much longer reasoning paths (averaging 712 tokens), yields a significantly greater performance gain (3.6 points) compared to simple CoT (2.6) and no CoT (1.1). This suggests that longer self-play reasoning paths provide richer thought processes and feedback, enabling the model to discover higher-reward solutions.

	# Avg. Generated Tokens	Δ Avg. Gain from RL
Direct Response ($\hat{y}$)	82	1.1
Simple CoT (x_0, y_0)	281	2.6
Complex CoT $(\hat{x}, \hat{y})$	**712**	**3.6**

Table 3: Comparison of RL improvement. "# Avg. Tokens" indicates the average number of response tokens. **Δ** represents the gain from RL, as detailed in Table 1.

3. PPO Outperforms DPO and RLOO Using the same reward function, we further compared different RL-related algorithms, including the preference learning algorithm DPO (Rafailov et al., 2024) and the REINFORCE-style algorithm RLOO (Ahmadian et al., 2024). Detailed implementation information is provided in Appendix H. Comparing PPO, RLOO, and DPO, we find that PPO performs best, followed by RLOO and DPO. The weaker performance of DPO is likely due to its off-policy nature, while PPO benefits from its use of value models, despite higher memory consumption.

	MedQA	MedMCQA	PubMedQA	MMLU-Pro (Med)	GPQA (Med)	Avg. Gain
Training on LLaMA-3.1-8B-Instruct						
LLaMA-3.1-8B-Instruct	58.7	56.0	75.2	58.2	44.1	(0.0)
HuatuoGPT-o1-Llama-8B	72.6	60.4	79.2	63.1	57.5	(↑ 8.1)
Training on Qwen2.5-7B-Instruct						
Qwen2.5-7B-Instruct	58.7	55.6	72.7	60.3	46.9	(0.0)
HuatuoGPT-o1-Qwen-7B	72.0	62.5	78.6	68.3	54.4	(↑ 8.3)

Table 4: Performance improvement on different backbones using the proposed method.

Reliability of the Verifier The verifier plays a critical role in our methods. To evaluate its reliability, we manually verified 200 scoring instances sampled from Stage 1 and Stage 2. As shown in Figure 4, the GPT-4o (we used) achieved 96.5% accuracy in Stage 1 and 94.5% in Stage 2, demonstrating its reliability. In contrast, the Exact Match method (Luong et al., 2024), which is rule-based and widely used in mathematical verification, performed significantly worse, with accuracies of only 70.5% in Stage 1 and 74.5% in Stage 2. This underscores the critical role of LLM-based verifiers. Furthermore, we fine-tuned an 8B verifier on LLaMA-3.1-8B with 20K scoring samples for low-cost verification for the research community. The fine-tuned verifier also demonstrated reliability, achieving over 90% accuracy.

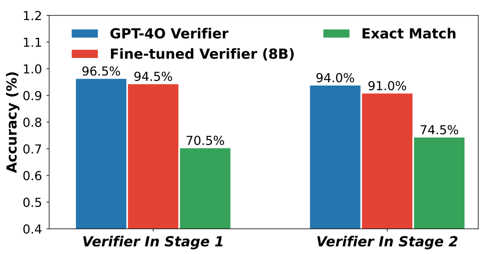


Figure 4: Accuracy of verifiers. Accuracy is based on 200 manually annotated samples.

	MedQA (Chinese)	CMExam	CMMLU (Med)
HuatuoGPT2-7B	73.7	67.4	58.4
Yi-1.5-9B-Chat	75.8	70.4	70.5
GLM-4-9B-Chat	75.6	70.5	69.1
DeepSeek-R1-Distill-Qwen-7B	83.2	77.8	75.3
Qwen2.5-7B-Instruct	83.9	77.0	77.2
HuatuoGPT-o1-7B-Chinese	**87.4**	**81.5**	**81.6**

Table 5: Cross-lingual adaptation (**Chinese**) results on Chinese medical benchmarks. **(Med)** indicates that only the medical portion is evaluated.

Domain Compatibility Our approach uses verifiable questions to enhance domain reasoning and is theoretically applicable across languages and fields. To validate this, we conducted experiments in the **Chinese** medical and **chemistry** domains. For **Chinese** adaptation, we built 40K Chinese Medical Verifiable Problems from the CMB (Wang et al., 2023c) and trained *HuatuoGPT-o1-7B-Chinese* on *Qwen2.5-7B-Instruct*. For **chemistry** adaptation, we created 15K Chemistry Verifiable Problems from SciKnowEval (Feng et al., 2024), mixed them with 20K medical questions, and trained *HuatuoGPT-o1-8B-Chem* on *LLaMA-3.1-8B-Instruct*. Implementation details are in Appendix K. As shown in Table 5 and Table 6, both models achieved notable improvements, highlighting our method's adaptability for other domains.

	ChemBench	MMLU-Pro (Chem)	GPQA (Chem)
LLaMA-3.1-8B-Instruct	55.1	42.2	29.9
DeepSeek-R1-Distill-Llama-8B	56.4	33.8	37.2
Gemma2-9B-it	58.0	45.5	42.6
Qwen2.5-7B-Instruct	58.3	65.4	37.4
HuatuoGPT-o1-8B-Chem	**61.4**	**68.5**	**44.8**

Table 6: Cross-domain adaptation (**Chemistry**) results on chemistry benchmarks. **(Med)** indicates that only the chemistry portion is evaluated.

Experiments with Other Backbones In addition to the *LLaMA-3.1* series, we also trained on *Qwen2.5-7B-Instruct* to assess the effectiveness of our method across different backbones. The results, presented in Table 8, demonstrate that our approach transfers effectively to other backbone LLMs.

5 Conclusion

This study advances the medical reasoning capabilities of LLMs. Firstly, we construct the medical verifiable problems and a medical verifier. This enabled a two-stage training process: (1) learning complex reasoning and (2) enhancing it through RL. We developed HuatuoGPT-o1, a medical LLM with *thinks-before-it-answers* behavior, achieving outstanding performance in medical benchmarks. Experiments show that complex reasoning improves medical problem-solving and benefits obviously from RL. Additional validation in Chinese medical contexts shows the method's adaptability to other fields. We believe our approach can enhance domain-specific reasoning beyond mathematics.

Limitations

Lack of Scalable Reinforcement Learning Appendix C outlines our RL training process. However, we did not attempt to scale RL training, as seen in in OpenAI-o1 (OpenAI, 2024) and Deepseek-R1 (Guo et al., 2025). The reason for this is the limited availability of verifiable problems. Additionally, we found that further training steps could degrade the model's performance. Therefore, scaling verifiable problems and stabilizing RL is an important direction, which we leave for future research.

API Costs for Verification This work utilizes GPT-4o as the verifier, which could incur significant API costs, making reproducing our work expensive. In response, we conducted an additional experiment to verify that a fine-tuned smaller verifier can achieve similar verification performance. We are also open-sourcing this smaller verifier to support research within the community.

Dependence on Exam Questions Our approach relies on exam questions to construct verifiable datasets, which requires collecting a large number of such questions. We have not yet explored synthesizing verifiable questions from other sources. For some non-medical domains, exam questions may be scarce. In the future, incorporating alternative sources for question synthesis could enhance the adaptability of our method.

Acknowledgements

This work was supported by Shenzhen Medical Research Fund (No.C2406002) from the Shenzhen Medical Academy of Research and Translation (SMART), the Shenzhen Science and Technology Program (JCYJ20220818103001002), 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), Shenzhen Stability Science Program 2023, and National Natural Science Foundation of China (NSFC) (72495131).

References

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. 2024. Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms. *arXiv preprint arXiv:2402.14740*.

Sina Alemohammad, Josue Casco-Rodriguez, Lorenzo Luzi, Ahmed Imtiaz Humayun, Hossein Babaei, Daniel LeJeune, Ali Siahkoohi, and Richard G. Baraniuk. 2024. Self-consuming generative models go MAD. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

Zhijie Bao, Wei Chen, Shengze Xiao, Kuang Ren, Jiaao Wu, Cheng Zhong, Jiajie Peng, Xuanjing Huang, and Zhongyu Wei. 2023. Disc-medllm: Bridging general large language models and real-world medical consultation. *Preprint*, arXiv:2308.14346.

Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. 2024. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*.

Angelica Chen, Jérémy Scheurer, Tomasz Korbak, Jon Ander Campos, Jun Shern Chan, Samuel R. Bowman, Kyunghyun Cho, and Ethan Perez. 2023a. Improving code generation by training with natural language feedback. *CoRR*, abs/2303.16749.

Junying Chen, Chi Gui, Anningzhe Gao, Ke Ji, Xidong Wang, Xiang Wan, and Benyou Wang. 2024a. Cod, towards an interpretable medical agent using chain of diagnosis. *arXiv preprint arXiv:2407.13301*.

Junying Chen, Chi Gui, Ruyi Ouyang, Anningzhe Gao, Shunian Chen, Guiming Hardy Chen, Xidong Wang, Ruifei Zhang, Zhenyang Cai, Ke Ji, et al. 2024b. Huatuogpt-vision, towards injecting medical visual knowledge into multimodal llms at scale. *arXiv preprint arXiv:2406.19280*.

Junying Chen, Xidong Wang, Ke Ji, Anningzhe Gao, Feng Jiang, Shunian Chen, Hongbo Zhang, Dingjie Song, Wenya Xie, Chuyi Kong, et al. 2023b. Huatuogpt-ii, one-stage training for medical adaption of llms. *arXiv preprint arXiv:2311.09774*.

Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, et al. 2023c. Meditron-70b: Scaling medical pretraining for large language models. *arXiv preprint arXiv:2311.16079*.

Daixuan Cheng, Shaohan Huang, and Furu Wei. 2023. Adapting large language models via reading comprehension. In *The Twelfth International Conference on Learning Representations*.

Clément Christophe, Praveen K Kanithi, Prateek Munjal, Tathagata Raha, Nasir Hayat, Ronnie Rajan, Ahmed Al-Mahrooqi, Avani Gupta, Muhammad Umar Salman, Gurpreet Gosal, et al. 2024. Med42–evaluating fine-tuning strategies for medical llms: Full-parameter vs. parameter-efficient approaches. *arXiv preprint arXiv:2404.14779*.

Zi-Yi Dou, Cheng-Fu Yang, Xueqing Wu, Kai-Wei Chang, and Nanyun Peng. 2024. Re-rest: Reflection-reinforced self-training for language agents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 15394–15411. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Kehua Feng, Keyan Ding, Weijie Wang, Xiang Zhuang, Zeyuan Wang, Ming Qin, Yu Zhao, Jianhua Yao, Qiang Zhang, and Huajun Chen. 2024. Sciknoweval: Evaluating multi-level scientific knowledge of large language models. *Preprint*, arXiv:2406.09098.

Kanishk Gandhi, Denise Lee, Gabriel Grand, Muxin Liu, Winson Cheng, Archit Sharma, and Noah D Goodman. 2024. Stream of search (sos): Learning to search in language. *arXiv preprint arXiv:2404.03683*.

Bofei Gao, Zefan Cai, Runxin Xu, Peiyi Wang, Ce Zheng, Runji Lin, Keming Lu, Junyang Lin, Chang Zhou, Wen Xiao, Junjie Hu, Tianyu Liu, and Baobao Chang. 2024. LLM critics help catch bugs in mathematics: Towards a better mathematical verifier with natural language feedback. *CoRR*, abs/2406.14024.

Melody Y. Guan, Manas Joglekar, Eric Wallace, Saachi Jain, Boaz Barak, Alec Heylar, Rachel Dias, Andrea Vallone, Hongyu Ren, Jason Wei, Hyung Won Chung, Sam Toyer, Johannes Heidecke, Alex Beutel, and Amelia Glaese. 2024. Deliberative alignment: Reasoning enables safer language models. *OpenAI Blog*.

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

Tianyu Han, Lisa C Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K Bressem. 2023. Medalpaca–an open-source collection of medical conversational ai models and training data. *arXiv preprint arXiv:2304.08247*.

Alexander Havrilla, Sharath Chandra Raparthy, Christoforos Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, and Roberta Raileanu. 2024. Glore: When, where, and how to improve LLM reasoning via global and local refinements. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Jiaxin Huang, Shixiang Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2023. Large language models can self-improve. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 1051–1068. Association for Computational Linguistics.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

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

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2021. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William W Cohen, and Xinghua Lu. 2019. Pubmedqa: A dataset for biomedical research question answering. *arXiv preprint arXiv:1909.06146*.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D. Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, Lei M. Zhang, Kay McKinney, Disha Shrivastava, Cosmin Paduraru, George Tucker, Doina Precup, Feryal M. P. Behbahani, and Aleksandra Faust. 2024. Training language models to self-correct via reinforcement learning. *CoRR*, abs/2409.12917.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. Biomistral: A collection of open-source pretrained large language models for medical domains. *arXiv preprint arXiv:2402.10373*.

Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. 2023. Cmmlu: Measuring massive multitask language understanding in chinese. *arXiv preprint arXiv:2306.09212*.

Junkai Li, Siyu Wang, Meng Zhang, Weitao Li, Yunghwei Lai, Xinhui Kang, Weizhi Ma, and Yang Liu. 2024. Agent hospital: A simulacrum of hospital with evolvable medical agents. *arXiv preprint arXiv:2405.02957*.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Junling Liu, Peilin Zhou, Yining Hua, Dading Chong, Zhongyu Tian, Andrew Liu, Helin Wang, Chenyu You, Zhenhua Guo, Lei Zhu, et al. 2024. Benchmarking large language models on cmexam-a comprehensive chinese medical exam dataset. *Advances in Neural Information Processing Systems*, 36.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. 2023. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *CoRR*, abs/2308.09583.

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

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 11048–11064. Association for Computational Linguistics.

Adrian Mirza, Nawaf Alampara, Sreekanth Kunchapu, Benedict Emoekabu, Aswanth Krishnan, Mara Wilhelmi, Macjonathan Okereke, Juliane Eberhardt, Amir Mohammad Elahi, Maximilian Greiner, Caroline T. Holick, Tanya Gupta, Mehrdad Asgari, Christina Glaubitz, Lea C. Klepsch, Yannik Köster, Jakob Meyer, Santiago Miret, Tim Hoffmann, Fabian Alexander Kreth, Michael Ringleb, Nicole Roesner, Ulrich S. Schubert, Leanne M. Stafast, Dinga Wonanke, Michael Pieler, Philippe Schwaller, and Kevin Maik Jablonka. 2024. Are large language models superhuman chemists? *Preprint*, arXiv:2404.01475.

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

Harsha Nori, Yin Tat Lee, Sheng Zhang, Dean Carignan, Richard Edgar, Nicolo Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, et al. 2023. Can generalist foundation models outcompete special-purpose tuning? case study in medicine. *arXiv preprint arXiv:2311.16452*.

Harsha Nori, Naoto Usuyama, Nicholas King, Scott Mayer McKinney, Xavier Fernandes, Sheng Zhang, and Eric Horvitz. 2024. From medprompt to o1: Exploration of run-time strategies for medical challenge problems and beyond. *arXiv preprint arXiv:2411.03590*.

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

OpenAI. 2024. Learning to reason with llms.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022a. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on health, inference, and learning*, pages 248–260. PMLR.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022b. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on Health, Inference, and Learning*, pages 248–260. PMLR.

Malaikannan Sankarasubbu Ankit Pal and Malaikannan Sankarasubbu. 2024. Openbiollms: Advancing open-source large language models for healthcare and life sciences.

Vimla L Patel, José F Arocha, and Jiajie Zhang. 2005. Thinking and reasoning in medicine. *The Cambridge handbook of thinking and reasoning*, 14:727–750.

Debjit Paul, Mete Ismayilzada, Maxime Peyrard, Beatriz Borges, Antoine Bosselut, Robert West, and Boi Faltings. 2024. REFINER: reasoning feedback on intermediate representations. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2024 - Volume 1: Long Papers, St. Julian's, Malta, March 17-22, 2024*, pages 1100–1126. Association for Computational Linguistics.

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

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.

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

Martin Riedmiller, Roland Hafner, Thomas Lampe, Michael Neunert, Jonas Degrave, Tom Wiele, Vlad Mnih, Nicolas Heess, and Jost Tobias Springenberg. 2018. Learning by playing solving sparse reward tasks from scratch. In *International conference on machine learning*, pages 4344–4353. PMLR.

Khaled Saab, Tao Tu, Wei-Hung Weng, Ryutaro Tanno, David Stutz, Ellery Wulczyn, Fan Zhang, Tim Strother, Chunjong Park, Elahe Vedadi, et al. 2024. Capabilities of gemini models in medicine. *arXiv preprint arXiv:2404.18416*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Yarin Gal, Nicolas Papernot, and Ross J. Anderson. 2023. The curse of recursion: Training on generated data makes models forget. *CoRR*, abs/2305.17493.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. 2017. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. 2023a. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, et al. 2023b. Towards expert-level medical question answering with large language models. *arXiv preprint arXiv:2305.09617*.

Yisheng Song, Ting Wang, Puyu Cai, Subrota K. Mondal, and Jyoti Prakash Sahoo. 2023. A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities. *ACM Comput. Surv.*, 55(13s):271:1–271:40.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Qwen Team. 2024a. Qwen2.5: A party of foundation models.

Qwen Team. 2024b. Qwq: Reflect deeply on the boundaries of the unknown.

Mohamad-Hani Temsah, Amr Jamal, Khalid Alhasan, Abdulkarim A Temsah, and Khalid H Malki. 2024. Openai o1-preview vs. chatgpt in healthcare: A new frontier in medical ai reasoning. *Cureus*, 16(10):e70640.

Alexander Trott, Stephan Zheng, Caiming Xiong, and Richard Socher. 2019. Keeping your distance: Solving sparse reward tasks using self-balancing shaped rewards. *Advances in Neural Information Processing Systems*, 32.

Haochun Wang, Chi Liu, Nuwa Xi, Zewen Qiang, Sendong Zhao, Bing Qin, and Ting Liu. 2023a. Huatuo: Tuning llama model with chinese medical knowledge. *Preprint*, arXiv:2304.06975.

Jun Wang, Meng Fang, Ziyu Wan, Muning Wen, Jiachen Zhu, Anjie Liu, Ziqin Gong, Yan Song, Lei Chen, Lionel M Ni, et al. 2024a. Openr: An open source framework for advanced reasoning with large language models. *arXiv preprint arXiv:2410.09671*.

Tianlu Wang, Ping Yu, Xiaoqing Ellen Tan, Sean O'Brien, Ramakanth Pasunuru, Jane Dwivedi-Yu, Olga Golovneva, Luke Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. 2023b. Shepherd: A critic for language model generation. *CoRR*, abs/2308.04592.

Xidong Wang, Guiming Hardy Chen, Dingjie Song, Zhiyi Zhang, Zhihong Chen, Qingying Xiao, Feng Jiang, Jianquan Li, Xiang Wan, Benyou Wang, et al. 2023c. Cmb: A comprehensive medical benchmark in chinese. *arXiv preprint arXiv:2308.08833*.

Xidong Wang, Nuo Chen, Junyin Chen, Yan Hu, Yidong Wang, Xiangbo Wu, Anningzhe Gao, Xiang Wan, Haizhou Li, and Benyou Wang. 2024b. Apollo: Lightweight multilingual medical llms towards democratizing medical ai to 6b people. *arXiv preprint arXiv:2403.03640*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023d. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. 2024c. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *arXiv preprint arXiv:2406.01574*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Sean Welleck, Ximing Lu, Peter West, Faeze Brahman, Tianxiao Shen, Daniel Khashabi, and Yejin Choi. 2023. Generating sequences by learning to self-correct. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Chaoyi Wu, Weixiong Lin, Xiaoman Zhang, Ya Zhang, Weidi Xie, and Yanfeng Wang. 2024. Pmc-llama: toward building open-source language models for medicine. *Journal of the American Medical Informatics Association*, page ocae045.

Zhiheng Xi, Senjie Jin, Yuhao Zhou, Rui Zheng, Songyang Gao, Jia Liu, Tao Gui, Qi Zhang, and Xuanjing Huang. 2023. Self-polish: Enhance reasoning in large language models via problem refinement. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 11383–11406. Association for Computational Linguistics.

Yunfei Xie, Juncheng Wu, Haoqin Tu, Siwei Yang, Bingchen Zhao, Yongshuo Zong, Qiao Jin, Cihang Xie, and Yuyin Zhou. 2024. A preliminary study of o1 in medicine: Are we closer to an ai doctor? *arXiv preprint arXiv:2409.15277*.

Fengli Xu, Qianyue Hao, Zefang Zong, Jingwei Wang, Yunke Zhang, Jingyi Wang, Xiaochong Lan, Jiahui Gong, Tianjian Ouyang, Fanjin Meng, et al. 2025. Towards large reasoning models: A survey of reinforced reasoning with large language models. *arXiv preprint arXiv:2501.09686*.

Guowei Xu, Peng Jin, Li Hao, Yibing Song, Lichao Sun, and Li Yuan. 2024a. Llava-o1: Let vision language models reason step-by-step. *arXiv preprint arXiv:2411.10440*.

Ming Xu. 2023. Medicalgpt: Training medical gpt model. `https://github.com/shibing624/MedicalGPT`.

Shaochen Xu, Yifan Zhou, Zhengliang Liu, Zihao Wu, Tianyang Zhong, Huaqin Zhao, Yiwei Li, Hanqi Jiang, Yi Pan, Junhao Chen, et al. 2024b. Towards next-generation medical agent: How o1 is reshaping decision-making in medical scenarios. *arXiv preprint arXiv:2411.14461*.

Wenda Xu, Guanglei Zhu, Xuandong Zhao, Liangming Pan, Lei Li, and William Wang. 2024c. Pride and prejudice: LLM amplifies self-bias in self-refinement. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 15474–15492. Association for Computational Linguistics.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, et al. 2024. Yi: Open foundation models by 01. ai. *arXiv preprint arXiv:2403.04652*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. 2022. Star: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, Zhiyuan Liu, Peng Zhang, Yuxiao Dong, and Jie Tang. 2023. GLM-130b: An open bilingual pre-trained model. In *The Eleventh International Conference on Learning Representations (ICLR)*.

Zhiyuan Zeng, Qinyuan Cheng, Zhangyue Yin, Bo Wang, Shimin Li, Yunhua Zhou, Qipeng Guo, Xuanjing Huang, and Xipeng Qiu. 2024. Scaling of search and learning: A roadmap to reproduce o1 from reinforcement learning perspective. *arXiv preprint arXiv:2412.14135*.

Di Zhang, Jianbo Wu, Jingdi Lei, Tong Che, Jiatong Li, Tong Xie, Xiaoshui Huang, Shufei Zhang, Marco Pavone, Yuqiang Li, et al. 2024a. Llama-berry: Pairwise optimization for o1-like olympiad-level mathematical reasoning. *arXiv preprint arXiv:2410.02884*.

Kai Zhang, Jun Yu, Eashan Adhikarla, Rong Zhou, Zhiling Yan, Yixin Liu, Zhengliang Liu, Lifang He, Brian Davison, Xiang Li, et al. 2023. Biomedgpt: a unified and generalist biomedical generative pre-trained transformer for vision, language, and multi-modal tasks. *arXiv e-prints*, pages arXiv–2305.

Kaiyan Zhang, Sihang Zeng, Ermo Hua, Ning Ding, Zhang-Ren Chen, Zhiyuan Ma, Haoxin Li, Ganqu

Cui, Biqing Qi, Xuekai Zhu, et al. 2024b. Ultramedical: Building specialized generalists in biomedicine. *arXiv preprint arXiv:2406.03949*.

Yuxiang Zhang, Shangxi Wu, Yuqi Yang, Jiangming Shu, Jinlin Xiao, Chao Kong, and Jitao Sang. 2024c. o1-coder: an o1 replication for coding. *arXiv preprint arXiv:2412.00154*.

Yu Zhao, Huifeng Yin, Bo Zeng, Hao Wang, Tianqi Shi, Chenyang Lyu, Longyue Wang, Weihua Luo, and Kaifu Zhang. 2024. Marco-o1: Towards open reasoning models for open-ended solutions. *arXiv preprint arXiv:2411.14405*.

Guorui Zheng, Xidong Wang, Juhao Liang, Nuo Chen, Yuping Zheng, and Benyou Wang. 2024a. Efficiently democratizing medical llms for 50 languages via a mixture of language family experts. *arXiv preprint arXiv:2410.10626*.

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

Tianyang Zhong, Zhengliang Liu, Yi Pan, Yutong Zhang, Yifan Zhou, Shizhe Liang, Zihao Wu, Yanjun Lyu, Peng Shu, Xiaowei Yu, et al. 2024. Evaluation of openai o1: Opportunities and challenges of agi. *arXiv preprint arXiv:2409.18486*.

Aojun Zhou, Ke Wang, Zimu Lu, Weikang Shi, Sichun Luo, Zipeng Qin, Shaoqing Lu, Anya Jia, Linqi Song, Mingjie Zhan, and Hongsheng Li. 2024. Solving challenging math word problems using GPT-4 code interpreter with code-based self-verification. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

A Ethical Statement

Although the proposed model is a medical LLM with complex reasoning capabilities, it may still produce content that includes hallucinations or inaccuracies. Therefore, the current model is not suitable for real-world applications. Consequently, we will impose strict limitations on the use of our model. The models are not permitted for use in clinical or other industry applications where such inaccuracies could lead to unintended consequences. We emphasize the ethical responsibility of users to adhere to these restrictions in order to safeguard the safety and integrity of their applications.

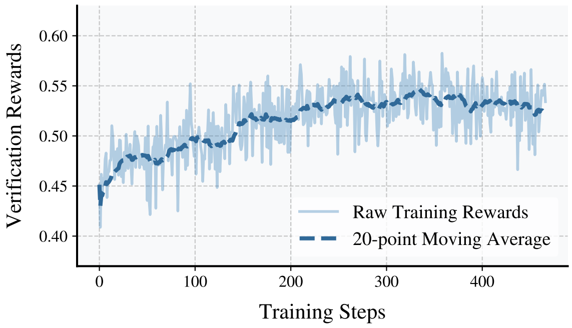
B Related Work

Research on o1 Recent studies have extensively analyzed the roadmap and core techniques of OpenAI's o1 (Qin et al., 2024; Wang et al., 2024a; Zeng et al., 2024), offering foundational insights into its architecture and methodology. Extensions such as LLaMA-Berry (Zhang et al., 2024a), LLaVA-o1 (Xu et al., 2024a), o1-Coder (Zhang et al., 2024c), and Marco-o1 (Zhao et al., 2024) have explored o1-like reasoning in various domains, including mathematics, vision-language integration, and open-ended problem-solving. However, these efforts have yet to address applications in medical or other highly specialized fields. In contrast, research focused on medicine (Xie et al., 2024; Nori et al., 2024; Temsah et al., 2024) highlights o1's potential for deliberate, chain-of-thought reasoning in healthcare contexts. Meanwhile, several o1-inspired models, such as DeepSeek-R1-Lite-Preview (Bi et al., 2024), QwQ (Team, 2024b), and Gemini-2.0 Flash Thinking (Team et al., 2023), have emerged. Despite their promise, most of these models remain closed-source, leaving substantial opportunities for further exploration and application of o1's capabilities across diverse fields.

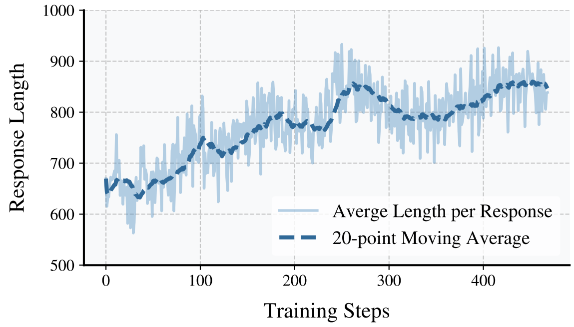
Medical LLMs The success of generalist LLMs has spurred interest in developing medical-specific LLMs to excel in the medical domain. Notably, the MedPaLM series (Singhal et al., 2023a,b) achieved over 60% accuracy on the MedQA benchmark, reportedly surpassing human experts. Previous medical LLMs typically follow two main approaches (Zhang et al., 2024b): **(1) Prompting Generalist LLMs** (Nori et al., 2023; Saab et al., 2024; Li et al., 2024; OpenAI, 2023; Chen et al., 2024a): This method employs task-specific prompts to adapt generalist models for medical applications. While efficient and training-free, it is inherently limited by the capabilities of the original LLMs. **(2) Further Training with Medical Data** (Xu, 2023; Wang et al., 2023a; Han et al., 2023; Wu et al., 2024; Pal and Sankarasubbu, 2024; Labrak et al., 2024; Bao et al., 2023; Zhang et al., 2023; Chen et al., 2024b; Wang et al., 2024b; Zheng et al., 2024a; Christophe et al., 2024): This involves training LLMs on medical pretraining corpora or medical instructions to embed medical knowledge and expertise. However, this always requires significant computational resources, such as the 1.4 billion and 3 billion training tokens used for Meditron (Chen et al., 2023c) and HuatuoGPT-II (Chen et al., 2023b). In contrast, our approach emphasizes enabling LLMs to excel in medical reasoning, offering a distinct solution.

Enhancing Reasoning in LLMs Chain-of-Thought (CoT) prompting enhances the reasoning capabilities of LLMs (Wei et al., 2022; Wang et al., 2023d), but scaling expert-labeled reasoning paths remains costly, especially for complex problems (Min et al., 2022; Song et al., 2023). To mitigate this, model-generated reasoning paths filtered through external supervision offer a partial solution (Zelikman et al., 2022; Huang et al., 2023), yet scalability challenges persist (Shumailov et al., 2023; Alemohammad et al., 2024). Reinforcement learning-based methods leveraging reward models or oracle functions show potential but often suffer from slow processing, high costs, and supervision bottlenecks (Lightman et al., 2024; Luo et al., 2023).

Complex Reasoning Developing models with reflective abilities like critique and self-correction has shown success in reasoning, planning, and coding tasks (Gandhi et al., 2024; Madaan et al., 2023; Chen et al., 2023a; Welleck et al., 2023; Xi et al., 2023; Paul et al., 2024), though underexplored in specialized domains like medicine. While prompting techniques can generate self-critical reasoning (Bai et al., 2022; Madaan et al., 2023), they struggle without reliable reward functions or verifiers, particularly in complex domains (Huang et al., 2024; Xu et al., 2024c). Fine-tuning and reinforcement learning methods offer solutions but require extensive human annotations or intricate reward designs (Wang et al., 2023b; Gao et al., 2024; Zhou et al., 2024; Havrilla et al., 2024). Additionally, self-training methods present a promising direction for developing self-correction capabilities (Welleck



(a) Average Training Reward



(b) Average Response Length

Figure 5: The average verification rewards and response length of HuatuoGPT-o1-8B PPO training.

Model	MedQA	MedMCQA	MMLU-Pro (Med)
LLaMA-3.1-8B (Backbone)	58.7	56.0	58.2
LLaMA-3.1-8B + **20K** SFT	69.0	57.9	59.4
Adding 20K fine-tuning data			
LLaMA-3.1-8B + **40K** SFT	69.6	58.1	59.8
Reinforcement learning with 20K data			
LLaMA-3.1-8B + **20K** SFT + **20K** RL	**72.6**	**60.4**	**63.1**

Table 7: Performance comparison with increasing amounts of SFT data.

	MedQA	MedMCQA	PubMedQA	MMLU-Pro (Med)	GPQA (Med)	Avg. Gain
Training on LLaMA-3.1-8B-Instruct						
LLaMA-3.1-8B-Instruct	58.7	56.0	75.2	58.2	44.1	(0.0)
HuatuoGPT-o1-Llama-8B	72.6	60.4	79.2	63.1	57.5	(↑ 8.1)
Training on Qwen2.5-7B-Instruct						
Qwen2.5-7B-Instruct	58.7	55.6	72.7	60.3	46.9	(0.0)
HuatuoGPT-o1-Qwen-7B	72.0	62.5	78.6	68.3	54.4	(↑ 8.3)

Table 8: Performance improvement on different backbones using the proposed method.

et al., 2023; Zheng et al., 2024b; Kumar et al., 2024).

C Reinforcement Learning Training

The PPO training process of HuatuoGPT-o1-8B is shown in Figure 5. As the training progresses, the accuracy of the verification gradually increases, and the response length also increases (mainly because the reasoning process takes longer). The rise in accuracy is likely attributed to a deeper reasoning process, involving more reflection and iteration. However, we also observe that after a certain number of steps, the model's performance begins to deteriorate, often producing responses that either fail to terminate or output disorganized, garbled content.

D Success Rate of Search Depth and Search Attempts

We analyze the distribution of search depths in the SFT dataset, as shown in Table 10. It can be observed that nearly 40% of the data requires reflection to obtain the correct answer. This highlights the significance of reflection, which can be further leveraged to fine-tune models for better comprehension of reflective reasoning.

Search Depth i	# Data Points	Proportion
0	12,677	62%
1	3,884	19%
2	2,494	12%
3	1,411	7%

Table 9: Distribution of search depths in the SFT dataset.

Despite this, failures can still occur even when the search depth reaches $N = 3$. To address this issue, we adopt a strategy where, upon reaching the maximum depth without finding the correct answer, the search restarts from scratch. This approach improves search efficiency and reduces computational costs. Our findings indicate that setting a reasonable number of search attempts significantly enhances the success rate of data construction, with only 4% of the data ultimately failing.

- 85% of data succeeds on the first attempt.
- 8% on the second attempt.
- 3% on the third attempt.
- Only approximately 4% is discarded.

Notably, increasing the search depth leads to longer input data, thereby increasing construction costs. Therefore, a balance between search depth and computational efficiency should be carefully considered.

E Experiments with Other Backbones

In addition to the original experiments based on the *LLaMA-3.1* series, we further validate our approach on *Qwen2.5-7B*, a different but comparable backbone, using the same training settings. The results are shown in Table 8. These results confirm that our method transfers well to other backbones. Additionally, all models based on both the LLaMA and Qwen series have been open-sourced.

F Does More SFT Data Matter?

Our experiments demonstrate the effectiveness of RL training, even with different SFT datasets. A natural question arises: *can increasing the amount of SFT data achieve similar effects?* We provide results using additional SFT data (40K, the full set of verifiable questions) as Table 7. The results indicate that increasing SFT data alone does not significantly improve performance. In contrast, the gain from RL remains substantial. We believe this is due to the inherent limitations of synthetic data—search-based augmentation does not necessarily yield the optimal solution. Meanwhile, self-learning via RL enables the model to discover better reasoning pathways.

G Increasing Search Depth

The search depth is adjustable, and users can set it to 11 iterations or more using our provided code. It is important to note that we employ a stream search approach (Gandhi et al., 2024), not a tree search. This means that each search iteration requires the complete history of previous searches as input, making the computational cost proportional to the search depth. To reduce costs, we set the depth to 3 while employing multiple resampling attempts.

Nonetheless, we tested 4K examples with different synthesis lengths:

Iteration Depth (4K Data)	Thinking (Length)	MedQA (SFT result)
Default (3)	564	64.6
6	977↑	67.1↑

Table 10: Effect of increasing search depth on inference length and fine-tuning performance.

The results show that increasing search depth leads to longer inference chains, which in turn improves fine-tuning performance.

H Settings of other RL training

we further compared different RL-related algorithms with PPO. Specifically, we employed the preference-learning algorithm DPO and the REINFORCE-style algorithm RLOO.

DPO For DPO, we had the model generate five answers for each question offline and used a verifier to identify pairs of one correct and one incorrect answer. If no such pairs were found, the data was discarded. Verified correct answers were used as positive examples, while failed verifications served as negative examples for training DPO. The hyperparameters for DPO training were set as follows: learning rate of 1e-6, batch size of 128, and a regularization parameter of 1.

RLOO For RLOO, we used the same reward function as PPO. The parameters were also identical to those of PPO, with an additional parameter rloo_k set to 2.

I Definition of Complex Reasoning

In this paper, complex reasoning refers to the process of generating long chains of thought (CoT) to replicate human-like thinking processes, such as reflection (Jaech et al., 2024; Xu et al., 2025). Reflection, in this context, means that LLMs assess their own generated answers and refine them if necessary (Dou et al., 2024). LLMs equipped with complex reasoning will perform such human-like thinking processes before providing their final response, such as models like OpenAI-o1 and DeepSeek-R1.

Model	MedQA (%)	MedMCQA (%)	Generated Tokens / Question	Time / Question
LLaMA-3.1-70B-Instruct	78.4	72.5	117	90ms
DeepSeek-R1-Distill-70B	85.6	74.3	988	421 ms
HuatuoGPT-o1-70B	**88.1**	**77.6**	664	241ms

Table 11: Performance and efficiency comparison of 70B-scale models. All models were tested using `vLLM` with tensor parallelism on a cluster of $4\times$ NVIDIA A800 80GB GPUs.

J Efficiency Comparison

We acknowledge the importance of evaluating both computational efficiency and performance. Table 11 presents a comparison between our model **HuatuoGPT-o1-70B** and two other strong baselines at the 70B scale: **LLaMA-3.1-70B-Instruct** and **DeepSeek-R1-Distill-LLaMA-70B**, evaluated on the MedQA and MedMCQA benchmarks.

Findings.

- **LLaMA-3.1-70B-Instruct** offers the lowest latency and fewest generated tokens per question but underperforms on both benchmarks.
- **DeepSeek-R1-Distill-70B** achieves decent performance but with significant computational cost—over $4\times$ the tokens and latency of LLaMA.
- **HuatuoGPT-o1-70B** provides a favorable trade-off, outperforming all baselines in accuracy while maintaining reasonable efficiency.

This analysis confirms that HuatuoGPT-o1-70B delivers strong medical QA performance without excessive computational overhead. We will include this comparison in the revised version of the paper.

K Domain Adaptation Beyond English Medical Domains

K.1 Chinese Domain Adaptation

Model Training For the Chinese medical domain, we replaced the exam questions in the CMB training set with verifiable medical questions in Chinese. Following the same training process used for the English version of HuatuoGPT-o1, we developed *HuatuoGPT-o1-7B-Chinese*, which is based on the *Qwen2.5-7B-Instruct* model.

Chinese Medical Evaluation To evaluate the model's performance in the Chinese medical domain, we assessed it using three Chinese medical benchmarks: the Chinese test set from MedQA (MCMLE) (Jin et al., 2021), the CMExam test set (Liu et al., 2024), and the medical section of the Chinese general benchmark CMMLU (Li et al., 2023). The CMMLU benchmark includes tracks such as **clinical knowledge**, **agronomy**, **college medicine**, **genetics**, **nutrition**, **Traditional Chinese Medicine**, and **virology**.

Comparison Models We compared the performance of our model with three general-purpose Chinese language models of similar size: Qwen2.5 (Team, 2024a), GLM-4 (Zeng et al., 2023), and Yi-1.5 (Young et al., 2024). Additionally, we included a comparison with a specialized Chinese medical model, HuatuoGPT-2-7B (Chen et al., 2023b).

K.2 Chemistry Domain Adaptation

To validate the effectiveness of our approach in non-medical domains, we focused on the chemistry domain.

Model Training For the chemistry domain, we obtained 20,000 chemistry-related questions from the SciKnowEval dataset (Feng et al., 2024) and selected challenging problems to build 15,000 verifiable chemistry questions. Due to the limited number of chemistry questions, we supplemented them with 20,000 existing medical verifiable questions to develop *HuatuoGPT-o1-7B-Chem*, built on the *LLaMA-3.1-8B-Instruct* model.

Chemistry Domain Evaluation To assess the chemistry capabilities, we evaluated the models on three chemistry benchmarks: 1) ChemBench (Mirza et al., 2024), a comprehensive evaluation of chemistry capabilities, reporting the accuracy across all questions; 2) the chemistry track of the MMLU-Pro test set; 3) the high-level chemistry track of GPQA, where we used the main set and reported the accuracy for its multiple-choice formal.

Comparison Models We primarily compared the performance of advanced LLMs with similar parameters, including *Gemma2-9B-it*, *Qwen2.5-7B-Instruct*, *Deepseek-R1-Distll-Llama-8B*, and *Llama-3.1-8B-Instruct*.

L Constructing Medical Verifiable Problems

To construct Medical Verifiable Problems, we begin by employing small models and rule-based methods to identify challenging questions. Subsequently, we leverage GPT-4o to perform data filtering, isolating questions that have been suitably transformed. The prompt used for this data filtering process is illustrated in Figure 6. After selecting appropriate data, we reformat multiple-choice medical exam questions into open-ended verifiable problems using the prompt provided in Figure 7.

The prompt for filtering Multiple-choice Questions

<Multiple-choice Question>
`{Question}`
`{Options}`
Correct Answer: `{Answer}`
</Multiple-choice Question>

You are an expert in filtering and evaluating multiple-choice questions for advanced reasoning tasks. Your job is to evaluate a given question and determine whether it meets the following criteria:
1. **Depth of Reasoning:** The question should require deeper reasoning. If the question appears too simple, mark it as "Too Simple."
2. **Unambiguous Correct Answer:** The question must have a unique and unambiguous correct answer. If the question asks for "incorrect options" or allows for multiple correct answers, mark it as "Ambiguous Answer."
3. **Open-Ended Reformulation Feasibility:** The question should be suitable for reformatting into an open-ended format. If the question cannot be easily reformulated into an open-ended problem and a clear ground-truth answer, mark it as "Not Reformulatable."

For each question, provide one of the following evaluations:
- "Pass" (The question meets all the criteria.)
- "Too Simple"
- "Ambiguous Answer"
- "Not Reformulatable"

Figure 6: The prompt for filtering Multiple-choice Questions. Here, `{Question}` and `{Options}` represents the multiple-choice question and options, and `{Answer}` represents the correct option for the multiple-choice question.

The prompt for reformatting multiple-choice questions to open-ended verifiable problems

I will provide you with a multiple-choice question, and your task is to rewrite it into an open-ended question, along with a standard answer. The requirements are:

1. The question must be specific, targeting the point being tested in the original multiple-choice question. Ensure it is open-ended, meaning no options are provided, but there must be a definitive standard answer.
2. Based on the correct answer from the original question, provide a concise standard answer. The answer should allow for precise matching to determine whether the model's response is correct.

Here is the multiple-choice question for you to rewrite:
<Multiple-choice Question>
`{Question}`
`{Options}`
Correct Answer: `{Answer}`
</Multiple-choice Question>

Please output the result in the following JSON format:
```json
{{
"Open-ended Verifiable Question": "...",
"Standard Answer": "..."
}}
```

Figure 7: The prompt for reformatting multiple-choice questions to open-ended verifiable problems. Here, `{Question}` and `{Options}` represents the multiple-choice question and options, and `{Answer}` represents the correct option for the multiple-choice question.

M The Prompt of Verifier

GPT-4o serves as the verifier to assess the correctness of model-generated outputs. Using the prompt depicted in Figure 8, we present GPT-4o with both the model's output and the ground-truth answer to evaluate the correctness of the response. The verifier returns a Boolean value: **True** if the response is accurate and **False** otherwise.

The Prompt for Verifier

<Model Response>
`{Model Response}`
</Model Response>

<Reference Answer>
`{Ground-true Answer}`
</Reference Answer>

You are provided with a model-generated response (<Model Response>) and a reference answer (<Reference Answer>). Compare the model response with the reference answer and determine its correctness. Your task is to simply output "True" if the response is correct, and "False" otherwise.

Figure 8: The prompt for the GPT-4o verifier. `{Model Response}` represents the output of the model to be verified. `{Ground-true Answer}` represents the ground-truth answer for medical verifiable problems.

N Prompts for Searching Trajectories

This section outlines the prompts used for constructing complex Chain-of-Thought (CoT) reasoning pathways. Initially, a question x is presented to GPT-4o, which generates an initial CoT response using the prompt shown in Figure 9. If the verifier determines the response to be incorrect, GPT-4o employs one of several search strategies to iteratively refine the output until it is accurate. The prompts for these four search strategies — **Backtracking**, **Exploring New Paths**, **Correction**, and **Verification** — are detailed in Figures 11, 11, 12, and 13, respectively.

The prompt for initial CoT

<question>
{Question}
</question>

Please respond to the above question <question> using the Chain of Thought (CoT) reasoning method. Your response should consist of multiple steps, each of which includes three types of actions: **"Inner Thinking"**, **"Final Conclusion"**, and **"Verification"**:

- **'Inner Thinking'**: This is the step where thinking is done. Note that multiple 'Inner Thinking' steps are required to describe thorough reasoning. Each step should first generate a brief title.
- **'Final Conclusion'**: At this stage, you summarize the correct reasoning from previous 'Inner Thinking' steps and provide the final answer. No title is required here.
- **'Verification'**: At this stage, you verify the conclusion from the "Final Conclusion" step. If the conclusion holds, end the process. If not, return to "Inner Thinking" for further reasoning. No title is required here.

The output format must strictly follow the JSON structure below:
“‘json
{
"CoT": [
{"action": "Inner Thinking", "title": "...", "content": "..."},
...,
{"action": "Final Conclusion", "content": "..."},
{"action": "Verification", "content": "..."}
]
}
“‘

Figure 9: The prompt for initial CoT. {Question} represents the input question, i.e., the question x of the medical verifiable problems.

The Prompt for Backtracking Breask Search Strategy

<question>
{Question}
</question>

<previous reasoning>
{Previous_CoT}
<previous reasoning>

<response requirements>
Your response must include the following steps, each composed of three types of actions: **"Inner Thinking"**, **"Final Conclusion"**, and **"Verification"**:

1. **Inner Thinking**: Break down the reasoning process into multiple concise steps. Each step should start with a brief title to clarify its purpose.
2. **Final Conclusion**: Summarize the correct reasoning from all previous 'Inner Thinking' steps and provide the final answer. No title is needed for this section.
3. **Verification**: Verify the accuracy of the "Final Conclusion". If it holds, conclude the process. Otherwise, return to "Inner Thinking" for further refinement.
</response requirements>

<question> represents the question to be answered, and <previous reasoning> contains your prior reasoning. Your task is to continue from the current 'Verification' step. I have manually reviewed the reasoning and determined that the **Final Conclusion** is false. Your 'Verification' results must align with mine. Proceed to refine the reasoning using **backtracking** to revisit earlier points of reasoning and construct a new Final Conclusion.

Output Format
Strictly follow the JSON structure below. You do not need to repeat your previous reasoning. Begin directly from the next 'Verification' stage.

“‘json
{
"CoT": [
{"action": "Verification", "content": "..."},
{"action": "Inner Thinking", "title": "...", "content": "..."},
...,
{"action": "Final Conclusion", "content": "..."},
{"action": "Verification", "content": "..."}
]
}
“‘

Figure 10: The prompt for **Backtracking** search strategy. Here, {Question} represents the problem x of the medical verifiable problems, and {Previous_CoT} represents the previous chain of thought process, i.e., $[e_0, y_0, \dots, e_{i-1}, y_{i-1}]$.

The Prompt for Exploring New Paths Breask Search Strategy

<question>
{Question}
</question>

<previous reasoning>
{Previous_CoT}
<previous reasoning>

<response requirements>
Your response must include the following steps, each composed of three types of actions: **"Inner Thinking"**,

"Final Conclusion", and **"Verification"**:

1. **Inner Thinking**: Break down the reasoning process into multiple concise steps. Each step should start with a brief title to clarify its purpose.
2. **Final Conclusion**: Summarize the correct reasoning from all previous 'Inner Thinking' steps and provide the final answer. No title is needed for this section.
3. **Verification**: Verify the accuracy of the "Final Conclusion". If it holds, conclude the process. Otherwise, return to "Inner Thinking" for further refinement.

</response requirements>

<question> represents the question to be answered, and <previous reasoning> contains your prior reasoning. Your task is to continue from the current 'Verification' step. I have manually reviewed the reasoning and determined that the **Final Conclusion** is false. Your 'Verification' results must align with mine. Proceed to refine the reasoning by exploring new approaches to solving this problem and construct a new Final Conclusion.

Output Format
Strictly follow the JSON structure below. You do not need to repeat your previous reasoning. Begin directly from the next 'Verification' stage.

```
“‘json
{
"CoT": [
{"action": "Verification", "content": "..."},
{"action": "Inner Thinking", "title": "...", "content": "..."},
...,
{"action": "Final Conclusion", "content": "..."},
{"action": "Verification", "content": "..."}
]
}
“‘
```

Figure 11: The prompt for **Exploring New Paths** search strategy. Here, `{Question}` represents the problem x of the medical verifiable problems, and `{Previous_CoT}` represents the previous chain of thought process, i.e., $[e_0, y_0, \ldots, e_{i-1}, y_{i-1}]$.

The Prompt for **Correction** Breask Search Strategy

<question>
`{Question}`
</question>

<previous reasoning>
`{Previous_CoT}`
<previous reasoning>

<response requirements>
Your response must include the following steps, each composed of three types of actions: **"Inner Thinking"**, **"Final Conclusion"**, and **"Verification"**:

1. **Inner Thinking**: Break down the reasoning process into multiple concise steps. Each step should start with a brief title to clarify its purpose.
2. **Final Conclusion**: Summarize the correct reasoning from all previous 'Inner Thinking' steps and provide the final answer. No title is needed for this section.
3. **Verification**: Verify the accuracy of the "Final Conclusion". If it holds, conclude the process. Otherwise, return to "Inner Thinking" for further refinement.

</response requirements>

<question> represents the question to be answered, and <previous reasoning> contains your prior reasoning. Your task is to continue from the current 'Verification' step. I have manually reviewed the reasoning and determined that the **Final Conclusion** is false. Your 'Verification' results must align with mine. Proceed to refine the reasoning by making precise **corrections** to address prior flaws and construct a new Final Conclusion.

Output Format
Strictly follow the JSON structure below. You do not need to repeat your previous reasoning. Begin directly from the next 'Verification' stage.

```
“‘json
{
"CoT": [
{"action": "Verification", "content": "..."},
{"action": "Inner Thinking", "title": "...", "content": "..."},
...,
{"action": "Final Conclusion", "content": "..."},
{"action": "Verification", "content": "..."}
]
}
“‘
```

Figure 12: The prompt for **Correction** search strategy. Here, `{Question}` represents the problem x of the medical verifiable problems, and `{Previous_CoT}` represents the previous chain of thought process, i.e., $[e_0, y_0, \ldots, e_{i-1}, y_{i-1}]$.

The Prompt for **Verification** Breask Search Strategy

<question>
`{Question}`
</question>

<previous reasoning>
`{Previous_CoT}`
<previous reasoning>

<response requirements>
Your response must include the following steps, each composed of three types of actions: **"Inner Thinking"**, **"Final Conclusion"**, and **"Verification"**:

1. **Inner Thinking**: Break down the reasoning process into multiple concise steps. Each step should start with a brief title to clarify its purpose.
2. **Final Conclusion**: Summarize the correct reasoning from all previous 'Inner Thinking' steps and provide the final answer. No title is needed for this section.
3. **Verification**: Verify the accuracy of the "Final Conclusion". If it holds, conclude the process. Otherwise, return to "Inner Thinking" for further refinement.

</response requirements>

<question> represents the question to be answered, and <previous reasoning> contains your prior reasoning. Your task is to continue from the current 'Verification' step. I have manually reviewed the reasoning and determined that the **Final Conclusion** is false. Your 'Verification' results must align with mine. Proceed to refine the reasoning by conducting a thorough **validation** process to ensure validity and construct a new Final Conclusion.

Output Format
Strictly follow the JSON structure below. You do not need to repeat your previous reasoning. Begin directly from the next 'Verification' stage.

“‘json
{
"CoT": [
{"action": "Verification", "content": "..."},
{"action": "Inner Thinking", "title": "...", "content": "..."},
...,
{"action": "Final Conclusion", "content": "..."},
{"action": "Verification", "content": "..."}
]
}
“‘

Figure 13: The prompt for **Verification** search strategy. Here, {Question} represents the problem x of the medical verifiable problems, and {Previous_CoT} represents the previous chain of thought process, i.e., $[e_0, y_0, \dots, e_{i-1}, y_{i-1}]$.

O Prompts for Constructing SFT Training Data

When a successful trajectory $[e_0, y_0, \dots, e_i, y_i]$ is found, it is reformatted into a coherent, natural language reasoning process $\hat{e}$ (*Complex CoT*) using the prompt shown in Figure 14. This reformatting avoids rigid structures, using smooth transitions (e.g., “hmm,” “also,” “wait”) to streamline reasoning and reduce token usage. The model then generates a formal response $\hat{y}$ for for question x using the conclusion of $\hat{e}$ with the prompt in Figure 14.

The prompt for reformatting a reasoning trajectory to complex CoT

<Thought Process>
{Thought_Process}
</Thought Process>

<Question>
{Question}
</Question>

The <Thought Process> above reflects the model's reasoning based on the <Question>. Your task is to rewrite the <Thought Process> to resemble a more human-like, intuitive natural thinking process. The new version should:

1. Be presented as step-by-step reasoning, with each thought on a new line separated by a line break.
2. Avoid structured titles or formatting, focusing on natural transitions. Use casual and natural language for transitions or validations, such as "hmm," "oh," "also," or "wait."
3. Expand the content, making the reasoning richer, more detailed, and logically clear while still being conversational and intuitive.

Return directly the revised natural thinking in JSON format as follows:
“‘json
{
"NaturalReasoning": "..."
}

Figure 14: The prompt for reformatting a reasoning trajectory to complex CoT $\hat{e}$. Here, {Thought_Process} represents the successful reasoning trajectory of $[e_0, y_0, \dots, e_i, y_i]$, and {Question} represents the question x.

The prompt for generating a formal response with complex CoT

<Internal Thinking>
{Complex_CoT}
</Internal Thinking>

<Question>
{Question}
</Question>

The <Internal Thinking> represents your internal thoughts about the <Question>. Based on this, generate a rich and high-quality final response to the user. If there is a clear answer, provide it first. Ensure your final response closely follows the <Question>. The response style should resemble GPT-4's style as much as possible. Output only your final response, without any additional content.

Figure 15: The prompt for generating a formal response $\hat{y}$ with complex CoT $\hat{e}$. Here, {Complex_CoT} represents the complex CoT $\hat{e}$, and {Question} represents the question x.