Unlocking the Mysteries of OpenAI o1: A Survey of the Reasoning Abilities of Large Language Models

Anonymous ACL submission

Abstract

The release of OpenAI's o1 marks a significant milestone in AI, achieving proficiency comparable to PhD-level expertise in mathematics and coding. While o1 excels at solving complex reasoning tasks, it remains a closed-resource model, limiting its accessibility and broader application in academic and industrial contexts. Despite numerous efforts to replicate o1's results, these attempts often focus on isolated aspects of the model (e.g., training, inference), neglecting the holistic interplay between components and failing to provide a global picture of the pathways to enhance LLMs' reasoning capabilities, and replicate o1's performance. Currently, there is no systematic review of these replication efforts, nor a clear survey of the major issues that must be addressed to achieve comparable performance to o1.

In this survey, we provide a systematic review of the most up-to-date state of knowledge on reasoning LLMs, helping researchers understand the current challenges and advancements in this field. Specifically, we will (1) review the basic concepts and techniques behind two representative reasoning LLMs, o1 and DeepSeek R1, exploring their key components and capabilities; (2) detail recent efforts to replicate o1's performances, and more importantly, address the key obstacles in enhancing the reasoning abilities; (3) explore the emerging class of LLMs designed for multi-modal reasoning, which extends the capabilities of traditional LLMs by bridging the gap between language understanding and sensory perception; and (4) summarize the current challenges and discuss opportunities for further improvement of reasoning large language models.

1 Introduction

Large language models (LLMs) (Jiang et al., 2023; Bai et al., 2023; OpenAI, 2023; Yang et al., 2024a; Dubey et al., 2024; OpenAI, 2024a; Mistral AI, 2024; Team et al., 2024; Liu et al., 2024b,a; Wake et al., 2024; Shao et al., 2024a; OpenAI, 2024b; GLM et al., 2024) have achieved remarkable performance in numerous language tasks (Sun et al., 2023b; Wang et al., 2023b; Wan et al., 2023; Sun et al., 2023c,a; Wang et al., 2023a; Sun et al., 2023d; Liu et al., 2024c; Yao et al., 2024; Wang et al., 2024d). Despite their impressive capabilities, LLMs still face significant challenges in reasoning. They struggle with tasks that require logical deduction, numerical calculations, or consistent chains of thought. Errors are observed even in simple tasks that demand multi-step thinking, highlighting gaps in how these models acquire, represent, and apply knowledge (Cobbe et al., 2021; Wei et al., 2022; Wang and Lu, 2023; Shakarian et al., 2023; Shi et al., 2023; Chang et al., 2024; Ahn et al., 2024).

The release of OpenAI o1 (OpenAI, 2024b) marks a significant milestone in AI, particularly in enhancing its reasoning abilities. OpenAI o1 is capable of solving complex reasoning tasks and demonstrates capabilities comparable to PhD-level proficiency in math and coding. Unfortunately, o1 is a closed-resource model, which limits its accessibility and potential for broader academic and industrial use. This restricted access hinders collaborative efforts to further refine its abilities and limits the opportunity for researchers and developers to build upon its foundation. Additionally, the lack of transparency in the model's underlying architecture and training data raises concerns about bias and fairness, making it difficult to fully understand its decision-making processes.

As a result of the closed-resource nature of o1, numerous efforts have emerged to replicate o1's impressive results (Shao et al., 2024a; Mistral AI, 2024; Team, 2024b; o1 Team, 2024; Zhao et al., 2024; Team, 2024a; DeepSeek-AI et al., 2024). o1, however, is a highly complex system, with substantial improvements across multiple AI modules, including training methodologies, inference mechanisms, datasets, and evaluation processes. Existing efforts to replicate o1 tend to focus on isolated aspects of the model, often neglecting the holistic interplay between these components, and, as a result, missing the full picture in enhancing LLMs' reasoning abilities. As a result, there is currently no systematic review of the efforts to replicate o1, and more importantly, no clear survey of the major issues that must be addressed to achieve comparable performance across all these dimensions.

In this survey, we provide a systematic review of the most up-to-date state of knowledge on the challenges and opportunities involved in reasoning LLMs, in particularly with OpenAI o1 (OpenAI, 2024b) and DeepSeek R1 (DeepSeek-AI et al., 2025). Specifically, Sec. 2 reviews the basic concepts and techniques behind two representative reasoning LLMs, o1 and DeepSeek R1, exploring their key components and capabilities; Sec. 3 details recent efforts to replicate o1's performances, and more importantly, addresses the key obstacles in enhancing the reasoning abilities; Sec. 4 explores the emerging class of LLMs designed for multimodal reasoning, which extends the capabilities of traditional LLMs by bridging the gap between language understanding and sensory perception; Sec. 5 summarizes the current challenges and discusses opportunities for further improvement of reasoning LLMs; and Sec. 6 concludes this survey.

2 Overview of OpenAI o1 and DeepSeek R1

OpenAI o1 and DeepSeek R1 represent two significant advancements in artificial intelligence, each emphasizing enhanced reasoning capabilities to tackle complex, multi-step tasks. In this section, we first illustrate one of their key features: "Thinking before Response Generation", which allows them to produce detailed, step-by-step solutions that excel in domains such as science, mathematics, and programming. Then, we explore the speculated techniques in o1 and insights from R1 to enhance the reasoning capabilities of LLMs across three dimensions: (1) At the **dataset** level, we review recent efforts in constructing high-quality, diverse, and representative datasets that play a pivotal role in enhancing LLMs' reasoning capabilities; (2) At the **training** level, we delve into different training techniques that harness these datasets to ensure the model acquires robust logical reasoning and structured problem-solving capabilities; and (3) At the **inference** level, we provide an in-depth review of techniques designed to identify and execute reasoning paths during the decoding process in LLMs. Additionally, since both the o1 and R1 series consist of multiple models, each tailored with varying sizes, inference speeds, and prices to meet different use cases, we provide a detailed breakdown of the release dates, capabilities, features of the various o1 and R1 models, along with their performance on benchmarks, in the Appendix A.

2.1 Key Feature: Thinking before Response Generation

A hallmark of both the o1 and R1 models is the deliberate allocation of additional time for "thinking" before generating responses. This extended processing period allows the models to produce detailed, step-by-step reasoning paths that are particularly effective in solving complex mathematical, programming, and scientific challenges, such as:

The Given Question:

How many Rs are in "strawberry"?

Generated by o1:

There are three 'R's in the word "strawberry."

Here's the breakdown: *S - T - R - A - W - B - E - R - R - Y*

- 3rd letter: R
- 8th letter: R
- 9th letter: R

So, the letter 'R' appears three times in "strawberry."

By allocating more "thinking" time, the model can decompose complex, multi-step problems into smaller, manageable components. On one hand, this allows the model to produce well-structured and coherent solutions to reasoning problems; on the other, it offers transparency into the reasoning process, enabling users to understand how the model reached its answers.

2.2 Speculated Techniques in o1 and Insights from R1

OpenAI o1 is a closed-source model, and as a result, there has been much speculation regarding the techniques behind its development. Various blogs, reports, and technical papers have attempted to deconstruct its capabilities, offering hypotheses about its underlying architecture, datasets, training methods, and reasoning processes. Some speculate that o1 leverages advanced forms of reinforcement learning and unsupervised learning, while others suggest a highly refined chain-of-thought (CoT) approach to enhance its multi-step reasoning. In parallel, the DeepSeek R1 series, detailed in the DeepSeek-R1 technical report (DeepSeek-AI et al., 2025), employs a transparent, multi-stage training pipeline that shares several conceptual similarities with the speculated techniques in o1. Below, we explore the speculated techniques and insights from R1 to enhance the reasoning capabilities of LLMs across three levels: dataset, training and inference.

2.2.1 Constructing Reasoning Formatted Datasets

Building the training dataset is the initial step toward equipping a model with advanced reasoning capabilities. For reasoning LLMs, this involves constructing a reasoning formatted dataset that emphasizes logical progression, multi-step thinking, and structured problem-solving. For example:

> *Input*: What is the sum of the first 10 positive integers?
>
> *Reasoning Path:*
>
> - Step 1: The first 10 positive integers are $1, 2, 3, \cdots, 10$.
> - Step 2: The sum of a sequence can be calculated using the formula $(n \times (n + 1))/2(n \times (n + 1))/2(n \times (n + 1))/2$.
> - Step 3: Substituting $n = 10$, we get $(10 \times 11)/2 = 55(10 \times 11)/2 = 55(10 \times 11)/2 = 55$.
> - Answer: The sum is 55.

In prior research (Swamy et al., 2024), OpenAI researchers discovered that exposing LLMs to reasoning formatted data enables them to learn logical patterns and enhance their accuracy. To construct such reasoning formatted datasets, most studies (Qin et al., 2024; Hwang et al., 2024a; Liao et al., 2024b; Lu et al., 2024; Shao et al., 2024b; Bansal et al., 2024; Tang et al., 2024) typically focus on one or a combination of the following strategies: the **machine-generated**, where a trained model provides feedback, responses or grades, and the **human-generated**, where humans are asked to provide feedback. Some datasets rely on a single strategy, while others combine both approaches, an overview of these datasets is shown in Table 2.

Recently, in DeepSeek R1, thousands of cold-start examples are collected with a clearly defined output format. Each sample is designed with special tokens that separate the detailed chain-of-thought from a concise summary, ensuring that the dataset not only promotes logical reasoning but also enhances readability. This method of dataset construction mirrors the idea that exposing models to structured reasoning data enables them to learn logical patterns and improve overall accuracy.

2.2.2 Training LLMs on Reasoning Formatted Datasets

After constructing the specialized reasoning formatted datasets, it has been speculated that the next step for models like o1 is to fine-tune them on this data using a combination of supervised fine-tuning (SFT) (Zhang et al., 2023) and reinforcement learning (RL) (Wang et al., 2024e). SFT is employed as an initial step to train the model to generate complete reasoning paths in response to given questions. This process teaches the model to follow logical chains and produce coherent outputs, thereby establishing a solid foundation for tackling more advanced reasoning challenges (Gou et al., 2023; Tian et al., 2024; Liao et al., 2024a). RL, on the other hand, is used to further refine these capabilities (Saunders et al., 2022; Yu et al., 2023; Liu et al., 2023; Wang et al., 2024b; Zhang et al., 2025; Zeng et al., 2024; Zheng et al., 2024; Song et al., 2025; Zhang et al., 2024c; Chen et al., 2024c; Liu et al., 2023; Hwang et al., 2024b; Putta et al., 2024). Techniques such as large-scale Reinforcement Learning from Human Feedback (RLHF) and alternative strategies like Monte Carlo Tree Search (MCTS) are speculated to drive the exploration of multiple solution paths and to optimize the model's performance by using reward models that capture correctness and logical consistency. We put technical details on these approaches in Appendix B, and a summary of each of these paradigms in Table 3.

In contrast, DeepSeek R1 adopts a two-phase training strategy to develop its reasoning abilities: (1) **R1-Zero** is developed through pure RL without any SFT. This phase allows the model to self-evolve its reasoning strategies and explore various solution paths, although it may sometimes produce outputs with challenges in readability and language consistency. (2) **R1** incorporates a modest amount of high-quality cold-start data to guide and stabilize the reasoning process. This phase is further refined through additional RL training stages and SFT (including rejection sampling for high-quality responses), which helps the model generate coherent and user-friendly chain-of-thought outputs.

This dual approach ensures that the model learns to generate complete and logical reasoning paths in response to given questions, establishing a robust foundation for addressing more complex problems.

2.2.3 Inference with Advanced Thinking Strategies

At inference time, researchers speculate that o1-like models employ a series of advanced reasoning techniques, many of which are also utilized in R1. Below, we present examples to demonstrate these techniques, while additional technical details are provided in Appendix C, and an overview of recent inference methods is summarized in Table 4.

Problem Breakdown. Deconstructing complex problems into smaller, manageable parts (as illustrated below) to facilitate a step-by-step solution, an approach also central to the design of R1.

The Given Question:

How many Rs are in "strawberry"?

Decomposition:

- Identify all the words: *S* - *T* - *R* - *A* - *W* - *B* - *E* - *R* - *R* - *Y*
- Identify which positions are the word R: 3rd letter, 8th letter, and 9th letter.
- Calculate: The letter 'R' appears three times

Mistake Recognition and Self-Correction. In this strategy, the model detects and rectifies errors in their reasoning, much like a human reassessing and adjusting a flawed approach. Such as when get a response "The area is $10 \times 5 = 50$" by given a question "What is the area of a triangle with a base of 10 and a height of 5?". A critic model is employed (McAleese et al., 2024; Xi et al., 2024; Kalyanpur et al., 2024), or the o1-like reasoning LLM itself is prompted (Kumar et al.; Gao et al., 2024; Li et al., 2024), to evaluate whether the response is accurate. If an error is identified, the o1-like reasoning LLM will immediately generate a new answer. This iterative process continues until the critic model or the LLM determines that the generated response is correct. R1 demonstrates this capability through its iterative RL process, where an "aha moment", which is shown in Figure 4, occurs as the model dynamically adjusts its reasoning.

Solution Exploration. o1-like reasoning LLMs explore multiple potential solution paths before arriving at a final answer, ensuring the selection of the most logical and accurate outcome. This process resembles a tree structure, where the input problem serves as the root node, each node represents a step in the solution, and the path from a leaf node to the root forms a complete reasoning trajectory(Yao et al., 2024; Yuan et al., 2024; Feng et al., 2023; Zhang et al., 2024b; Tian et al., 2024; Xie et al., 2024; Chen et al., 2024b; Zhang et al., 2024a). To enhance accuracy, o1-like models employ various search strategies to construct this solution tree and evaluate the validity of each path, leading to more precise and insightful results. However, R1 does not adopt this approach due to the challenges of scaling token-level search, such as an exponentially expanding search space and the difficulty of training a reliable value model. Instead, R1 leverages extended test-time computation and reinforcement learning to refine its reasoning process without relying on explicit tree search algorithms.

3 Recent Efforts in Reproducing OpenAI o1

Currently, many efforts have been made to replicate OpenAI's o1 or specific capabilities of o1 (such as code generation and mathematical reasoning). We have collected nine such works, among which seven are open-source, five provide reports or papers, and five include comparisons with o1. For detailed information, please refer to Table 5. In the following section, we will focus on introducing three landmark works: (1) rStar-Math (Guan et al., 2025), which shows that smaller language models can match or even exceed the mathematical reasoning capabilities of OpenAI's o1 model, without requiring distillation from larger models; (2) Kimi-k1.5 (Team et al., 2025), a multi-modal LLM that represents a major advancement in scaling reinforcement learning; and (3) DeepSeek-R1 (DeepSeek-AI et al., 2025), a state-of-the-art reasoning model that achieves performance comparable to OpenAI's o1 series models. For additional similar works in replicating o1's reasoning abilities, please refer to Appendix D.

3.1 rStar-Math

rStar-Math (Guan et al., 2025), developed by Microsoft, demonstrates that small language models (SLMs) can match or surpass the mathematical reasoning abilities of OpenAI's o1 model, without the need for distillation from larger models. This is achieved through "deep thinking" via MCTS, where an SLM-based math policy conducts test-time searches, guided by a process preference model (PPM) also built on SLMs. The core advancements of rStar-Math lie in three key solutions designed to overcome training challenges for the two SLMs: (1) a code-enhanced CoT data synthesis method, (2) a PPM training framework, and (3) a self-evolution strategy. Extensive experiments demonstrate significant improvements on the MATH (Hendrycks et al., 2021) benchmark. rStar-Math enhances the accuracy of Qwen2.5-Math-7B (Yang et al., 2024a) from 58.8% to 90.0% and Phi3-mini-3.8B (Abdin et al., 2024) from 41.4% to 86.4%, outperforming o1-preview by +4.5% and +0.9%, respectively. On the American Invitational Mathematics Examination (AIME) (AI-MO, 2025), rStar-Math successfully solves an average of 53.3% (8/15) of the problems, placing it within the top 20% of high school math students.

rStar-Math trains a math policy SLM and a PPM integrated with MCTS for deep thinking. The training involves three key innovations. **First**, a code-augmented CoT data synthesis method uses MCTS rollouts to generate step-by-step reasoning trajectories annotated with self-assigned Q-values. The policy SLM samples candidate nodes, producing one-step CoTs and corresponding Python code. Only nodes with successful code execution are retained, reducing errors. Q-values are assigned to each step based on its contribution, ensuring accurate reasoning trajectories. **Second**, a PPM to enable reliable prediction of reward labels for math reasoning steps. Rather than using noisy Q-values directly, the PPM distinguishes correct steps from incorrect ones using preference pairs and optimizes its scoring with a pairwise ranking loss (Ouyang et al., 2022). This improves the accuracy of stepwise reward assignment compared to traditional methods (Chen et al., 2024b). **Finally**, a four-round self-evolution framework refines the policy model and PPM. Starting with a dataset of 747k math word problems, each round uses the updated models to generate better training data. This iterative process leads to: (1) a stronger policy SLM, (2) a more reliable PPM, (3) improved reasoning trajectories, and (4) expanded data coverage for more challenging math problems.

3.2 Kimi-k1.5

Kimi-k1.5 (Team et al., 2025), developed by Moonshot AI, is a multi-modal LLM which represents a significant advancement in scaling RL. The authors introduce a novel approach by focusing on long context scaling, extending the context window of RL to 128k, and refining policy optimization methods. Unlike traditional RL frameworks that rely on complex techniques such as MCTS, value functions, and process reward models, Kimi-k1.5 establishes a streamlined and effective RL framework. The model achieves state-of-the-art reasoning performance across various benchmarks and modalities, rivaling OpenAI's o1. Additionally, the authors introduce long2short methods that utilize long-CoT techniques to enhance short-CoT models, significantly outperforming existing models like GPT-4o (OpenAI, 2024a) and Claude Sonnet 3.5[1] by up to 550%.

The development of Kimi-k1.5 involves several stages: pretraining, vanilla SFT, long-CoT SFT, and RL. The primary innovation lies in the RL phase, where the authors construct a high-quality RL prompt set designed to guide the model toward robust reasoning while mitigating risks such as reward hacking and overfitting to superficial patterns. This prompt set is characterized by three key properties: diverse coverage, balanced difficulty, and accurate evaluability. During RL training, three critical strategies are employed: (1) **Online Policy Mirror Descent**: A variant of this algorithm is used to optimize the training process (Abbasi-Yadkori et al., 2019; Mei et al., 2019; Tomar et al., 2020); (2) **Length Penalty**: A reward mechanism is introduced to control the rapid growth of token length, enhancing token efficiency; and (3) **Sampling Methods**: Two sampling techniques are utilized to improve training efficiency: *a. Curriculum sampling* progressively trains the model from simpler to more complex tasks, enhancing both training efficiency and model performance. *b. Prioritized sampling* focuses on areas where the model underperforms by sampling problematic tasks more frequently, proportional to their failure rates, accelerating learning in weaker areas.

[1] `https://www.anthropic.com/news/claude-3-5-sonnet`

While long-CoT models demonstrate strong performance, they often require more test-time tokens compared to standard short-CoT LLMs. To address this, the authors propose four methods to transfer the reasoning capabilities of long-CoT models to short-CoT models, a challenge referred to as the long2short problem. These methods include: (1) **Model Merging**: Combining a long-CoT model with a shorter model by averaging their weights. (2) **Shortest Rejection Sampling**: Using the long-CoT model to generate multiple responses to the same question and selecting the shortest correct response for SFT. (3) **Direct Preference Optimization (DPO)**: Forming pairwise preference data using positive (shortest correct solution) and negative (longer solutions) samples for DPO training. (4) **Long2short RL**: A two-phase training approach where, after standard RL training, a model with optimal performance and token efficiency is selected for a second phase. In this phase, a length penalty is applied, and the maximum response length is reduced to encourage more concise responses.

3.3 DeepSeek-R1

DeepSeek-R1 (DeepSeek-AI et al., 2025), developed by DeepSeek, is a state-of-the-art reasoning model that achieves performance comparable to OpenAI's o1 series models. This work pioneers the use of pure RL to enhance language model reasoning capabilities, focusing on self-evolution without relying on supervised data. The authors first train DeepSeek-R1-Zero, a model derived from DeepSeek-V3-Base (Liu et al., 2024a), using large-scale RL without SFT. This preliminary model demonstrates significant reasoning improvements, with the pass@1 score on AIME 2024 (AI-MO, 2025) increasing from 15.6% to 71.0%. With majority voting, the score further rises to 86.7%, matching the performance of OpenAI-o1-0912. To address issues such as poor readability, language mixing, and to further boost reasoning performance, the authors introduce DeepSeek-R1. This enhanced model incorporates a small amount of cold-start data and a multi-stage training pipeline, achieving performance on par with OpenAI-o1-1217, which is shown in Figure 3.

The RL training process for DeepSeek-R1-Zero employs Group Relative Policy Optimization (GRPO) (Shao et al., 2024a), which eliminates the need for a critic model by estimating baselines from group scores. The reward system is rule-based, consisting of two main components: accuracy rewards and format rewards. The accuracy reward evaluates the correctness of responses, while the format reward enforces the use of '<think>' and '</think>' tags to structure the reasoning process. During training, an intermediate version of the model exhibited an "aha moment", which is shown in Figure 4, where it learned to allocate more time to reevaluate its initial approach, demonstrating the evolving reasoning capabilities facilitated by RL.

The training process for DeepSeek-R1 consists of two alternating stages of SFT and RL: (1) **Initial Cold Start SFT**: The process begins with the collection of thousands of high-quality, readability-focused long CoT datasets. These datasets are used to fine-tune DeepSeek-V3-Base, establishing a robust foundation for subsequent RL training; (2) **First Reasoning-oriented RL Stage**: The model undergoes large-scale reasoning-oriented RL, leveraging the same methodology applied in DeepSeek-R1-Zero to enhance its reasoning abilities. Upon convergence, the resulting checkpoint is utilized to gather additional SFT data for the next stage; (3) **Second SFT Stage**: This phase focuses on further refining the model through a combination of reasoning and non-reasoning data. For reasoning data, specialized prompts are curated, and reasoning trajectories are generated via rejection sampling using the RL checkpoint from the previous stage. For non-reasoning data, such as writing, factual QA, self-cognition, and translation, the DeepSeek-V3 pipeline is utilized, integrating portions of the DeepSeek-V3 SFT dataset. Finally, DeepSeek-V3-Base is fine-tuned for two epochs using this comprehensive dataset to ensure optimal performance across a wide range of tasks; and (4) **Second RL Stage for all Scenarios**: A final RL phase is conducted to align the model with human preferences by fine-tuning the model with a combination of reward signals and diverse prompt distributions, enhancing its helpfulness and harmlessness while further refining its reasoning abilities.

4 Beyond Language: Multi-modal Reasoning LLMs

While the current key feature of the o1 models, "Thinking before Response Generation,"does not yet support multi-modal functions, future versions hold significant promise for integrating them. Multi-modal reasoning LLMs (MLLMs) are a class of models that enhance the capabilities of traditional LLMs by bridging the gap between language understanding and sensory perception. These models enable more sophisticated reasoning across various types of data. By incorporating the ability to process and integrate information from different data modalities, such as text, images, and audio, MLLMs have the potential to greatly improve holistic reasoning, offering users a richer and more comprehensive experience. Currently, MLLMs are an emerging category, with many studies focusing on enhancing their multi-step visual reasoning capabilities (Dong et al., 2024; Hu et al., 2024; Team, 2024a), fewer studies addressing their spatial understanding and reasoning abilities (Carbune et al., 2024; Chen et al., 2024a), and ongoing efforts to improve LLMs' reasoning with table-based data (Wang et al., 2024f). An overview of these MLLMs is provided in Figure 6, and more details about these works can be found in Appendix E.

5 Evaluation & Analysis & Future

This section explores the reasoning capabilities of LLMs by reviewing recent research that approaches the topic from various angles. It considers aspects such as token bias, the length of reasoning steps, and the reliability of CoT explanations, offering a thorough assessment of how reasoning influences model alignment, safety, and generalization. In addition, we provide an overview of the current evaluation benchmarks in use and highlight the need for future benchmarks specifically tailored to evaluate reasoning skills.

5.1 Evaluation

The o1-like model demonstrates strong reasoning capabilities across a variety of benchmarks. Currently, evaluations tend to focus on (1) scientific domains, such as mathematics, physics, chemistry, and biology, with datasets including GPQA (Rein et al., 2023), OlympiadBench (He et al., 2024), Minerva (Lewkowycz et al., 2022), GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021) and American Invitational Mathematics Examination (AIME) (AI-MO, 2025); and (2) programming contests (coding), such as Codeforces (Mirzayanov, 2025). These tasks primarily involve high-difficulty, competitive-level problems, including some PhD-level questions in science and engineering. For further details about those datasets, please refer to Appendix F. In the future, in addition to the accuracy and difficulty of evaluation benchmarks, there is significant opportunity for research in developing new benchmarks specifically aimed at assessing LLMs' reasoning abilities. These could include factors such as evaluating the correctness and length of the generated reasoning steps, as well as the relevance and contribution of each reasoning step to the final answer.

5.2 Safety & Policy

Guan et al. (2024) introduce deliberative alignment, a novel training paradigm that leverages LLMs' reasoning capabilities to improve their safety. This approach trains models to explicitly recall and reason through safety specifications before generating responses. When applied to OpenAI's o-series models, deliberative alignment enables the use of CoT reasoning to analyze user prompts, reference relevant policy guidelines, and produce safer outputs. Experimental results demonstrate that o-series models aligned through deliberative alignment achieve precise compliance with OpenAI's safety policies without relying on human-authored chain-of-thoughts or answers. Additionally, deliberative alignment advances the Pareto frontier by strengthening resistance to jailbreak attempts, lowering overrefusal rates, and enhancing generalization to out-of-distribution contexts. These outcomes underscore that reasoning over clearly defined policies fosters more scalable, reliable, and transparent model alignment.

5.3 Faithfulness of LLMs' Reasoning Process

Faithfulness refers to how accurately the model's reasoning process (e.g., CoT) aligns with the actual internal computation or representations used to derive the output. Recent research indicates that this alignment is frequently lacking, making the enhancement of faithfulness a critical challenge for achieving greater transparency, alignment, and reliability in large language models.

5.3.1 Faithfulness of CoT Reasoning

Lanham et al. (2023) examine whether the reasoning presented in CoT explanations accurately reflects the actual reasoning processes of LLMs. **First**, they evaluate post-hoc reasoning, where reasoning is generated after the conclusion has already been determined, by truncating or introducing errors into the CoT before the final answer. Their findings reveal significant variation in LLMs' reliance on CoT across tasks: some tasks exhibit no dependence on CoT, while others rely on it heavily. Interestingly, post-hoc reasoning tends to worsen with more capable models, indicating that smaller models may be more reliable for tasks requiring faithful reasoning. **Second**, they investigate whether CoT's performance gains stem from increased test-time computation. By replacing CoT with uninformative filler text, they find no accuracy improvements, suggesting that test-time computation alone does not account for CoT's effectiveness. **Third**, they explore whether CoT encodes task-relevant information in ways inaccessible to human interpretation. By substituting CoT with paraphrased versions, they observe no performance degradation, indicating that the specific phrasing of CoT is not crucial to its success. In conclusion, these findings emphasize the major challenges in CoT faithfulness and underscore the importance of developing systems with more transparent and reliable reasoning processes.

5.3.2 Token Bias in LLMs' Reasoning

Jiang et al. (2024) introduce the concept of token bias: an LLM exhibits token bias in a reasoning task if changes to some or all tokens in the task description, while maintaining the underlying logic, predictably alter the model's output. To determine whether LLMs are capable of genuine reasoning or if their performance is primarily driven by token biases, the authors propose a hypothesis-testing framework. This framework outlines a set of hypotheses where token biases are readily identifiable, with all null hypotheses assuming the genuine reasoning capabilities of LLMs. By doing so, they show that while LLMs may perform well on classic problems, their success is largely driven by recognizing superficial patterns influenced by strong token bias. This raises concerns about their true reasoning and generalization capabilities. These findings suggest that CoT prompting and in-context learning may not invoke genuine reasoning in LLMs. Instead, they may lead to semantic shortcuts that superficially mimic the desired behavior. This highlights the need for further investigation into the underlying mechanisms and limitations of LLMs, particularly with respect to their reasoning abilities.

5.4 Controlling Reasoning Length in LLMs

Recent studies have highlighted the need for controlling reasoning length in LLMs, as issues such as overthinking, and redundant computations contribute to inefficient resource allocation and increased costs (Nori et al., 2024; Sprague et al., 2024; Yang et al., 2024b; Jin et al., 2024).

Specifically, to address the overthinking issue in reasoning-focused LLMs, where excessive computational resources are allocated to simple tasks without proportional benefits, Chen et al. (2024d) suggest strategies like length preference optimization and response simplification to streamline the reasoning process. In contrast, Han et al. (2024) introduce TALE, a token-budget-aware reasoning framework that dynamically adjusts token budgets based on the complexity of the task. These approaches successfully optimize the balance between computational efficiency and performance, reducing unnecessary reasoning steps while preserving effectiveness, and offer promising solutions for intelligent resource scaling in reasoning tasks. Regarding redundant inference costs, Jang et al. (2024) propose VARR, a framework for reducing sentence-level rationale based on a principled verbosity criterion. VARR employs a likelihood-based method to identify and eliminate redundant reasoning steps during training, keeping only the most crucial steps. This approach ensures the integrity of reasoning while minimizing the risk of generating incorrect answers, effectively balancing both efficiency and accuracy.

6 Conclusion

In this survey, we presented a review of reasoning LLMs by focusing on dataset construction, supervised fine-tuning, reinforcement learning, and advanced inference strategies (chain-of-thought and automated critiques) through the lens of OpenAI's o1 model and DeepSeek's R1 model. Despite the progress, several challenges exist. Formal verification and robust error detection are necessary to improve the interpretability and trustworthiness of reasoning trace. Reliance on purely text-based logic necessitates neuro-symbolic frameworks that combine continuous embeddings with external symbolic manipulators for advanced mathematics, proofs, or legal argumentation. Beyond the targeted fine-tuning in math or coding, real-world applications demand broader domain adaptation and multi-modal reasoning, integrating signals from text, vision, audio, and beyond. The transition of LLMs from mere next-token predictors to structured reasoners is under way, and while o1 showcases the promise of today's solutions, forging robust, trustworthy, and multi-modal reasoning engines necessitates substantial future works.

7 Limitations

In this survey, we focus on providing a systematic review of reasoning LLMs, but several technical aspects, such as the optimization techniques and iteration processes of the widely used reinforcement learning algorithm Group Relative Policy Optimization (GRPO), have not been discussed in depth. Additionally, as reasoning LLMs are an emerging class, numerous communities, research groups, and companies are working to replicate the performance of o1 and R1 in order to develop their own powerful reasoning LLMs. Some of the existing works in this area, such as training frameworks, evaluation benchmarks, and considerations regarding the safety and faithfulness of the reasoning process, are still in progress and not yet complete. We will continue to track developments in this field and update the latest advancements related to reasoning LLMs as they emerge.

References

Yasin Abbasi-Yadkori, Peter Bartlett, Kush Bhatia, Nevena Lazic, Csaba Szepesvari, and Gellért Weisz. 2019. Politex: Regret bounds for policy iteration using expert prediction. In *International Conference on Machine Learning*, pages 3692–3702. PMLR.

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, and 1 others. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Janice Ahn, Rishu Verma, Renze Lou, Di Liu, Rui Zhang, and Wenpeng Yin. 2024. Large language models for mathematical reasoning: Progresses and challenges. *arXiv preprint arXiv:2402.00157*.

AI-MO. 2025. Aimo validation dataset. Accessed: 2025-01-12.

Afra Amini, Tim Vieira, and Ryan Cotterell. 2024. Direct preference optimization with an offset. *arXiv preprint arXiv:2402.10571*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Hritik Bansal, Arian Hosseini, Rishabh Agarwal, Vinh Q Tran, and Mehran Kazemi. 2024. Smaller, weaker, yet better: Training llm reasoners via compute-optimal sampling. *arXiv preprint arXiv:2408.16737*.

Andrew G Barto, Richard S Sutton, and Charles W Anderson. 1983. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE transactions on systems, man, and cybernetics*, (5):834–846.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V Le, Christopher Ré, and Azalia Mirhoseini. 2024. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*.

Victor Carbune, Hassan Mansoor, Fangyu Liu, Rahul Aralikatte, Gilles Baechler, Jindong Chen, and Abhanshu Sharma. 2024. Chart-based reasoning: Transferring capabilities from llms to vlms. *arXiv preprint arXiv:2403.12596*.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, and 1 others. 2024. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. 2024a. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14455–14465.

Guoxin Chen, Minpeng Liao, Chengxi Li, and Kai Fan. 2024b. Alphamath almost zero: process supervision without process. *arXiv preprint arXiv:2405.03553*.

Guoxin Chen, Minpeng Liao, Chengxi Li, and Kai Fan. 2024c. Step-level value preference optimization for mathematical reasoning. *arXiv preprint arXiv:2406.10858*.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W Cohen. 2022. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *arXiv preprint arXiv:2211.12588*.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2020. Tabfact : A large-scale dataset for table-based fact verification. In *International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia.

Xi Chen, Xiao Wang, Lucas Beyer, Alexander Kolesnikov, Jialin Wu, Paul Voigtlaender, Basil Mustafa, Sebastian Goodman, Ibrahim Alabdulmohsin, Piotr Padlewski, and 1 others. 2023. Pali-3 vision language models: Smaller, faster, stronger. *arXiv preprint arXiv:2310.09199*.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, and 1 others. 2024d. Do not think that much for 2+ 3=? on the overthinking of o1-like llms. *arXiv preprint arXiv:2412.21187*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 181 others. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

DeepSeek-R1-Lite-Preview. 2024. Deepseek-r1-lite-preview.

Yuhao Dong, Zuyan Liu, Hai-Long Sun, Jingkang Yang, Winston Hu, Yongming Rao, and Ziwei Liu. 2024. Insight-v: Exploring long-chain visual reasoning with multimodal large language models. *arXiv preprint arXiv:2411.14432*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Xidong Feng, Ziyu Wan, Muning Wen, Stephen Marcus McAleer, Ying Wen, Weinan Zhang, and Jun Wang. 2023. Alphazero-like tree-search can guide large language model decoding and training. *arXiv preprint arXiv:2309.17179*.

Jiahui Gao, Renjie Pi, Jipeng Zhang, Jiacheng Ye, Wanjun Zhong, Yufei Wang, Lanqing Hong, Jianhua Han, Hang Xu, Zhenguo Li, and 1 others. 2023. G-llava: Solving geometric problem with multi-modal large language model. *arXiv preprint arXiv:2312.11370*.

Kuofeng Gao, Huanqia Cai, Qingyao Shuai, Dihong Gong, and Zhifeng Li. 2024. Embedding self-correction as an inherent ability in large language models for enhanced mathematical reasoning. *arXiv preprint arXiv:2410.10735*.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, and 1 others. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*.

Google AI. Thinking Mode - Gemini API Documentation. https://ai.google.dev/gemini-api/docs/thinking-mode. Accessed: 2025-01-16.

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

Melody Y Guan, Manas Joglekar, Eric Wallace, Saachi Jain, Boaz Barak, Alec Heylar, Rachel Dias, Andrea Vallone, Hongyu Ren, Jason Wei, and 1 others. 2024. Deliberative alignment: Reasoning enables safer language models. *arXiv preprint arXiv:2412.16339*.

Xinyu Guan, Li Lyna Zhang, Yifei Liu, Ning Shang, Youran Sun, Yi Zhu, Fan Yang, and Mao Yang. 2025. rstar-math: Small llms can master math reasoning with self-evolved deep thinking. *arXiv preprint arXiv:2501.04519*.

Tingxu Han, Chunrong Fang, Shiyu Zhao, Shiqing Ma, Zhenyu Chen, and Zhenting Wang. 2024. Token-budget-aware llm reasoning. *arXiv preprint arXiv:2412.18547*.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, and 1 others. 2024. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint arXiv:2402.14008*.

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*.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander 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.

Yushi Hu, Weijia Shi, Xingyu Fu, Dan Roth, Mari Ostendorf, Luke Zettlemoyer, Noah A Smith, and Ranjay Krishna. 2024. Visual sketchpad: Sketching as a visual chain of thought for multimodal language models. *arXiv preprint arXiv:2406.09403*.

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

Hyeonbin Hwang, Doyoung Kim, Seungone Kim, Seonghyeon Ye, and Minjoon Seo. 2024a. Self-explore: Enhancing mathematical reasoning in language models with fine-grained rewards. *arXiv preprint arXiv:2404.10346*.

Hyeonbin Hwang, Doyoung Kim, Seungone Kim, Seonghyeon Ye, and Minjoon Seo. 2024b. Self-explore: Enhancing mathematical reasoning in language models with fine-grained rewards. *arXiv preprint arXiv:2404.10346*.

Joonwon Jang, Jaehee Kim, Wonbin Kweon, and Hwanjo Yu. 2024. Verbosity-aware rationale reduction: Effective reduction of redundant rationale via principled criteria. *arXiv preprint arXiv:2412.21006*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, and 1 others. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Bowen Jiang, Yangxinyu Xie, Zhuoqun Hao, Xiaomeng Wang, Tanwi Mallick, Weijie J Su, Camillo J Taylor, and Dan Roth. 2024. A peek into token bias: Large language models are not yet genuine reasoners. *arXiv preprint arXiv:2406.11050*.

Mingyu Jin, Qinkai Yu, Dong Shu, Haiyan Zhao, Wenyue Hua, Yanda Meng, Yongfeng Zhang, and Mengnan Du. 2024. The impact of reasoning step length on large language models. *arXiv preprint arXiv:2401.04925*.

Aditya Kalyanpur, Kailash Karthik Saravanakumar, Victor Barres, Jennifer Chu-Carroll, David Melville, and David Ferrucci. 2024. Llm-arc: Enhancing llms with an automated reasoning critic. *arXiv preprint arXiv:2406.17663*.

Jikun Kang, Xin Zhe Li, Xi Chen, Amirreza Kazemi, Qianyi Sun, Boxing Chen, Dong Li, Xu He, Quan He, Feng Wen, and 1 others. 2024. Mindstar: Enhancing math reasoning in pre-trained llms at inference time. *arXiv preprint arXiv:2405.16265*.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, and 1 others. Training language models to self-correct via reinforcement learning, 2024. *URL https://arxiv.org/abs/2409.12917*.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, and 1 others. 2023. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, and 1 others. 2022. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857.

Chengpeng Li, Guanting Dong, Mingfeng Xue, Ru Peng, Xiang Wang, and Dayiheng Liu. 2024. Dotamath: Decomposition of thought with code assistance and self-correction for mathematical reasoning. *arXiv preprint arXiv:2407.04078*.

Minpeng Liao, Chengxi Li, Wei Luo, Wu Jing, and Kai Fan. 2024a. MARIO: MAth reasoning with code interpreter output - a reproducible pipeline. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 905–924, Bangkok, Thailand. Association for Computational Linguistics.

Minpeng Liao, Wei Luo, Chengxi Li, Jing Wu, and Kai Fan. 2024b. Mario: Math reasoning with code interpreter output–a reproducible pipeline. *arXiv preprint arXiv:2401.08190*.

Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, and 1 others. 2024a. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024b. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Jiacheng Liu, Andrew Cohen, Ramakanth Pasunuru, Yejin Choi, Hannaneh Hajishirzi, and Asli Celikyilmaz. 2023. Don't throw away your value model! generating more preferable text with value-guided monte-carlo tree search decoding. *arXiv preprint arXiv:2309.15028*.

Shengcai Liu, Caishun Chen, Xinghua Qu, Ke Tang, and Yew-Soon Ong. 2024c. Large language models as evolutionary optimizers. pages 1–8.

Zimu Lu, Aojun Zhou, Houxing Ren, Ke Wang, Weikang Shi, Junting Pan, Mingjie Zhan, and Hongsheng Li. 2024. Mathgenie: Generating synthetic data with question back-translation for enhancing mathematical reasoning of llms. *arXiv preprint arXiv:2402.16352*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, and 1 others. 2024. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36.

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

Jincheng Mei, Chenjun Xiao, Ruitong Huang, Dale Schuurmans, and Martin Müller. 2019. On principled entropy exploration in policy optimization. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 3130–3136.

Yingqian Min, Zhipeng Chen, Jinhao Jiang, Jie Chen, Jia Deng, Yiwen Hu, Yiru Tang, Jiapeng Wang, Xiaoxue Cheng, Huatong Song, and 1 others. 2024. Imitate, explore, and self-improve: A reproduction report on slow-thinking reasoning systems. *arXiv preprint arXiv:2412.09413*.

Mike Mirzayanov. 2025. Codeforces. Accessed: 2025-01-12.

Mistral AI. 2024. Mixtral-8x22b-v0.1. https://huggingface.co/mistralai/Mixtral-8x22B-v0.1.

Linyong Nan, Chiachun Hsieh, Ziming Mao, Xi Victoria Lin, Neha Verma, Rui Zhang, Wojciech Kryściński, Hailey Schoelkopf, Riley Kong, Xiangru Tang, Mutethia Mutuma, Ben Rosand, Isabel Trindade, Renusree Bandaru, Jacob Cunningham, Caiming Xiong, and Dragomir Radev. 2022. Fetaqa: Free-form table question answering. *Transactions of the Association for Computational Linguistics*, 10:35–49.

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*.

Open Source O1. 2025. Open-o1. Accessed: 2025-01-17.

OpenAI o1 Contributors. 2024. Learning to reason with llms. https://openai.com/index/learning-to-reason-with-llms/.

Skywork o1 Team. 2024. Skywork-o1 open series. https://huggingface.co/Skywork.

OpenAI. 2023. Gpt-4 technical report.

OpenAI. 2024a. Hello, GPT-4o. https://openai.com/index/hello-gpt-4o/.

OpenAI. 2024b. O-1: Optimization for language models with continuous integration. https://openai.com/o1/.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1470–1480, Beijing, China. Association for Computational Linguistics.

Pranav Putta, Edmund Mills, Naman Garg, Sumeet Motwani, Chelsea Finn, Divyansh Garg, and Rafael Rafailov. 2024. Agent q: Advanced reasoning and learning for autonomous ai agents. *arXiv preprint arXiv:2408.07199*.

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

QwenLM, QVQ. QVQ-72B Preview - QwenLM Blog. https://qwenlm.github.io/blog/qvq-72b-preview. Accessed: 2025-01-16.

QwenLM, QwQ. QWQ-32B Preview - QwenLM Blog. https://qwenlm.github.io/blog/qwq-32b-preview. Accessed: 2025-01-16.

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*.

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

Paulo Shakarian, Abhinav Koyyalamudi, Noel Ngu, and Lakshmivihari Mareedu. 2023. An independent evaluation of chatgpt on mathematical word problems (mwp). *arXiv preprint arXiv:2302.13814*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, and 1 others. 2024a. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, and 1 others. 2024b. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*, pages 31210–31227. PMLR.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*.

Mingyang Song, Zhaochen Su, Xiaoye Qu, Jiawei Zhou, and Yu Cheng. 2025. Prmbench: A fine-grained and challenging benchmark for process-level reward models. *arXiv preprint arXiv:2501.03124*.

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. 2024. To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning. *arXiv preprint arXiv:2409.12183*.

Hao Sun, Alihan Hüyük, and Mihaela van der Schaar. 2023a. Query-dependent prompt evaluation and optimization with offline inverse rl.

Xiaofei Sun, Linfeng Dong, Xiaoya Li, Zhen Wan, Shuhe Wang, Tianwei Zhang, Jiwei Li, Fei Cheng, Lingjuan Lyu, Fei Wu, and Guoyin Wang. 2023b. Pushing the limits of chatgpt on nlp tasks.

Xiaofei Sun, Xiaoya Li, Jiwei Li, Fei Wu, Shangwei Guo, Tianwei Zhang, and Guoyin Wang. 2023c. Text classification via large language models. *arXiv preprint arXiv:2305.08377*.

Xiaofei Sun, Xiaoya Li, Shengyu Zhang, Shuhe Wang, Fei Wu, Jiwei Li, Tianwei Zhang, and Guoyin Wang. 2023d. Sentiment analysis through llm negotiations. *arXiv preprint arXiv:2311.01876*.

Gokul Swamy, Christoph Dann, Rahul Kidambi, Zhiwei Steven Wu, and Alekh Agarwal. 2024. A minimaximalist approach to reinforcement learning from human feedback. *arXiv preprint arXiv:2401.04056*.

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

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, and 1 others. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, and 1 others. 2025. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*.

Qwen Team. 2024a. Qvq: To see the world with wisdom.

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

Ye Tian, Baolin Peng, Linfeng Song, Lifeng Jin, Dian Yu, Haitao Mi, and Dong Yu. 2024. Toward self-improvement of llms via imagination, searching, and criticizing. *arXiv preprint arXiv:2404.12253*.

Manan Tomar, Lior Shani, Yonathan Efroni, and Mohammad Ghavamzadeh. 2020. Mirror descent policy optimization. *arXiv preprint arXiv:2005.09814*.

Alan Wake, Albert Wang, Bei Chen, CX Lv, Chao Li, Chengen Huang, Chenglin Cai, Chujie Zheng, Daniel Cooper, Ethan Dai, and 1 others. 2024. Yi-lightning technical report. *arXiv preprint arXiv:2412.01253*.

Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi. 2023. Gpt-re: In-context learning for relation extraction using large language models. *arXiv preprint arXiv:2305.02105*.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. 2024a. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in Neural Information Processing Systems*, 37:95095–95169.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. 2024b. Math-shepherd: Verify and reinforce LLMs step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9426–9439, Bangkok, Thailand. Association for Computational Linguistics.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024c. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Qineng Wang, Zihao Wang, Ying Su, Hanghang Tong, and Yangqiu Song. 2024d. Rethinking the bounds of llm reasoning: Are multi-agent discussions the key? *arXiv preprint arXiv:2402.18272*.

Shuhe Wang, Beiming Cao, Shengyu Zhang, Xiaoya Li, Jiwei Li, Fei Wu, Guoyin Wang, and Eduard Hovy. 2023a. Sim-gpt: Text similarity via gpt annotated data. *arXiv preprint arXiv:2312.05603*.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023b. Gpt-ner: Named entity recognition via large language models. *arXiv preprint arXiv:2304.10428*.

Shuhe Wang, Shengyu Zhang, Jie Zhang, Runyi Hu, Xiaoya Li, Tianwei Zhang, Jiwei Li, Fei Wu, Guoyin Wang, and Eduard Hovy. 2024e. Reinforcement learning enhanced llms: A survey. *arXiv preprint arXiv:2412.10400*.

Tianduo Wang and Wei Lu. 2023. Learning multi-step reasoning by solving arithmetic tasks. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1229–1238.

Zilong Wang, Hao Zhang, Chun-Liang Li, Julian Martin Eisenschlos, Vincent Perot, Zifeng Wang, Lesly Miculicich, Yasuhisa Fujii, Jingbo Shang, Chen-Yu Lee, and Tomas Pfister. 2024f. Chain-of-table: Evolving tables in the reasoning chain for table understanding. *ICLR*.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, and 1 others. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Penghao Wu and Saining Xie. 2023. V*: Guided visual search as a core mechanism in multimodal llms. *arXiv preprint arXiv:2312.14135*.

Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. 2024. Inference scaling laws: An empirical analysis of compute-optimal inference for problem-solving with language models. *arXiv preprint arXiv:2408.00724*.

Zhiheng Xi, Dingwen Yang, Jixuan Huang, Jiafu Tang, Guanyu Li, Yiwen Ding, Wei He, Boyang Hong, Shihan Do, Wenyu Zhan, and 1 others. 2024. Enhancing llm reasoning via critique models with test-time and training-time supervision. *arXiv preprint arXiv:2411.16579*.

Wenyi Xiao, Zechuan Wang, Leilei Gan, Shuai Zhao, Wanggui He, Luu Anh Tuan, Long Chen, Hao Jiang, Zhou Zhao, and Fei Wu. 2024. A comprehensive survey of direct preference optimization: Datasets, theories, variants, and applications. *arXiv preprint arXiv:2410.15595*.

Yuxi Xie, Anirudh Goyal, Wenyue Zheng, Min-Yen Kan, Timothy P Lillicrap, Kenji Kawaguchi, and Michael Shieh. 2024. Monte carlo tree search boosts reasoning via iterative preference learning. *arXiv preprint arXiv:2405.00451*.

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

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024a. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Sohee Yang, Elena Gribovskaya, Nora Kassner, Mor Geva, and Sebastian Riedel. 2024b. Do large language models latently perform multi-hop reasoning? *arXiv preprint arXiv:2402.16837*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36.

Fei Yu, Anningzhe Gao, and Benyou Wang. 2023. Outcome-supervised verifiers for planning in mathematical reasoning. *arXiv preprint arXiv:2311.09724*.

Lifan Yuan, Ganqu Cui, Hanbin Wang, Ning Ding, Xingyao Wang, Jia Deng, Boji Shan, Huimin Chen, Ruobing Xie, Yankai Lin, and 1 others. 2024. Advancing llm reasoning generalists with preference trees. *arXiv preprint arXiv:2404.02078*.

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*.

Dan Zhang, Sining Zhoubian, Ziniu Hu, Yisong Yue, Yuxiao Dong, and Jie Tang. 2024a. Rest-mcts*: Llm self-training via process reward guided tree search. *arXiv preprint arXiv:2406.03816*.

Di Zhang, Xiaoshui Huang, Dongzhan Zhou, Yuqiang Li, and Wanli Ouyang. 2024b. Accessing gpt-4 level mathematical olympiad solutions via monte carlo tree self-refine with llama-3 8b. *arXiv preprint arXiv:2406.07394*.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and 1 others. 2023. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.

Xuan Zhang, Chao Du, Tianyu Pang, Qian Liu, Wei Gao, and Min Lin. 2024c. Chain of preference optimization: Improving chain-of-thought reasoning in llms. *Advances in Neural Information Processing Systems*, 37:333–356.

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

Zhenru Zhang, Chujie Zheng, Yangzhen Wu, Beichen Zhang, Runji Lin, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. 2025. The lessons of developing process reward models in mathematical reasoning. *arXiv preprint arXiv:2501.07301*.

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*.

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

A The Family of OpenAI o1 and DeepSeek R1

A.1 Different Versions

The o1 series represents a line of models, with each version designed with different model sizes, inference speeds, and prices to cater to diverse use cases. Additionally, each version has specific reasoning capabilities that allow it to perform better in various domains like mathematics, programming, and science. In parallel, the R1 series has evolved through a multi-stage training pipeline that not only enhances its reasoning capabilities but also improves output readability and computational efficiency. The following is a breakdown of the release dates, abilities, and features of the different o1 and R1 models:

- **o1-Preview**, released on September 12, 2024 offers advanced multi-step reasoning and excels in complex problem-solving tasks. It features the "chain of thought" process to enhance reasoning accuracy but requires higher computational resources and has slower inference speed.

- **o1-Mini**, released on September 12, 2024, is a faster, more cost-effective alternative to the o1-Preview, offering 80% lower costs while still providing good reasoning for tasks like coding and STEM problems. It's designed for users who prioritize speed over the depth of reasoning and is ideal for developers, students, and quick technical applications.

- **Full o1**, released on December 5, 2024, provides the highest level of reasoning power, with capabilities on par with PhD-level expertise in fields like math, science, and programming. It's the most accurate and reliable option for professionals requiring precise, multi-step analysis but comes with higher computational costs and slower performance.

- **o1-Lite**, to be released on January 2025, is a lightweight, cost-efficient version of the full o1, offering moderate reasoning abilities for general tasks at faster speeds. It's designed for small businesses, educational platforms, and individual users who need basic problem-solving capabilities at a reduced price, without the need for deep analysis or heavy computational resources.

- **R1-Zero**, released on January 20, 2025, is developed using pure reinforcement learning without any supervised fine-tuning. It demonstrated significant improvements in reasoning capabilities but encountered challenges in output readability and language consistency.

- **R1**, released on January 20, 2025, is an enhanced version which incorporates a modest amount of high-quality cold-start data to optimize the chain-of-thought process and improve user-friendly outputs. With these improvements, its performance reaches a level comparable to that of OpenAI-o1-1217.

- **R1-Distill**, released on January 20, 2025, employs model distillation to transfer the advanced reasoning abilities of R1 to smaller dense models. This distilled version maintains competitive performance on benchmarks such as AIME 2024 and MATH-500 while significantly reducing model size and computational resource requirements, making it ideal for deployment in resource-constrained environments.

A.2 Performances on Benchmarks

Both OpenAI o1 and DeepSeek R1 models have demonstrated remarkable performance on a wide range of challenging tasks, showcasing advanced reasoning capabilities that rival expert human performance in various domains. Below, we compare their results across several benchmarks. Note: The o1 scores listed here are based on the initial release reports.

- **AIME 2024 (Math):**
  - **o1:** Achieved a score of 13.9, placing it among the top 500 students nationally and above the cutoff for the USA Mathematical Olympiad, with a pass@1 accuracy of 74%.
  - **R1:** Recorded a pass@1 accuracy of 79.8%, reflecting its robust multi-step reasoning and precise problem-solving approach.

- **GPQA Diamond (Chemistry, Physics, and Biology):**
  - **o1:** Achieved a pass@1 accuracy of 78.0%, surpassing the performance of recruited human experts with PhDs and becoming the first model to achieve such a feat.
  - **R1:** Achieved a pass@1 accuracy of 71.5%, underscoring its competitive edge in scientific reasoning.

- **Programming Contests Hosted by Codeforces (Coding):**

  - **o1:** Achieved an Elo rating of 1807, performing better than 93% of competitors.
  - **R1:** Demonstrated an even higher competitive edge with an Elo rating of 2029, performing better than 96% of competitors.

- **MATH-500 (Math):**
  - **o1:** Achieved a pass@1 accuracy of 94.8% on the benchmark, demonstrating its advanced reasoning capabilities.
  - **R1:** Excelled with a pass@1 accuracy of 97.3%, further highlighting its strong mathematical reasoning abilities.

As shown in Table 1 and Figure 1, the evaluation of o1 demonstrates substantial gains over GPT-4o on a diverse array of reasoning-intensive benchmarks, including competition math, code-generation challenges, and domain-specific question answering. On average, o1 exhibits considerably higher pass@1 and consensus@64 accuracy than its predecessor, indicating that its targeted architectural and training improvements have led to more robust reasoning capabilities.

Figure 2 presents some human evaluation results between openai o1 and gpt-4o. Interestingly, the human preference evaluations reveal that o1 does not uniformly outperform gpt-4o across all domains. While o1 demonstrates a pronounced advantage in more structured, logic-driven tasks, such as computer programming, data analysis, and mathematical calculation (where it wins over 70% of all cases), it lags slightly behind gpt-4o in more subjective or stylistic tasks like personal writing. For editing text, o1 and gpt-4o are comparable, suggesting that the model's improvements in reasoning do not necessarily translate to an equally strong edge in language polish or creative composition.

Similarly, R1 achieves performance that is competitive with, or even surpasses, that of o1 in several key areas. While o1 has set state-of-the-art benchmarks—such as outperforming PhD-level experts on GPQA Diamond and securing a top-500 rank on the AIME 2024 exam—R1 consistently delivers high accuracy on similar tasks. Detailed comparison results between o1 and R1 can be found in Figure 3.

B Training Details

As discussed earlier, reasoning datasets are meticulously structured, often representing step-by-step problem-solving processes. Effectively training LLMs on such datasets requires methodologies that maximize the utility of each reasoning step, whether correct or erroneous. This section explores three key training paradigms designed to leverage reasoning datasets for training o1-like reasoning LLMs:

1 **Supervised Fine-tuning (SFT)**: A foundational technique that refines pre-trained LLMs by explicitly teaching structured reasoning patterns through labeled (INSTRUCTION, ANSWER) pairs.

2 **Reinforcement Learning from Human Feedback (RLHF)**: A refinement approach that aligns LLM outputs with human preferences or quality signals, further enhancing reasoning skills through iterative optimization.

3 **Direct Preference Optimization (DPO)**: A simplified alternative to RLHF that directly optimizes fine-tuned LLMs for preferred reasoning outputs without requiring intermediate reward modeling.

A summary of each of these paradigms is presented in Table 3. Below we present their methodologies, strengths, and contributions to reasoning-focused training in details.

B.1 Supervised Fine-tuning

Supervised Fine-tuning (SFT) serves as the cornerstone for developing reasoning capabilities in LLMs. By utilizing structured (*instruction*, *answer*) pairs, SFT provides explicit guidance, enabling models to learn systematic reasoning patterns and produce accurate outputs across complex reasoning tasks. The process typically begins with a pre-trained LLM, which embodies extensive general knowledge and linguistic understanding. SFT fine-tunes these models on task-specific datasets, emphasizing logical reasoning, problem-solving, and domain-specific expertise. These datasets often reflect deterministic reasoning frameworks, enabling the model to generate consistent and interpretable outputs for tasks such as mathematical problem-solving, program synthesis, and logical deduction. To further optimize performance, SFT is frequently integrated with complementary training paradigms. For instance, multi-task fine-tuning leverages diverse datasets to improve generalization, while curriculum learning structures training

Dataset	Metric	gpt-4o	o1-preview	o1
Competition Math AIME (2024)	cons@64	13.4	56.7	83.3
	pass@1	9.3	44.6	74.4
Competition Code CodeForces	Elo	808	1,258	1,673
	Percentile	11.0	62.0	89.0
GPQA Diamond	cons@64	56.1	78.3	78.0
	pass@1	50.6	73.3	77.3
Biology	cons@64	63.2	73.7	68.4
	pass@1	61.6	65.9	69.2
Chemistry	cons@64	43.0	60.2	65.6
	pass@1	40.2	59.9	64.7
Physics	cons@64	68.6	89.5	94.2
	pass@1	59.5	89.4	92.8
MATH	pass@1	60.3	85.5	94.8
MMLU	pass@1	88.0	90.8	92.3
MMMU (val)	pass@1	69.1	n/a	78.2
MathVista (testmini)	pass@1	63.8	n/a	73.9

Table 1: Official evaluation results of o1 on typical benchmarks (o1 Contributors, 2024).

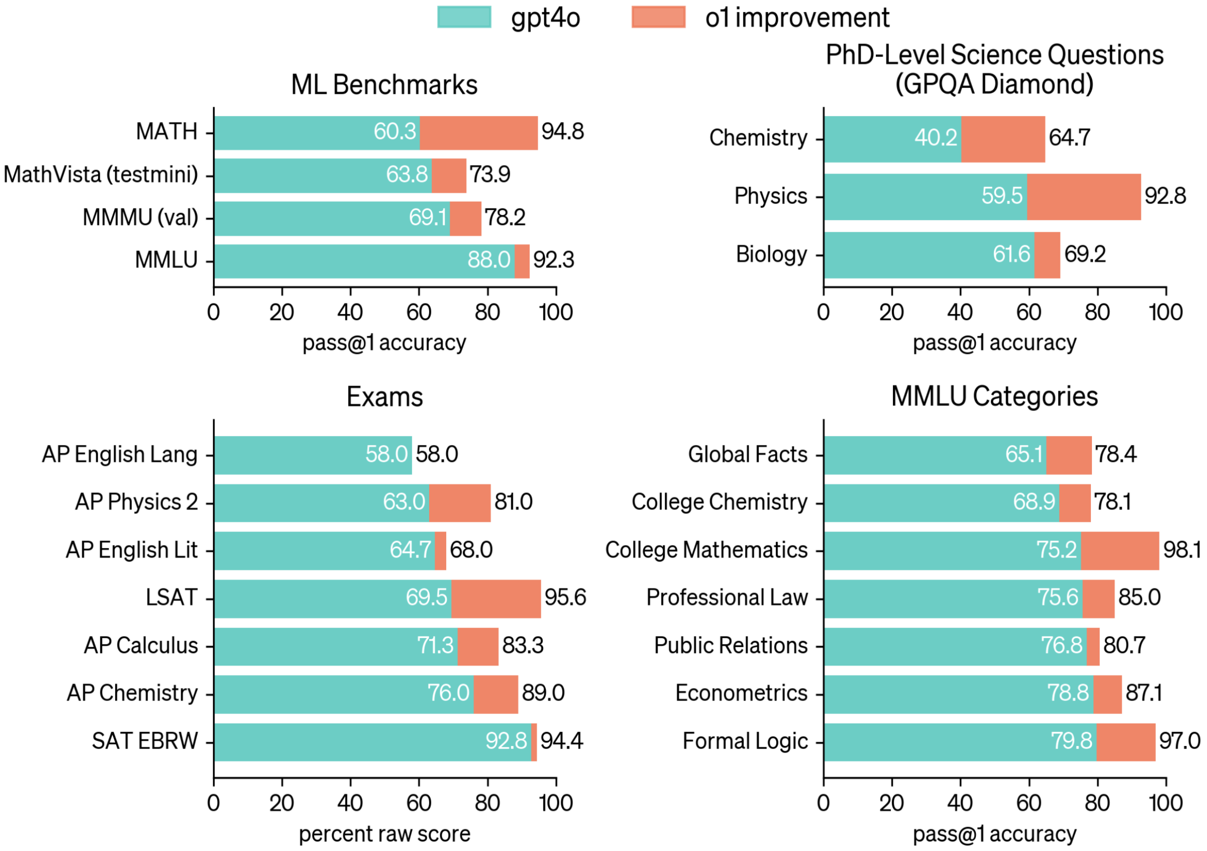

Figure 1: Official sub-category evaluation results of o1 on typical benchmarks (o1 Contributors, 2024).

data to progressively increase task difficulty. This adaptability allows SFT to be tailored to specific reasoning requirements, making it a versatile and essential component of LLM training.

B.2 Reinforcement Learning with Human Feedback (RLHF)

Reinforcement Learning from Human Feedback (RLHF) has emerged as a critical paradigm for aligning large language models with human preferences, enabling improved reasoning and alignment

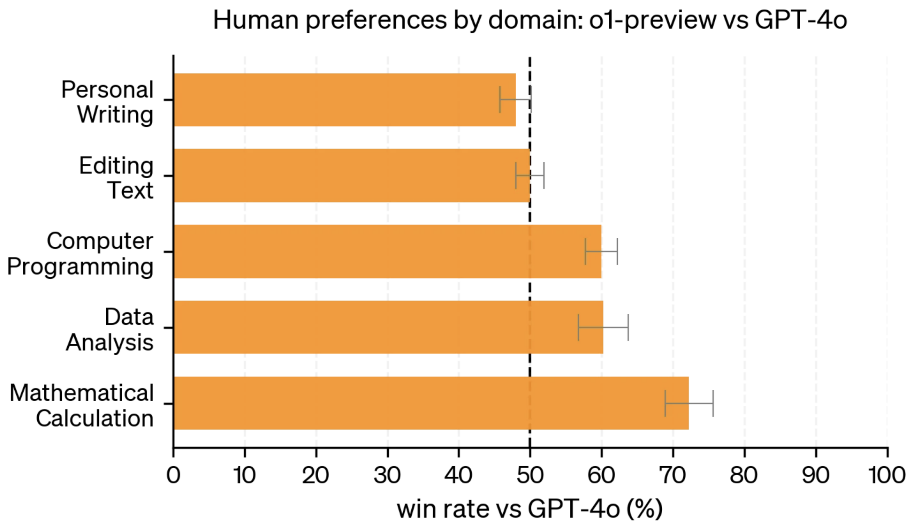

Figure 2: Official human evaluation results of o1 (o1 Contributors, 2024).

Dataset	Data Source	Data Scale	Machine Generated	Human Generated	Open-source
PRM800K (Swamy et al., 2024)[1]	MATH	800K annotations	✓	✓	✓[1]
O1-Journey (Qin et al., 2024)[2]	MATH, PRM800K	677 instances	✓	✓	✓[2]
Self-Explore (Hwang et al., 2024a)	GSM8K, MATH	Model-specific	✓	✗	✗
MARIO (Liao et al., 2024b)[3]	GSM8K, MATH, MetaMath	28.8K instances	✓	✓	✓[3]
MathGenie (Lu et al., 2024)	GSM8K, MATH	170K qa pairs	✓	✗	✗
DeepSeekMath (Shao et al., 2024b)[4]	AlgebraicStack, arXiv, GitHub	120B tokens	✓	✓	✓[4]
Compute-Optimal Sampling (Bansal et al., 2024)	GSM8K, MATH, etc.	Model-specific	✓	✗	✗
MathScale (Tang et al., 2024)[5]	GSM8K, MATH	2M qa pairs	✓	✗	✓[5]
G-LLaVA (Gao et al., 2023)[6]	Geometry3K, GeoQA+	170K instances	✓	✗	✓[6]

[1] https://github.com/openai/prm800k
[2] https://github.com/GAIR-NLP/O1-Journey
[3] https://github.com/MARIO-Math-Reasoning/MARIO
[4] https://github.com/deepseek-ai/DeepSeek-Math
[5] https://github.com/XylonFu/MathScale
[6] https://github.com/pipilurj/G-LLaVA

Table 2: An overview of currently widely-adopted reasoning datasets for LLMs.

capabilities. By incorporating iterative feedback and leveraging reinforcement learning techniques, RLHF enhances the models' ability to evaluate, refine, and generate outputs aligned with human expectations. Recent works have advanced this approach by introducing innovative frameworks for training reward models, integrating guided decoding, and enabling self-improvement without reliance on extensive human annotations.

B.3 Direct Preference Optimization (DPO)

Direct Preference Optimization (DPO) (Rafailov et al., 2024; Xiao et al., 2024; Amini et al., 2024) is an emerging training paradigm designed as a simpler alternative to RLHF. Unlike RLHF, which relies on reward modeling and reinforcement learning algorithms like Proximal Policy Optimization (PPO), DPO directly optimizes a language model's outputs to align with human preferences by fine-tuning the model on comparison data. This approach eliminates the complexity of learning a reward function and instead leverages pairwise preference data to improve the quality and alignment of generated outputs, offering an efficient and scalable solution for enhancing reasoning capabilities in LLMs.

C Inference Details

Multi-step reasoning tasks are prone to errors at any step, as small mistakes can cascade into incorrect final answers. To address this, reasoning LLMs often generate multiple reasoning paths for a given input question during the inference stage and choose the answer that aligns best with the most logically consistent and broadly supported reasoning steps.

In this section covers the following three widely used techniques in reasoning LLMs, which we

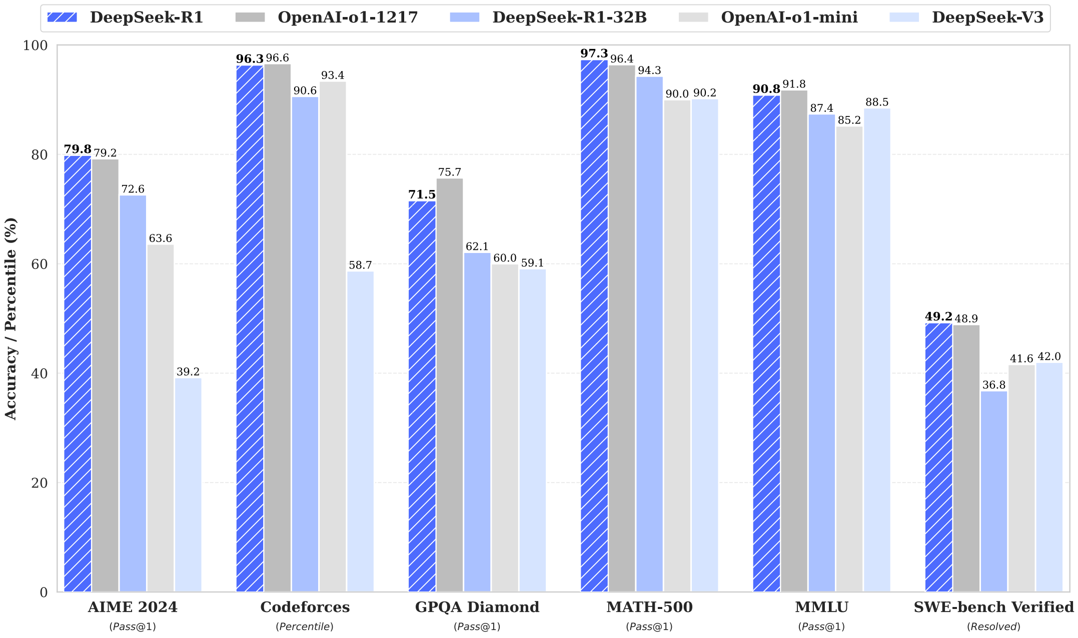

Figure 3: Results of DeepSeek-R1 (DeepSeek-AI et al., 2025). The figure is adapted from DeepSeek-AI et al. (2025).

think are crucial in building o1-like reasoning models:

1 **Tree of Thoughts**, which represents the reasoning process as a tree structure and explores various branches to determine the most effective path.

2 **Automated Reasoning Critic**, which employs a trained critic model to evaluate and validate the reasoning steps generated by the LLMs.

3 **Self-Correction**, where the LLM mimics human critical thinking by iteratively reviewing, identifying errors, and refining its reasoning steps to enhance accuracy and logical consistency.

In addition to these three inference techniques, we will also explore "Inference Scaling Laws", which provide insights into how reasoning performance improves as inference time increases, enabling us to balance the trade-off between computational efficiency and reasoning accuracy. An overview of this section is provided in Table 4.

C.1 Tree of Thoughts

In complex reasoning tasks, systematically exploring multiple paths of thought is crucial for finding optimal solutions. Tree of Thoughts represents the reasoning process as a tree structure, enabling models to systematically explore and evaluate different solution branches. This approach not only helps models find optimal solutions but also prevents them from getting stuck in local optima. The framework employs various tree search strategies, starting from fundamental methods like Breadth-first and Depth-first Search (Yao et al., 2024; Yuan et al., 2024; Feng et al., 2023) and advancing to more sophisticated approaches such as Monte Carlo Tree Search (Zhang et al., 2024b; Tian et al., 2024; Xie et al., 2024; Chen et al., 2024b; Zhang et al., 2024a).

C.2 Automated Reasoning Critic

In the reasoning process of LLMs, the ability to identify and correct faulty reasoning steps is essential. Automated Reasoning Critic (Barto et al., 1983; Saunders et al., 2022) introduces dedicated critic models to evaluate the correctness and logical consistency of reasoning steps, thereby improving the reliability of the reasoning process. This approach mirrors how humans validate their thinking process when solving complex problems, providing a systematic way to assess and improve the quality of generated reasoning.

C.3 Self-Correction

Errors in reasoning often accumulate progressively, where small mistakes can lead to significant devi-

Paper	Key Innovation	Main Techniques
Sec. B.1 Supervised Fine-tuning (SFT)		
ToRA (Gou et al., 2023)	Tool-integrated Mathematical Focused Reasoning Agents	Imitation Learning, Output Space Shaping
AlphaLLM (Tian et al., 2024)	Self Improving Training	SFT with Monte Carlo Tree Search
MARIO (Liao et al., 2024a)	Mathematical Reasoning Framework	Data Enhancement with GPT-4, Human Review, and Self-training
Sec. B.2 Reinforcement Learning from Human Feedback (RLHF)		
Self-Critiquing (Saunders et al., 2022)	Fundations of Language Model Self-Critiquing	AI-assisted Human Feedback,
OVM (Yu et al., 2023)	Evaluating the Potential of Incomplete Reasoning Paths	Outcome-supervised Value Models
PPO-MCTS (Liu et al., 2023)	Value-Guided Decoding trhough PPO	Proximal Policy Optimization, Monte Carlo Tree Search
MATH-SHEPHERD (Wang et al., 2024b)	Eliminatioin of Human Annotation	step-wise verification through MCTS
Qwen-2.5-math (Zhang et al., 2025)	Enhanced Process Reward Model	LLM-as-a-judge
Roadmap to o1 (Zeng et al., 2024)	Combination of Various Techniques to Reproduce o1	Policy Initialization, Reward Shaping, Policy Gradient
PROCESSBENCH (Zheng et al., 2024)	Benchmark for Error Identification in Mathematical Reasoning	Step-level Error Detection
PRMBENCH (Song et al., 2025)	Fine-grained Benchmark for Process Reward Models	Multi-dimensional Evaluation Benchmark
Sec. B.3 Direct Preference Optimization (DPO)		
CPO (Zhang et al., 2024c)	Fine-tuning CoT reasoning with ToT	Tree-of-Thoughts framework, Paired preference fine-tuning
SVPO (Chen et al., 2024c)	Step-level preferences for reasoning improvement	MCTS for step-level preferences, Value model integration
PPO-MCTS (Liu et al., 2023)	Value-guided decoding during inference	PPO value network MCTS
Self-Explore (Hwang et al., 2024b)	Self-guided Learning with Fine-grained Rewards	First-pit Identifications
Agent Q (Putta et al., 2024)	Enhanced Agentic reasoning	MCTS with AI feedback, Offline DPO

Table 3: An overview of LLM Training Techniques.

ations in final results. The Self-Correction mechanism (Huang et al., 2023; Madaan et al., 2024) enables large language models to engage in self-reflection and correction, similar to human critical thinking. Through iteratively reviewing, identifying errors, and refining reasoning steps, this approach improves the accuracy of final answers. This methodology is particularly effective for tasks requiring multi-step reasoning.

C.4 Inference Scaling Laws

Inference Scaling Laws examine the relationships between inference time, computational resource allocation, and reasoning performance. Research on inference-time compute scaling suggests that spending more computational resources during inference can significantly enhance model performance, a principle applied in o1 models. Very recently, DeepSeek models (DeepSeek-R1-Lite-Preview, 2024) also demonstrate the inference scal-

Question: If $a > 1$, then the sum of the real solutions of $\sqrt{a - \sqrt{a+x}} = x$ is equal to

Response: <think>
To solve the equation $\sqrt{a - \sqrt{a+x}} = x$, let's start by squaring both $\cdots$
$\left(\sqrt{a - \sqrt{a+x}}\right)^2 = x^2 \implies a - \sqrt{a+x} = x^2.$
Rearrange to isolate the inner square root term:
$(a - x^2)^2 = a + x \implies a^2 - 2ax^2 + (x^2)^2 = a + x \implies x^4 - 2ax^2 - x + (a^2 - a) = 0$
$\ldots$
Wait, wait. Wait. That's an aha moment I can flag here.
Let's reevaluate this step-by-step to identify if the correct sum can be $\cdots$
We started with the equation:
$\sqrt{a - \sqrt{a+x}} = x$
First, let's square both sides:
$a - \sqrt{a+x} = x^2 \implies \sqrt{a+x} = a - x^2$
Next, I could square both sides again, treating the equation: $\cdots$
$\ldots$

Figure 4: An intriguing "aha moment" observed in an intermediate version of DeepSeek-R1-Zero (DeepSeek-AI et al., 2025), where the model demonstrates the ability to rethink its approach using an anthropomorphic tone. This moment not only highlights the model's evolving reasoning capabilities but also underscores the remarkable potential and elegance of reinforcement learning in fostering advanced cognitive behaviors. The figure is adapted from DeepSeek-AI et al. (2025).

ing law, as shown in Figure 5. This sub-section explores how understanding these laws can guide the optimal configuration of computational resources, providing theoretical insights for maximizing reasoning capabilities while maintaining efficiency.

D O1-like Reasoning LLMs

D.1 Marco-o1

Macro-o1 (Zhao et al., 2024), developed by Alibaba, explores the generalization capabilities of the o1 model in open-ended domains lacking clear standards or quantifiable rewards, unlike disciplines with standard answers such as mathematics, physics, or coding. It employs techniques including CoT fine-tuning, Monte Carlo Tree Search (MCTS), reflective processes, and advanced reasoning to address complex real-world challenges. Experimental results indicate that Macro-o1 exhibits o1-like reasoning abilities, achieving significant accuracy gains of +6.17% on the MGSM (English) dataset and +5.60% on the MGSM (Chinese) dataset, highlighting its improved reasoning performance. Additionally, it pioneers the application of large reasoning models (LRMs) in machine translation, particularly excelling in translating slang expressions, while investigating inference-time scaling laws in multilingual contexts.

Macro-o1's core idea is to first fine-tune a base LLM using a combined dataset and then perform inference with MCTS to expand the solution space. The fine-tuning dataset comprises three components: the refined Open-O1 CoT Dataset (O1, 2025), a Marco-o1 CoT Dataset generated via MCTS, and the Marco Instruction Dataset. During inference, two action strategies are applied within the MCTS framework: "step as action" for efficient exploration and "mini-step as action" (32 or 64 tokens) for finer granularity. The latter broadens the solution space by incorporating more detailed reasoning steps, enhancing the model's capacity to handle complex tasks. A reflection mechanism further improves performance by prompting the model to reevaluate its reasoning with phrases like: "Wait! Maybe I made some mistakes! I need to rethink from scratch." This self-reflection helps correct errors in difficult problems. The final solutions are selected based on calculated confidence scores.

D.2 o1-Coder

o1-Coder (Zhang et al., 2024d), developed by Beijing Jiaotong University, aims to evaluate the performance of OpenAI's o1 model in coding tasks by adapting it to better address programming-related problem-solving challenges. The goal is to enhance the model's capabilities through focused im-

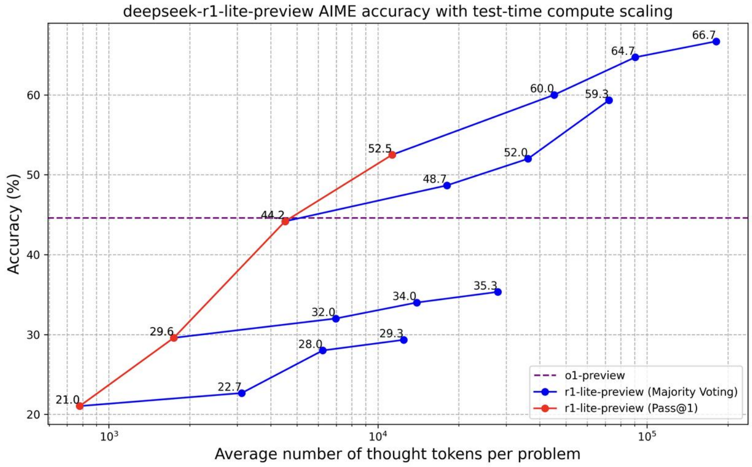

Figure 5: DeepSeek-R1-Lite-Preview (DeepSeek-R1-Lite-Preview, 2024) shows consistent score improvements on AIME as the length of reasoning increases. The figure is adapted from (DeepSeek-R1-Lite-Preview, 2024).

provements. o1-Coder combines RL with MCTS to strengthen the model's System-2 reasoning abilities. The system involves training a Test Case Generator (TCG) for standardized testing, utilizing MCTS to generate reasoning-augmented code data, and iteratively refining the policy model to evolve from pseudocode to fully functional code.

o1-Coder consists of six key steps: 1. The process begins by training a TCG, denoted as γ_{TCG}, to automatically create test cases based on the given problem descriptions. 2. Next, MCTS is applied to the original code dataset, producing a new dataset $\mathcal{D}_{\text{process}}$. This dataset incorporates reasoning processes and a validity indicator to distinguish correct from incorrect steps. 3. The dataset is then used to fine-tune the policy model π_θ, encouraging it to adopt a "think before acting" approach. 4. The reasoning data from the previous step is used to initialize a process reward model (PRM), ρ_{PRM}, which evaluates the quality of reasoning steps. 5. Both the ρ_{PRM}, and γ_{TCG}, provide rewards based on process and outcome, respectively. This enables reinforcement learning to iteratively update the policy model π_θ. 6. Finally, the updated policy model generates new reasoning data, which is used to refine the ρ_{PRM}, creating a self-improving iterative cycle through steps 4, 5, and 6. This approach forms a feedback loop that enhances the model's reasoning and coding performance over time.

E Multi-modal Reasoning LLMs

E.1 Insight-V

Insight-V (Dong et al., 2024) is a framework designed to enhance the multi-step visual reasoning capabilities of MLLMs by constructing reliable multi-step reasoning data and developing a refined training process. The authors propose the following techniques: (1) a flexible strategy for generating multi-step reasoning data for complex multi-modal tasks, (2) a multi-agent system that divides task handling processes into reasoning and summarization parts to enhance response quality, and (3) a two-stage training process to better cultivate agents' abilities. The data generation pipeline utilizes a progressive method to create formatted multi-step reasoning data with various reasoning paths and a multi-level assessment system to evaluate the quality of the generated reasoning data and divide them into different datasets. The multi-agent system employs a reasoning agent to generate detailed reasoning steps and a summarization agent to extract core logic and generate concise response. The two-stage training pipeline involves SFT of a base MLLM to develop the reasoning and summarization agents, followed by iterative DPO to align the reasoning agent with human preferences. The proposed techniques lead to significantly improved performance on complex multi-modal visual reasoning benchmarks and effortlessly retains or elevates its performance on multi-modal perception tasks.

E.2 Sketchpad

Sketchpad (Hu et al., 2024) is a framework designed to enhance the multi-step multi-modality reasoning process by inserting image processing behaviors in the inference phase. The authors introduce this technique to remedy the shortcomings of current CoT and tool-use paradigms, which rely

Paper	Key Innovation	Main Techniques
Sec. C.1 Tree of Thoughts		
Tree of Thoughts (Yao et al., 2024)	1st tree-structured reasoning framework	BFS/DFS search, Self-evaluation, Backtracking
EURUS (Yuan et al., 2024)	Tree-structured alignment dataset	ULTRAINTERACT dataset, Preference learning
TS-LLM (Feng et al., 2023)	AlphaZero-inspired framework	Markov Decision Process (MDP) formulation, Deep search (64 depth)
MCTSr (Zhang et al., 2024b)	Enhanced MCTS for math	Self-reflection, Dynamic pruning, Upper Confidence Bound (UCB)
ALPHALLM (Tian et al., 2024)	Self-improvement framework	Option-level MCTS, Adaptive branching, State merging
MCTS-DPO (Xie et al., 2024)	Step-level preference learning w/ MCTS	MCTS guided exploration, DPO updates, Step-level signals
AlphaMath (Chen et al., 2024b)	Self-supervised MCTS reasoning	Step-level value model, Beam search, Self-improvement
ReST-MCTS* (Zhang et al., 2024a)	Process-reward enhanced MCTS	Per-step rewards, Dual optimization, Dynamic exploration
Sec. C.2 Automated Reasoning Critic		
CriticGPT (McAleese et al., 2024)	LLM-based code critique	Tampered data generation, RLHF, Bugs identifying
AutoMathCritique (Xi et al., 2024)	Two-player math reasoning	Dynamic supervision, Error generation
LLM-ARC (Kalyanpur et al., 2024)	Neuro-symbolic reasoning	LLM + reasoning engine integration, Answer Set Programming (ASP) solver
Sec. C.3 Self-Correction		
SCoRe (Kumar et al.)	Multi-turn RL framework	Self-generated data, Two-stage training, Reward shaping
CoSC (Gao et al., 2024)	Embedded self-correction	Program generation, execution, and verification, Two-phase fine-tuning
DotaMath (Li et al., 2024)	Integrated mathematical reasoning	Multi-round correction, Python executor, Task decomposition
Sec. C.4 Inference Scaling Laws		
Scale-Compute (Snell et al., 2024)	Test-time compute analysis	Compute-optimal strategy, Process-based Reward Models (PRMs) search
REBASE (Wu et al., 2024)	Reward balanced search	Dynamic tree optimization, Pruning with a reward model
LLMonkeys (Brown et al., 2024)	Sampling analysis	Repeated sampling, Exponentiated power law, Cost optimization
STILL-2 (Min et al., 2024)	Three-phase training	Imitation, Exploration, Self-improvement, Long-form Thought Dataset
MindStar (Kang et al., 2024)	No-tuning enhancement	PRM-guided search, Dynamic exploration, Levin tree search

Table 4: An overview of LLM Inference Techniques.

solely on text during intermediate reasoning stages. Unlike prior works where language models (LMs) generate images via text-to-image models, the authors equip LMs with the ability to draw lines,

Model	Organization	# Params	Open Source	Report/Paper Available	Comparison with o1
Gemini 2.0 Flash (Google AI)	Google	-	✗	✗	✗
QVQ-72B-Preview (QwenLM, QVQ)	Alibaba	72B	✓[1]	✗	✓
Marco-o1 (Zhao et al., 2024)	Alibaba	7B	✓[2]	✓[8]	✗
Skywork o1 (o1 Team, 2024)	KUNLUN	8B	✓[3]	✗	✗
QwQ-32B-Preview (QwenLM, QwQ)	Alibaba	32B	✓[4]	✗	✓
o1-Coder (Zhang et al., 2024d)	北京交通大学 BEIJING JIAOTONG UNIVERSITY	-	✓[5]	✓[9]	✗
rStar-Math (Guan et al., 2025)	Microsoft	1.5B,3B,7B	✓[6]	✓[10]	✓
Kimi-k1.5 (Team et al., 2025)	Moonshot AI	-	✗	✓[11]	✓
DeepSeek-R1 (DeepSeek-AI et al., 2025)	deepseek	671B-A31B	✓[6]	✓[12]	✓

[1] https://huggingface.co/Qwen/QVQ-72B-Preview
[2] https://github.com/AIDC-AI/Marco-o1
[3] https://huggingface.co/Skywork/Skywork-o1-Open-Llama-3.1-8B
[4] https://huggingface.co/Qwen/QwQ-32B-Preview
[5] https://github.com/ADaM-BJTU/o1-Coder
[6] https://github.com/zhentingqi/rStar
[7] https://huggingface.co/deepseek-ai/DeepSeek-R1
[8] https://arxiv.org/pdf/2501.04519
[9] https://arxiv.org/pdf/2411.14405
[10] https://arxiv.org/pdf/2412.00154
[11] https://arxiv.org/pdf/2501.12599
[12] https://arxiv.org/pdf/2501.12948

Table 5: Overview of recent efforts in reproducing OpenAI o1. The format '671B-A31B' refers to MoE models with 671B total and 31B active parameters.

boxes, marks, etc., mimicking human sketching and thus improving the reasoning process. Additionally, to improve visual perception and reasoning, Sketchpad leverages specialized vision models to optimize its sketching process (*e.g.*, using object detection models to draw bounding boxes and segmentation models to create masks). Evaluation experiments for this work were conducted on several kinds of benchmark datasets, covering topics such as geometry, functions, graphs, chess, and challenging visual reasoning tasks. Compared to powerful baseline models without applying proposed technique, Sketchpad significantly boosts performance across all tasks. Specifically, it improves average performance on math tasks by 12.7% and visual tasks by 8.6%. Using the proposed technique, GPT-4o achieves the best performance across all benchmarks, such as V*Bench (Wu and Xie, 2023) with a score of 80.3%, and visual correspondence at 80.8%.

E.3 ChartPaLI-5B

ChartPaLI-5B (Carbune et al., 2024) is a MLLM based on PaLI3-5B (Chen et al., 2023) designed to improve the chart-related reasoning abilities of VLMs. To narrow the reasoning ability gap between smaller VLMs and LLMs, the authors propose a method to transfer knowledge from LLMs. First, they adopt the improved chart-to-table conversion (**?**) and use this refined chart representation to undergo pre-training. Then, they construct a dataset that is 20 times larger than the original training set. Following that, the authors design reasoning steps with table representations of charts to strengthen both reasoning and numerical capabilities. Finally, they fine-tune the model using a multitask loss (Hsieh et al., 2023) on the constructed datasets. These datasets contains reasoning steps generated by more powerful LLMs, enabling the transfer of reasoning abilities. ChartPaLI-5B achieves state-of-the-art performance on ChartQA and significantly improves performance on PlotQA and FigureQA. Moreover, even without an upstream OCR system, ChartPaLI-5B surpasses much larger models like PaLIX-55B while maintaining similar inference times as its base model PaLI3-5B. Additionally, by adopting a straightforward program-of-thought prompt (Chen et al., 2022) to refine the logic chain, ChartPaLI-5B even outperforms the recently released Gemini Ultra and GPT-4V.

E.4 SpatialVLM

SpatialVLM (Chen et al., 2024a) is a framework designed to enhance the spatial understanding and reasoning capabilities of VLMs by leveraging out-of-the-box vision models to generate spatial annotations on the training data. This work tackles the difficulties encountered by VLMs in spatial comprehension and reasoning, particularly in tasks involving the interpretation of numerical relationships between physical entities, such as variations in size and spatial distance. The authors suggest that this limitation arises from the lack of annotation of spatial information in the training data. The proposed solution is to enhance VLMs by training them on a large-scale spatial reasoning dataset. First, they develop an automated framework for generating visual question answering (VQA) data with rich spatial information annotations. By integrating techniques such as region captioning and segmentation, this framework annotates real-world data at scale and formats it for training VLMs on diverse tasks. With this framework and 10 million real-world images, they finally gain 2 billion VQA examples. Next, they explore several key factors in the training process, such as model architecture and data quality, trying to develop an optimized training mechanism. The natural language interface of a powerful VLM using SpatialVLM can support complex spatial reasoning by facilitating a CoT process, making it efficient for tackling sophisticated spatial problems. It also enables the model to serve as an open-vocabulary reward annotator for tasks involving rearrangement. Training a VLM on the dataset created using the proposed techniques improves the model's qualitative and quantitative spatial understanding and reasoning capabilities, enabling it to achieve significant performance improvements on related tasks. VLMs applying this technique can further carry out more complex spatial perception applications , thanks to their abilities to make quantitative estimations.

E.5 Chain-of-Table

Chain-of-Table (Wang et al., 2024f) is a framework designed to improve the reasoning abilities of LLMs when working with table-based data. While CoT and similar methods integrate reasoning processes as textual context, effectively incorporating tabular data into this reasoning chain remains a challenge. Table-based reasoning involves extracting semantics from unstructured questions and partially structured tabular information, which differs from conventional reasoning tasks. The authors propose a method that directly utilizes tabular data in the intermediate steps of the reasoning chain, carrying out progressive reasoning through tabular operations, thereby forming a chain of intermediate tables. The authors employ in-context learning to teach the model to use table operations (*e.g.*, adding columns, filtering rows, or grouping) step by step to refine or simplify the table. This enables LLMs to dynamically plan each subsequent action based on the intermediary tables in the operation history. Such a process better utilizes the semantics of the table that is continuously optimized during reasoning. Chain-of-Table sets a new benchmark in performance on the WikiTQ (Pasupat and Liang, 2015), FeTaQA (Nan et al., 2022), and TabFact (Chen et al., 2020) datasets.

E.6 QVQ-72B-Preview

QVQ-72B-Preview (Team, 2024a) is a MLLM built upon Qwen2-VL-72B (Wang et al., 2024c), designed to enhance visual reasoning capabilities through step-by-step reasoning. It aims to improve LLMs' cognitive abilities by incorporating visual understanding. However, few technical details are currently available. The team mainly presents evaluation results and discusses the model's limitations. QVQ-72B-Preview has achieved impressive results across several benchmarks, including an outstanding 70.3% on the MMMU benchmark, demonstrating QVQ's strong ability in multi-domain reasoning and comprehension. The model's substantial improvements on MathVision (Wang et al., 2024a) highlight its advancements in mathematical problem-solving. OlympiadBench (He et al., 2024) further showcases its enhanced capability to address complex challenges. Despite these achievements, the model has several limitations. For instance, it may mix languages or enter recursive reasoning loops, affecting response clarity and conciseness. Although it has made advancements in visual reasoning, it struggles with multi-step reasoning, occasionally hallucinating or losing focus, and does not outperform Qwen2-VL-72B in basic recognition tasks. Additionally, the model is limited to single-round dialogues and image outputs, with no support for video inputs.

F Evaluation Benchmarks

GPQA. The GPQA (Rein et al., 2023) dataset provides a challenging benchmark for evaluating reasoning abilities, particularly in scientific domains such as physics, chemistry, and biology. It consists of graduate-level multiple-choice questions carefully crafted by domain experts to test the limits of human and AI performance. What makes GPQA unique is its difficulty: even experts with PhDs or those pursuing advanced degrees in

Model or Framework	Base Model	Input Modality	Pretraining Data Scale	Fine-tuning Data Scale	Open-source
Insight-V (Dong et al., 2024)	Qwen-2.5-7B	Text/Image	558K	4M images	✓[1]
LLaVA-CoT-11B (Xu et al., 2024)	Llama-3.2-11B-Vision-Instruct	Text/Image	-	99K	✓[2]
Sketchpad (Hu et al., 2024)	GPT-4o	Text/Image	-	-	✓[3]
ChartPaLI-5B (Carbune et al., 2024)	PaLI-3	Text/Image(chart)	2.37M	544.9K	✗
SpatialVLM (Chen et al., 2024a)	PaLM 2-E	Text/Image(3d)	-	-	✓[5]
Chain-of-Table (Wang et al., 2024f)	PaLM 2-S, Llama-2-17B-chat	Text(table)	-	-	✓[4]
QVQ-72B-Preview (Team, 2024a)	Qwen2-VL-72B	Text/Image	-	-	✓[6]

[1] https://github.com/dongyh20/Insight-V
[2] https://github.com/PKU-YuanGroup/LLaVA-CoT
[3] https://github.com/Yushi-Hu/VisualSketchpad
[4] https://github.com/google-research/chain-of-table
[5] https://github.com/remyxai/VQASynth
[6] https://huggingface.co/Qwen/QVQ-72B-Preview

Table 6: An overview of emerging LLMs designed for multi-modal reasoning.

relevant fields achieve only 65% accuracy, which increases to 74% when accounting for errors identified retrospectively. Highly skilled non-experts, despite having unrestricted access to the internet, achieve a mere 34% accuracy. The dataset is also notably difficult for state-of-the-art AI systems like GPT-4, which achieves only 39% accuracy, significantly above random chance (25%). This makes GPQA an ideal testbed for evaluating large reasoning models. As the AI community continues to explore advanced reasoning capabilities, datasets like GPQA will be crucial in assessing whether AI models can handle tasks that are inherently difficult for both human experts and AI systems alike.

OlympiadBench. OlympiadBench (He et al., 2024) offers a comprehensive and rigorous benchmark for evaluating reasoning abilities, particularly in mathematics and physics, through a bilingual multimodal dataset. Comprising 8,476 challenging problems sourced from international Olympiads, Chinese Olympiads, and the Chinese College Entrance Exam (GaoKao), OlympiadBench pushes the boundaries of current AI models. Each problem is annotated with expert-level step-by-step reasoning, ensuring that the dataset captures the full depth of problem-solving processes. Additionally, OlympiadBench addresses a critical gap in existing benchmarks by incorporating multimodal reasoning, as many scientific tasks require not just textual analysis but also an understanding of visual or geometric information. With its rigorous design, OlympiadBench serves as an essential tool for assessing the true reasoning capabilities of state-of-the-art AI models, helping to guide future advancements in artificial general intelligence.

Minerva. Minerva (Lewkowycz et al., 2022) introduces a benchmark specifically focused on testing large language models in quantitative reasoning across various scientific domains, including mathematics, physics, chemistry, and biology. The dataset contains over 200 undergraduate-level problems drawn from MIT's OpenCourseWare (OCW) and other technical sources, providing a broad spectrum of challenges that require step-by-step reasoning and solution generation. Minerva pushes the boundaries of model performance by testing the ability to solve complex, real-world scientific problems without relying on external tools or solvers. The problems in Minerva involve not only natural language processing but also the integration of formal mathematical language, such as equations and diagrams, to model accurate problem-solving procedures. Minerva's diverse and robust set of problems offers a comprehensive platform for assessing how well AI systems can handle multi-step, quantitative reasoning tasks, providing a critical measure for the development of future AI assistants in scientific and engineering fields.

GSM8K. GSM8K (Cobbe et al., 2021) is a benchmark designed to evaluate the ability of language models to perform multi-step mathematical reasoning at the grade school level. It consists of 8.5K high-quality, linguistically diverse math word problems that cover a wide range of topics. Despite the simplicity of the underlying math concepts, the dataset poses significant challenges due to its high linguistic diversity, requiring models to demonstrate strong reasoning abilities in both interpreting natural language and solving mathematical problems. GSM8K provides a valuable resource for advancing the development of models capable of tackling elementary yet challenging quantitative reasoning tasks, serving as a key tool for testing the reasoning and problem-solving abilities of AI systems.

MATH. The MATH dataset (Hendrycks et al., 2021) presents a challenging benchmark specifically designed to evaluate the mathematical problem-solving abilities of machine learning models. Comprising 12,500 competition-level math problems from high school math competitions, MATH covers a broad range of topics including algebra, geometry, combinatorics, and number theory. Each problem is accompanied by a full step-by-step solution, enabling models to learn both the correct final answer and the reasoning process behind it. The dataset is particularly valuable for testing models' abilities to perform multi-step reasoning and generate coherent explanations. MATH's complexity, even for human experts, combined with its large scale and focus on structured problem-solving, makes it an essential benchmark for pushing the boundaries of AI's reasoning capabilities, particularly in the realm of mathematics.

AIME. The American Invitational Mathematics Examination (AIME) (AI-MO, 2025) serves as a prestigious benchmark for evaluating mathematical reasoning abilities, particularly for high school-level problem-solving. It is originally a selective 15-question, 3-hour exam that is open to students who perform in the top 5% of the AMC 12 exam (or top 2.5% of the AMC 10). The problems tested in the AIME primarily focus on algebra, geometry, trigonometry, number theory, probability, and combinatorics, and often require advanced problem-solving techniques not typically covered in standard high school curricula. For large models, the AIME dataset serves as an important benchmark for evaluating their capabilities in multi-step mathematical reasoning.

Codeforces. Codeforces (Mirzayanov, 2025) is a platform hosts regular programming contests, known as "Codeforces Rounds," which challenge participants to solve algorithmic problems under time pressure. The problems typically span a variety of topics in computer science, including graph theory, dynamic programming, data structures, and number theory, requiring strong analytical and computational reasoning skills. The Codeforces rating system, similar to the Elo system, evaluates contestants based on their performance across these contests. With divisions for different skill levels (Div. 1, Div. 2, Div. 3, and Div. 4), Codeforces offers a wide range of problems suitable for evaluating AI systems at various levels of difficulty. This makes Codeforces an excellent resource for assessing the ability of large models to solve algorithmic and coding problems, particularly those requiring multi-step, logical reasoning and optimization strategies.