Exploring the System 1 Thinking Capability of Large Reasoning Models

Wenyuan Zhang[1,2], **Shuaiyi Nie**[1,2], **Xinghua Zhang**[3], **Zefeng Zhang**[1,2], **Tingwen Liu**[1,2]
[1]Institute of Information Engineering, Chinese Academy of Sciences
[2]School of Cyber Security, University of Chinese Academy of Sciences
[3]Tongyi Lab, Alibaba Group
{zhangwenyuan, nieshuaiyi, zhangzefeng, liutingwen}@iie.ac.cn,
zhangxinghua.zxh@alibaba-inc.com

Abstract

This paper explores the *system 1* thinking capability of Large Reasoning Models (LRMs), the intuitive ability to respond efficiently with minimal token usage. While existing LRMs rely on long-chain reasoning and excel at complex tasks, their *system 1* thinking ability remains largely underexplored. This capability is essential as it reflects models' difficulty awareness and reasoning efficiency, both critical for real-world applications. We propose S1-Bench, a multi-domain, multilingual benchmark comprising model-simple *system 1* questions. Our investigation of 28 LRMs reveals under-accuracy and inefficiency on *system 1* problems. We find existing efficient reasoning methods either generalize poorly to simple questions or sacrifice performance for efficiency. Further exploration uncovers LRMs' early difficulty awareness accompanied by lower confidence, and shows that problem difficulty is implicitly encoded in hidden states. Code and the extended version of the paper with appendices are available at https://github.com/WYRipple/S1_Bench.

1 Introduction

Recent advances in Large Reasoning Models (LRMs), notably OpenAI's o1/o3 [OpenAI, 2024] and the DeepSeek-R1 [Guo *et al.*, 2025] series, have propelled the development of Large Language Models (LLMs). Unlike traditional LLMs that exhibit intuitive, heuristic *system 1* thinking, LRMs demonstrate deliberate and analytical *system 2* reasoning [Qu *et al.*, 2025; Li *et al.*, 2025c] by explicitly generating external chain-of-thought [Wei *et al.*, 2022] before producing final answers.

While most prior research focuses on enhancing LRMs' reasoning capabilities to achieve superior performance on complex tasks requiring *system 2* [Yang *et al.*, 2025; Li *et al.*, 2025a; Yeo *et al.*, 2025], exploration of their *system 1* thinking abilities remains limited. Specifically, *system 1* thinking refers to the human-like intuitive ability [Kahneman, 2011] to answer questions rapidly with minimal token usage. This capability reflects a model's perception of task difficulty as well as its proficiency in efficiently responding to questions it recognizes as simple [Booch *et al.*, 2021]. In real-world scenarios, where the vast majority of user queries are relatively straightforward for models, ensuring accurate and efficient *system 1* thinking is particularly crucial [Yang *et al.*, 2024].

Benchmark	Cross Domain	Realistic Scenarios	Multi-lingual	Acc.
AIME	✗	✓	✗	6.67
GPQA	✓	✓	✗	24.94
Olympiad	✓	✓	✓	27.94
AMC	✗	✓	✗	31.88
MATH500	✗	✓	✗	58.30
MMLU	✓	✓	✗	66.27
GSM8K	✗	✓	✗	87.45
ASDIV	✗	✓	✗	97.51
GSM8K-zero	✗	✗	✗	77.98
RoR-Bench	✗	✗	✗	14.24
S1-Bench (ours)	✓	✓	✓	**100.00**

Table 1: Characteristics of S1-Bench, with "Acc." representing the average accuracy of four 7-9B LLMs.

In this work, we aim to explore the *system 1* thinking capabilities of LRMs, yet a suitable benchmark remains absent. Some studies attempt to evaluate LRMs using questions that humans consider simple (e.g., GSM8K-zero and RoR [Chiang and Lee, 2024; Yan *et al.*, 2025]), yet these questions remain challenging for models. Other works utilize single-domain benchmarks, such as basic mathematics (GSM8K and ASDIV [Cobbe *et al.*, 2021; Miao *et al.*, 2020]), but lack domain diversity. Other research on LRM efficiency primarily targets challenging problems, such as AIME. As shown in Table 1, the top six benchmarks which are most commonly used in these work remain difficult for small LLMs, making them unsuitable for exploring *system 1* thinking capabilities.

To complement the exploration foundation, we introduce the System 1 Thinking Capability Benchmark (**S1-Bench**), which measures the performance of LRMs across various simple tasks that are commonly encountered in real-world applications. S1-Bench has the following three characteristics: **(1)** ***Simple***. The questions can be easily answered by LLMs. As shown in Table 1, LLMs with 7-9B parameters can robustly provide correct answers across different temperatures. **(2)** ***Diverse***. It encompasses four major categories and 28 subcategories in two languages. **(3)** ***Natural***. The questions are clear, without any misleading elements or ambiguities, ensuring they can be answered intuitively.

We conduct extensive exploration on S1-Bench across

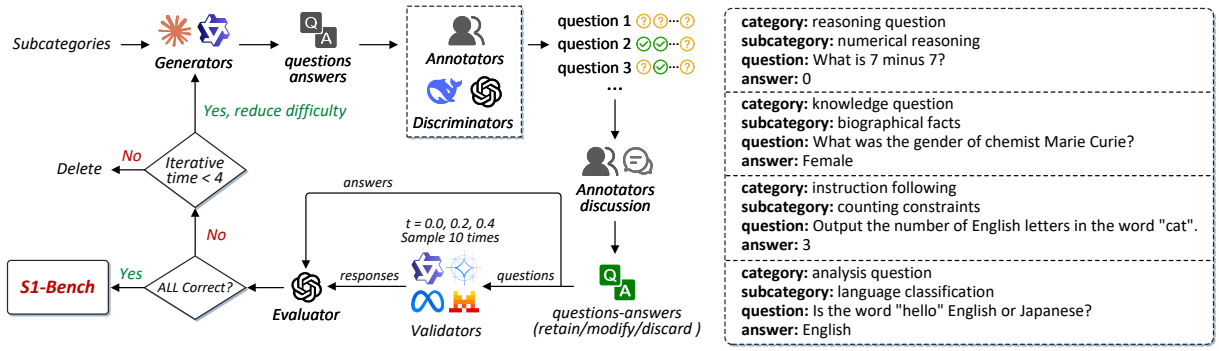

Figure 1: Construction workflow for S1-Bench and an illustrative example from each major category.

28 LRMs, yielding the following key findings: (1) **Under-accuracy**. Despite employing deep reasoning, several LRMs exhibit under-accuracy on simple questions, particularly in knowledge-based and instruction-following tasks, suggesting that long-chain reasoning may introduce catastrophic forgetting. (2) **Inefficiency**: Current LRMs lack *system 1* capabilities across all categories of questions, with average output lengths 13.8× longer than small LLMs on S1-Bench. (3) **Limitations of efficient reasoning methods**: Existing efficient reasoning methods either suffer from limited generalization, failing to effectively compress length on S1-Bench, or fail to balance performance and efficiency, leading to significant drops in accuracy. (4) **Early awareness of problem difficulty**: We observe that LRMs often exhibit explicit difficulty-related comments prior to formal reasoning, indicating that models possess an early awareness of problem difficulty. Moreover, even when identifying a question as simple, LRMs still generate redundant responses with higher average token entropy, indicating stronger exploratory. These reveal a gap between models' difficulty awareness and reasoning behavior. Furthermore, our representation-level analysis shows that question difficulty is implicitly encoded in model hidden states, as a simple single-layer MLP can capture difficulty-related signals.

Our contributions can be summarized as follows:

- We propose S1-Bench, a benchmark dedicated to evaluating LRMs' *system 1* thinking capabilities, and a semi-automated construction workflow for *system 1* questions.
- We reveal LRMs' under-accuracy and inefficiency in *system 1* thinking, and highlight the inadequacy of current efficient reasoning methods, informing future research.
- We explore LRMs' early difficulty awareness, revealing correlations between explicit responses and model confidence, and disentanglement of difficulty in hidden states.

2 S1-Bench

We introduce S1-Bench, a bilingual, multi-domain benchmark designed to evaluate *system 1* thinking capability of LRM on extremely simple questions. These questions are easily solvable by traditional LLMs. S1-Bench, which covers both *English* and *Chinese*, is organized into four major categories: *reasoning* (RSN), *knowledge* (KNO), *instruction following* (IF) and *analysis* (ANA), representing major dimensions commonly employed in LLM capability evaluation [Zheng *et al.*, 2023; Chang *et al.*, 2024]. This section begins with how simplicity is ensured, then the detailed construction workflow for S1-Bench, and concludes with an overview of the dataset statistics. Figure 1 shows the construction workflow and an illustrative example per category.

2.1 How to Ensure Simplicity?

We ensure questions are simple and suitable for *system 1* thinking through the following two aspects.

A Priori Simplicity Constraints

We begin by generating question–answer pairs through collaboration between humans and LLMs. Each pair is required to satisfy both the general and the category-specific simplicity criteria. The general simplicity criteria requires that: (1) Questions must be naturally and clearly expressed, unambiguous, and free of intentional traps. (2) Answers must be unique or easily falsifiable (e.g., providing a three-letter English word).

The category-specific simplicity criteria are as follows. **RSN**: Limited to problems solvable with minimal reasoning or intuition. **KNO**: Restricted to common knowledge with unique, verifiable answers from sources like Wikipedia. **IF**: Involve straightforward instructions without strict formatting requirements. **ANA**: Limited to questions whose answers can be directly inferred from the prompt, such as binary classification. These constraints ensure all questions remain straightforward for human respondents.

A Posteriori Simplicity Verification

Due to the biases existing between language models and humans [Gallegos *et al.*, 2024], questions that are simple for humans may be difficult for LLMs. Therefore, we introduce additional posteriori verification to ensure that questions are simple enough to be correctly and robustly answered by smaller LLMs from different families.

2.2 Construction Workflow

Subcategory Preparation. To ensure diversity, we refer to the subcategories included in existing benchmarks (e.g., MMLU, IFEval, and GSM8K) and evaluation surveys [Chang *et al.*, 2024] to select, merge, or design subcategories for S1-bench, ensuring that each meets the simplicity requirements.

Model	t=0.0	t=0.2	t=0.4	Tokens
Gemma2-9B	100.00	100.00	100.00	38.77
Llama3.1-8B	100.00	100.00	100.00	42.00
Mistral-8B	100.00	100.00	100.00	44.38
Qwen2.5-7B	100.00	100.00	100.00	42.81
DeepSeek-v3	100.00	100.00	100.00	79.53
Llama3.3-70B	100.00	99.76	99.76	53.71
Qwen2.5-14B	99.74	99.76	99.76	40.00
Qwen2.5-32B	99.98	99.98	99.98	43.17
Qwen2.5-72B	100.00	100.00	100.00	44.61
Qwen3-32B (w/o think)	100.00	100.00	100.00	103.30
Qwen3-14B (w/o think)	100.00	100.00	100.00	86.35
Qwen3-8B (w/o think)	100.00	100.00	99.76	90.54
Qwen3-1.7B (w/o think)	98.10	97.16	95.73	114.32

Table 2: Average accuracy (acc@k) and response token count of different LLMs, each sampled 10 times at three temperature settings.

Implementation of A Priori Simplicity. **First,** we use two data *generators*[1] to create 100 initial bilingual question-answer pairs for each candidate subcategory. The data generation prompt explicitly incorporates the subcategory definitions, along with both the general and category-specific simplicity criteria, while also aiming to ensure diversity in the generated questions. **Second,** these question–answer pairs are then independently evaluated by three annotators and two quality *discriminators*[2] according to the general and category-specific simplicity criteria, resulting in five evaluation outcomes per pair. The three annotators are experienced graduate students familiar with LLMs and well-acquainted with the goals of S1-Bench. **Finally,** based on these evaluation outcomes, three annotators discuss and collectively decide whether to retain, modify, or discard each question.

Implementation of A Posteriori Simplicity. **First**, each question obtained from the previous stage is input into the small LLM *validators*[3] with 7˜9 B parameters. For each question, we sample 10 answers at three different temperature settings (0, 0.2, and 0.4), resulting in a total of 30 responses per question. These responses are then individually evaluated for correctness using GPT-4o. **Second**, if all 30 sampled responses are correct, the question is accepted into S1-Bench. Otherwise, the question is returned to the *generators*, where a difficulty-reduction prompt is applied to simplify it. The simplified question then undergoes the same subsequent process. **Finally**, questions that fail to meet the full-accuracy criterion (i.e., 30 out of 30 correct) after three rounds of difficulty reduction are excluded from the workflow.

The final S1-Bench comprises questions validated by both human a priori constraints and LLM a posteriori verification.

2.3 Benchmark Statistics

S1-Bench comprises 422 question-answer pairs across four major categories and 28 subcategories, balanced with 220 English and 202 Chinese questions, with questions averaging 14.46 tokens. To ensure that the a posteriori verification process does not introduce simplicity only tailored to the small LLM validator, we evaluate S1-Bench on five additional LLMs and on Qwen3 Family with reasoning modes disabled. As shown in Table 2, even the 1.7B model achieves over 98% accuracy at temperature 0.

[1]We select Claude-3.7-Sonnet and Qwen2.5-72B-Instruct.

[2]We select GPT-4o and DeepSeek-V3-241226.

[3]Qwen2.5-7B, Llama3.1-8B, Mistral-8B, Gemma2-9B.

Model ID	size	pass@1↑	acc@k↑	f-acc↑	Tokens ↓
Validator LLMs	7-9B	100.00	100.00	–	42.00
close-source LRMs					
claude-sonnet-4	–	**100.00**	**100.00**	**100.00**	166.32
claude-3.7-sonnet	–	99.95	99.76	**100.00**	178.65
gemini-2.5-flash	–	98.48	98.01	**100.00**	309.03
o4-mini	–	99.53	99.53	**100.00**	**129.99**
Hunyuan-T1	–	99.91	99.53	**100.00**	542.31
open-source LRMs					
QwQ-32B	32B	**100.00**	**100.00**	**100.00**	720.10
Qwen3-A22B	235B	99.91	99.76	**100.00**	701.65
Qwen3-A3B	30B	99.95	99.76	**100.00**	638.40
Qwen3-32B	32B	99.91	99.53	99.91	668.69
Qwen3-14B	14B	99.95	99.76	99.95	582.99
Qwen3-8B	8B	99.95	99.76	99.95	657.76
Qwen3-1.7B	1.7B	99.34	97.39	99.81	595.90
DS-R1	671B	**100.00**	**100.00**	**100.00**	646.40
DS-R1-70B	70B	99.38	96.92	99.91	453.81
DS-R1-32B	32B	99.72	98.82	**100.00**	429.91
DS-R1-14B	14B	99.57	97.87	**100.00**	475.46
DS-R1-8B	8B	97.39	97.16	99.53	452.11
DS-R1-7B	7B	95.21	85.78	99.24	454.55
DS-R1-1.5B	1.5B	81.47	**54.50**	97.58	489.54
Nemotron-49B	49B	99.15	97.39	**100.00**	**362.54**
Nemotron-8B	8B	**79.81**	59.00	**84.31**	372.57
L-R1-32B	32B	94.74	79.62	95.07	**1095.36**
L-R1-32B-DS	32B	99.57	98.10	99.81	524.12
L-R1-14B-DS	14B	99.05	95.97	99.19	693.19
L-R1-7B-DS	7B	94.64	83.65	99.67	496.47
s1.1-32B	32B	99.48	98.10	99.53	998.00
s1.1-14B	14B	97.25	91.94	97.39	839.86
s1.1-7B	7B	88.58	63.98	88.96	711.49

Table 3: Accuracy results on S1-Bench, sorted by model family. Bold teal / teal mark the best / second-best; bold burgundy / burgundy mark the worst / second-worst.

3 Experiment Setting

3.1 Baseline Models and Configurations

We evaluated 28 different LRMs, which are explicitly trained to first respond with a ***thinking process***, and then generate a ***final answer***. The evaluated LRMs include both open-source models—such as DeepSeek [Guo *et al.*, 2025], Qwen [Yang *et al.*, 2025], Nemotron [Singhal *et al.*, 2025], Light-R1 [Wen *et al.*, 2025], and s1.1 [Muennighoff *et al.*, 2025]—and closed-source models including claude-sonnet-4-20250514, claude-3-7-sonnet-20250219, gemini-2.5-flash-preview-09-25, o4-mini, and Hunyuan-T1[4], covering parameter scales from 1.5B to 671B. For each model, we adopt a temperature of $t = 0.6$, top-$p = 0.95$, and a sampling size of $k = 5$, with the maximum generation length set to 10,000 tokens.

3.2 Evaluation Metrics

Format Metrics. We measure format accuracy (***f-acc***) as the proportion of responses that follow the required output format, averaged over five runs. A response is considered valid if an end-of-thinking marker (e.g., `</think>`) correctly separates the reasoning process from a non-empty final answer.

[4]https://llm.hunyuan.tencent.com

Model	pass@1	RSN	KNO	IF	ANA
DS-R1-1.5B	81.5	86.3	75.2	76.8	84.4
Nemotron-8B	79.8	84.8	71.3	76.2	83.7
s1.1-7B	88.6	93.1	83.7	82.9	91.4
Average	83.3	88.1	76.7	78.6	86.5

Table 4: Pass@1 by question type on S1-Bench (models with pass@1 $< 90\%$ only).

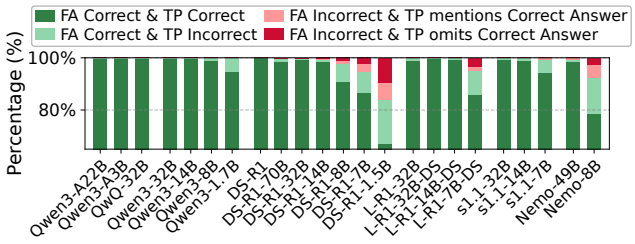

Figure 2: Distribution of Thinking Processes (TP) across four categories. Green / red bars: correct / incorrect Final Answers (FA).

Efficiency Metrics. We report the average response length in tokens (***Tokens***), excluding samples with endless thinking, with all token counts computed using the Qwen2.5 tokenizer to ensure fair comparison.

Accuracy Metrics. We calculate accuracy metrics only for responses with correct formatting. Following the LLM-as-judge setting [Zheng *et al.*, 2023], we use GPT-4o as the evaluator to assess the ***final answer*** part. We use two metrics: ***pass@1*** and ***acc@k***. ***pass@1*** follows the definition in DeepSeek-R1 [Guo *et al.*, 2025], computing the average accuracy over k samples, while ***acc@k*** is the percentage of questions where all k answers are correct. Both metrics use k=5. Notably, f-acc represents the upper bound for pass@1 and acc@k under the formatting requirement.

4 Main Experiment

In this section, we conduct a comprehensive evaluation of LRMs on S1-Bench from three perspectives: accuracy (Table 3), efficiency (Table 5), and the impact of RL-based efficient reasoning algorithms (Table 6).

4.1 Accuracy Analysis

We analyze the accuracy-related results, and as shown in Table 3, we obtain the following observations:

Although the vast majority of larger-scale or closed-source models can maintain the same level of accuracy as small LLM-based validators (often exceeding 99% accuracy), some smaller LRMs suffer from severe performance degradation. For example, DS-R1-1.5B and Nemotron-8B achieve acc@k values only slightly above 50%. This behavior directly contradicts the original goal of LRMs: scaling thinking length can actually undermine accuracy on System 1 problems. Moreover, many LRMs exhibit limited robustness, with pronounced gaps between pass@1 and acc@k consistently observed across models, including DS-R1-1.5B (↓26.97%), L-R1-32B (↓15.12%), DS-R1-7B (↓9.43%), L-R1-7B-DS (↓10.99%), Nemotron-8B (↓20.81%), and s1.1-7B (↓24.60%). Finally, certain models suffer from output format misalignment, as evidenced by format correctness below 90% for Nemotron-8B and s1.1-7B.

Accuracy degradation is particularly pronounced on instruction-following and knowledge-based questions. We conduct an question type analysis on models whose pass@1 falls below 90%. As shown in Table 4, a clear performance gap emerges across categories: instruction-following and knowledge questions reach average accuracies of only 76.7% and 78.6%, respectively, versus 88.1% on reasoning problems. This pattern suggests that long-chain reasoning does not consistently improve model behavior and may instead weaken instruction-following ability and factual reliability.

We further analyze the error cases and types in the responses. Specifically, we use DeepSeek-v3 to categorize LRM responses into four cases and compute their proportions. (1) Final answer correct; thinking process entirely accurate. (2) Final answer correct; thinking process contains intermediate errors. (3) Final answer incorrect; correct answer mentioned in thinking process. (4) Final answer incorrect; correct answer never mentioned in thinking process. Results are shown in Figure 2. Key findings include: **(1) Lower-accuracy LRMs tend to produce less reliable reasoning chains; even when they arrive at the correct final answer, their intermediate steps often contain errors (light green).** LRMs with high accuracy (e.g., DS-R1) show almost no flawed reasoning steps, whereas those with lower accuracy (e.g., DS-R1-1.5B) often generate incorrect intermediate conclusions, further indicating that they lack robust reasoning ability. **(2) Although LRMs sometimes mention the correct answer during reasoning, they may deviate and ultimately produce incorrect final answers (light red).** For example, in one case, the LRM initially arrived at the correct answer but undermined it through excessive verification. In another case, the LRM directly denied the correct answer that appeared during reasoning.

4.2 Efficiency Analysis

As shown in Table 3, we present the average response tokens for each problem category in S1-Bench sampled five times (2.11k sample points per model). We observe that: **(1) all LRMs exhibit overthinking on System 1 problems**. While closed-source LRMs show higher efficiency, they still consume over three times more tokens. Open-source LRMs are considerably less efficient, with s1.1 and Light-R1 family models exhibiting severe overthinking to achieve better performance on complex problems. **(2) LRMs within the same family show similar efficiency without demonstrating clear scaling phenomena**. Across DS, Qwen3, and Nemotron, response lengths cluster by model family rather than by size, showing no clear correlation between model scale and thinking length. This suggests that training methods and data composition may be more critical factors in determining System 1 thinking effective.

Analysis Across Question Types

We further present the average response tokens by fine-grained categories and languages, as shown in Table 5. The main find-

Model ID	size	S1-Bench-EN					S1-Bench-ZH					Avg
		RSN	KNO	IF	ANA	**Avg**	RSN	KNO	IF	ANA	**Avg**	
Gemma2-9B	9B	74.8	29.4	5.3	52.4	45.9	51.6	19.8	7.5	35.1	31.0	38.8
Llama3.1-8B	8B	91.0	35.4	12.4	61.9	56.0	44.0	28.3	15.2	18.7	26.7	42.0
Qwen2.5-7B	7B	65.5	46.3	6.4	49.6	46.5	50.5	46.6	9.8	36.9	38.8	42.8
Mistral-8B	8B	67.2	55.5	8.6	50.1	49.6	47.3	56.1	14.8	29.7	38.7	44.4
Column Avg	-	74.6	41.6	8.2	53.5	49.5	48.3	37.7	11.8	30.1	33.8	42.0
o4-mini	–	**180.5**	**114.2**	**63.1**	**181.1**	**147.2**	**148.1**	**105.9**	**33.5**	122.0	**111.2**	**130.0**
claude-sonnet-4	–	188.5	127.5	66.4	225.9	168.2	180.5	149.5	65.6	204.0	164.3	166.3
claude-3.7-sonnet	–	190.8	161.6	89.8	222.4	179.2	180.0	166.9	100.3	216.3	178.1	178.7
gemini-2.5-flash	–	295.1	141.8	122.2	398.6	268.1	383.0	220.0	271.3	465.3	353.6	309.0
Nemotron-49B	49B	599.7	587.6	396.5	526.1	540.4	232.9	157.3	235.5	**107.8**	168.8	362.5
Nemotron-8B	8B	561.0	585.1	458.0	303.1	462.6	369.5	326.0	288.1	166.7	273.5	372.6
DS-R1-32B	32B	421.8	504.4	414.7	521.1	473.7	362.2	385.6	343.1	408.8	382.2	429.9
DS-R1-8B	8B	472.2	528.9	530.7	462.7	491.2	521.9	404.4	266.2	395.5	409.4	452.1
DS-R1-70B	70B	464.1	501.3	378.5	536.1	484.0	450.8	450.2	328.4	416.7	420.9	453.8
DS-R1-7B	7B	447.5	623.9	353.8	510.0	495.5	446.5	463.2	339.5	373.0	409.4	454.5
DS-R1-14B	14B	503.7	674.7	367.3	494.2	519.0	452.0	465.4	375.3	405.8	428.0	475.5
DS-R1-1.5B	1.5B	480.8	584.7	417.4	577.2	529.1	493.0	497.4	329.8	423.1	446.0	489.5
L-R1-7B-DS	7B	568.1	667.1	501.7	566.3	580.3	444.8	454.6	344.1	366.4	405.0	496.5
L-R1-32B-DS	32B	574.5	706.6	647.6	632.8	636.3	431.2	367.0	377.1	418.7	402.2	524.1
Hunyuan-T1	–	561.6	693.8	380.9	435.0	521.2	676.8	553.8	505.1	523.8	565.3	542.3
Qwen3-14B	14B	700.4	639.5	286.2	575.0	579.8	730.4	557.2	403.1	586.0	586.5	583.0
Qwen3-1.7B	1.7B	790.4	720.6	399.9	526.2	624.6	689.8	563.6	406.4	545.9	564.7	595.9
Qwen3-A3B	30B	745.0	729.3	328.1	594.8	625.7	773.7	655.8	453.7	648.6	652.2	638.4
DS-R1	671B	786.1	723.8	711.4	529.2	672.5	727.3	638.5	607.9	533.9	617.9	646.4
Qwen3-8B	8B	853.7	753.1	394.4	629.5	683.2	749.2	623.8	459.3	624.0	630.0	657.8
Qwen3-32B	32B	805.7	774.2	356.9	645.5	674.7	780.2	695.2	446.6	645.3	662.1	668.7
L-R1-14B-DS	14B	951.0	1026.0	829.8	653.5	848.2	594.7	610.1	442.2	451.7	525.7	693.2
Qwen3-A22B	235B	925.3	864.3	487.2	605.7	734.5	803.3	713.4	487.2	611.3	665.9	701.7
s1.1-7B	7B	1039.5	840.8	1923.2	529.4	929.9	489.6	351.3	1034.3	332.4	475.6	711.5
QwQ-32B	32B	873.3	808.1	520.8	634.7	722.4	866.9	707.3	613.3	**667.7**	717.6	720.1
s1.1-14B	14B	871.8	746.2	**2233.1**	708.1	960.2	654.6	546.0	1512.6	579.7	710.7	839.9
s1.1-32B	32B	1077.9	889.7	2055.4	781.7	1081.7	995.6	**765.2**	**1634.6**	666.5	**906.5**	998.0
L-R1-32B	32B	**1614.0**	**1217.3**	1996.9	**930.1**	**1338.3**	**1035.6**	737.7	1240.7	610.2	835.3	**1095.4**
Column Avg	-	713.8	665.2	673.1	538.0	634.0	620.1	484.2	588.3	445.2	516.7	577.6
Improvement	-	×**9.6**	×**16.0**	×**82.1**	×**10.1**	×**12.8**	×**12.8**	×**12.8**	×**49.9**	×**14.8**	×**15.3**	×**13.8**

Table 5: Average response tokens under sampling decoding on S1-Bench. Bold teal / teal mark the best / second-best; bold burgundy / burgundy mark the worst / second-worst. Bold denotes the maximum Improvement per language.

Model ID	S1-Bench			MATH500			AIME 24		
	Tok.	p@1	RNT	Tok.	p@1	RNT	Tok.	p@1	RNT
1.5B Model									
DS-R1-1.5B	489	81.5	0.0	5534	82.1	0.0	12339	28.1	0.0
Shorterbetter-1.5B	143	86.7	0.0	1131	74.8	0.0	4328	18.9	0.0
Laser-D-1.5B	353	85.6	0.0	1872	84.2	0.0	5750	34.2	0.0
TLMRE-1.5B	185	87.2	0.0	2376	84.9	0.0	9459	31.6	0.0
Adaptthink-1.5B	365	84.8	23.8	1782	82.0	76.8	6670	31.0	40.4
Autothink-1.5B	111	60.1	91.4	2195	84.0	71.1	9514	34.6	6.4
7B Model									
DS-R1-7B	454	95.2	0.0	3593	92.0	0.0	10490	52.9	0.0
Shorterbetter-7B	163	97.2	0.0	1210	86.6	0.0	5288	53.3	0.0
Laser-D-7B	351	97.3	0.0	1836	92.2	0.0	5379	58.3	0.0
TLMRE-7B	270	97.4	0.0	2073	91.8	0.0	9410	52.3	0.0
Adaptthink-7B	317	97.0	26.1	1875	92.0	76.6	8051	55.6	6.3
Autothink-7B	66	91.8	93.3	2146	91.2	12.0	8599	54.8	3.0

Table 6: Results of RL-based methods for think-token reduction. p@1 denotes pass@1 and RNT the ratio of no-thinking responses.

ings are as follows: **(1) LRMs exhibit a substantial increase in response length across all four major categories, 28 subcategories, and two languages.** As shown in Table 5, for each of the four major categories, the average response length of LRMs exceeds that of LLMs by more than a factor of ten. Response lengths also increase significantly across all subcategories. This suggests that while LRMs are primarily trained on reasoning data to produce long CoT style responses, this stylistic pattern generalizes well across a wide range of question types. **(2) LRMs exhibit the most significant increase in tokens for instruction following questions and tend to over-explore when the solution space is vast.** As shown in Table 5, although small LLMs provide the most concise responses to instruction following questions, LRMs generate dramatically longer outputs—82.1 × longer in English and 49.9 × longer in Chinese than small LLMs. To investigate the cause, we further analyze the subcategories of instruction following questions. Average tokens is notably longer in the subcategories of *length constraints*, *character constraints*, and *sentence constraints*. These three question types share a similar characteristic: their correctness is easy to verify, but the solution space is vast. We find that, although the model quickly identifies a correct answer, it becomes trapped in the search space, continually exploring alternatives and failing to stop in time.

Thinking Solution Analysis

To better understand the causes of inefficiency in LRMs on S1-Bench, we analyze the solution rounds of their thinking processes. We first use DeepSeek-v3 to segment each thinking process into several solutions, each defined as a point at which LRMs explicitly arrives at a conclusion that matches the correct answer. We then compute the average token counts in the first solution. Our analysis reveals the following: **(1) The**

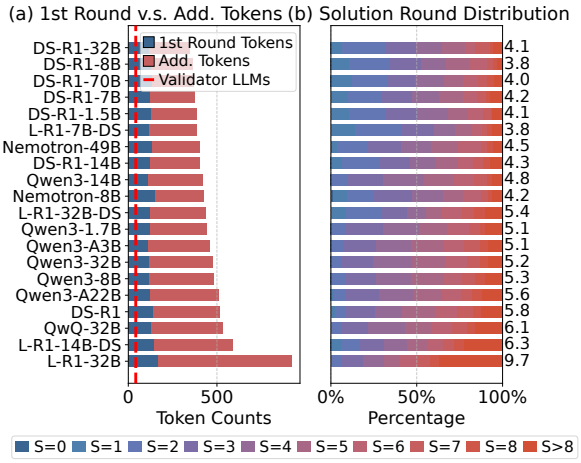

Figure 3: (a) Comparison of first round and additional token costs for each LRM. (b) Distribution of solution rounds for each LRM.

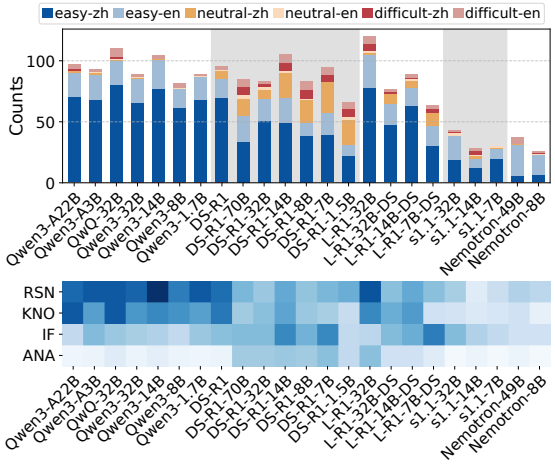

Figure 4: Top: Count of early difficulty awareness across models. Bottom: Probability of early difficulty awareness by question type.

token consumed in the first solution of LRMs significantly exceeds that of validator LLMs. As shown in Figure 3 (a), this suggests that LRMs may involve unnecessary reasoning steps in first solution, which could be one of the reasons for their inefficiency. **(2) The primary reason for efficiency gaps between LRMs lies in the number of redundant solution rounds they generate, rather than the token cost in the initial round.** As shown in Figure 3 (a), although total thinking token counts vary widely across LRMs, their token counts in the initial round are similar and only account for a small fraction of the total. Figure 3 (b) further shows the distribution of solution rounds on S1-Bench, revealing that LRMs with longer thinking processes tend to generate more solution round, and this redundancy greatly increases computational cost. Furthermore, further experiments reveal that the redundancy in the reasoning process gradually increases over time. Furthermore, the similarity across solution rounds progressively increases as reasoning proceeds, suggesting that later rounds contribute increasingly redundant information.

4.3 Efficient Reasoning Algorithms Analysis

Recently, several methods have been proposed to mitigate overthinking in LRMs based on reinforcement learning (RL). We evaluate representative approaches on S1-Bench, including (1) RL with length-based penalties (e.g., Shorterbetter [Yi *et al.*, 2025], Laser-D [Liu *et al.*, 2025], and TLMRE [Arora and Zanette, 2026]), and (2) adaptive mode selection (e.g., Adaptthink [Zhang *et al.*, 2025a] and Autothink [Tu *et al.*, 2025]). Table 6 reports results for models fine-tuned from DS-R1-1.5B/7B, leading to the following observations. **(1) Token reduction on challenging benchmarks does not necessarily translate to similar gains on S1-Bench.** For instance, Laser-D-1.5B and AdaptThink-1.5B reduce reasoning tokens by nearly 50% on AIME2024, but achieve only about 25% reduction on S1-Bench, indicating limited generalization of these methods to out-of-distribution problems. **(2) Some methods, such as ShorterBetter and AutoThink, achieve substantial compression on S1-Bench at the cost of severe accuracy degradation.** Specifically, ShorterBetter-1.5B attains only 18.9% pass@1 on AIME24, while AutoThink-1.5B achieves merely 60.1% accuracy on S1-Bench, suggesting difficulty in balancing reasoning efficiency with answer correctness.

5 Early Difficulty Awareness in LRMs

We observe that on S1-Bench, LRMs may form an early intuitive judgment of question difficulty before initiating formal reasoning. In some cases, models begin their responses with statements such as "This looks like a simple problem," prior to any substantive reasoning. We refer to this phenomenon as **early awareness of question difficulty**. To better understand this behavior, we conduct the following analyses.

Exploring Frequency and Stylistic Patterns of Difficulty Awareness. We employ GPT-4o to classify the initial portion of model responses—defined as the content before the first "\n\n"—based on whether and how the model comments on question difficulty. Each response is categorized into one of four types: easy, neutral, difficult, or no comment. Figure 4 presents the distribution and corresponding probabilities of these categories across four types of questions. From these results, we make several observations. First, all LRMs exhibit early difficulty awareness to varying degrees, with the phenomenon being particularly prominent in the Qwen, DeepSeek, and Light-R1 model families. Second, LRMs display clear stylistic differences in expressing such awareness. For instance, the Qwen family tends to characterize questions in S1-Bench as easy, whereas DeepSeek-distilled models produce more diverse difficulty-related comments. Third, this early difficulty awareness is most evident in reasoning-oriented questions, while it appears less frequently in analytical questions.

Exploring the Relationship Between Early Difficulty Awareness, Reasoning Length, and Entropy. To investigate whether the early sense of a question as "easy" leads to a corresponding reduction in response length, we compare the average response tokens for questions in the easy category versus all samples. The results are shown in Figure 5. Except for L-R1-32B, the remaining LRMs do not exhibit a noticeable

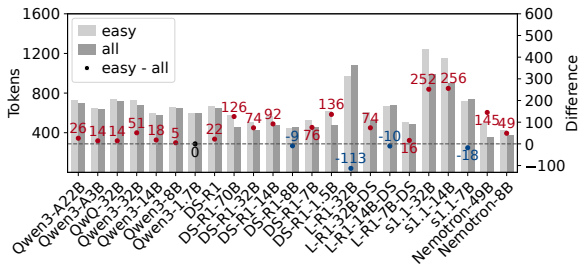

Figure 5: Average response tokens in the easy category vs. all samples. Dots show difference: easy minus all.

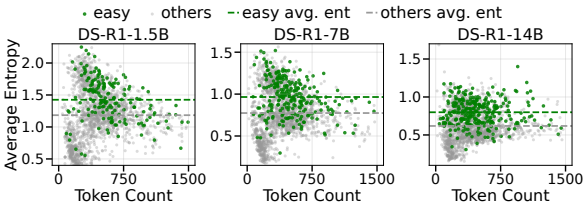

Figure 6: Average response entropy of easy and other responses.

decrease in response length when questions are perceived as easy. On the contrary, 18 out of 23 LRMs generate longer responses under this condition. We further analyze the average token entropy of responses associated with easy judgments, with the results presented in Figure 6. Interestingly, these responses exhibit higher average entropy, indicating stronger exploratory behavior during generation. One possible explanation is that models may exhibit self-doubt toward their own early difficulty judgments, leading them to engage in increased exploration and produce longer responses, despite initially perceiving the question as simple. **This suggests a discrepancy between the LRM's initial sense of difficulty and its generative behavior**, the causes and improvements of which warrant further investigation.

Exploring Activation Pattern Differences Across Question Difficulty. In contrast to the preceding analyses based on explicit difficulty-related cues in model outputs, we further examine whether question difficulty is reflected at the representation level through systematic differences in model activation patterns. Specifically, using DS-R1-1.5B and DS-R1-7B as representative examples, we analyze whether the final-layer representation of the last token in the encoded question varies systematically with question difficulty. To this end, we conduct a probing-based analysis by training a single-layer MLP as a weak linear classifier with limited training data, which serves solely to assess whether hidden representations contain information predictive of question difficulty. The training set is constructed by selecting one Chinese and one English example from each subcategory of S1-Bench as easy questions, and sampling an equal number of bilingual questions from AIME25 as hard questions, where Chinese examples are obtained via translation from English. We then evaluate the trained probe on the MATH500 dataset, which naturally contains five difficulty levels. For each level, we compute the proportion of questions classified as difficult. As shown in Table 7, this proportion increases consistently with question difficulty, indicating that model representations implicitly encode structured information aligned with problem difficulty.

Model	Level 1	Level 2	Level 3	Level 4	Level 5
DS-R1-1.5B	46.5	56.7	56.7	79.7	92.5
DS-R1-7B	27.9	43.3	60.9	81.3	91.1

Table 7: Proportion (%) of questions classified as difficult across five difficulty levels on the MATH500 dataset.

6 Related Work

6.1 Large Reasoning Models

Large Reasoning Models (LRMs), characterized by explicitly generating external thinking processes before final answers [Kumar *et al.*, 2025b; Li *et al.*, 2025b], achieve a paradigm shift from intuitive *system 1* thinking to deliberative *system 2* reasoning compared to traditional LLMs [Li *et al.*, 2025c; Qu *et al.*, 2025], thus achieving superior performance on complex tasks. The development of recent LRMs has largely followed two main approaches: large-scale reinforcement learning (RL) and model distillation. Models trained via RL [Ai *et al.*, 2026; Team *et al.*, 2025; Nie *et al.*, 2026] leverage reward-based optimization to gradually incentivize deliberative reasoning. In contrast, distillation-based LRMs [Min *et al.*, 2024; Ye *et al.*, 2025; Muennighoff *et al.*, 2025; Zhang *et al.*, 2025b] acquire such abilities by transferring structured reasoning patterns from advanced teacher models.

6.2 Limitations of LRMs

While LRMs have shown significant performance gains through deliberate reasoning, rigid adherence to this overly cautious thinking can introduce new limitations. On one hand, intermediate reasoning steps can cause excessive token generation and unnecessary solving attempts [Chen *et al.*, 2024; Kumar *et al.*, 2025a; Gong *et al.*, 2026], even leading to redundancy in the hidden layers [Chen *et al.*, 2025]. On the other hand, LRMs' performance can drop in specific contexts like safety scenarios [Jiang *et al.*, 2025], agents [Zhang *et al.*, 2026] and role-playing [Feng *et al.*, 2025]. However, prior studies mainly evaluated LRMs on complex tasks that more suited for deliberative *system 2* thinking. Our work examines how deliberative reasoning impacts extremely simple problems better matched to intuition-driven *system 1* processing.

7 Conclusion

This paper extensively explores the *system 1* thinking capabilities of LRMs. We construct S1-Bench, a simple, domain-diverse, and natural benchmark for evaluating *system 1* thinking abilities. Our exploration reveals two fundamental issues of LRMs on *system 1* problems, namely under-accuracy and over-reasoning, and exposes limitations of existing efficient training methods. We also demonstrate models' inherent perception of *system 1* problems from both linguistic and representational perspectives, providing insights for future research.

Acknowledgments

We would like to thank the anonymous reviewers, the area chairs and program chairs for their valuable comments and efforts. This work is supported by the National Natural Science Foundation of China (Grant No. 62572465).

Contribution Statement

Equal contribution: ***Wenyuan Zhang*** and ***Shuaiyi Nie***. Corresponding author: ***Tingwen Liu***.

References

[Ai *et al.*, 2026] Zhengyang Ai, Zikang Shan, Xiaodong Ai, Jingxian Tang, Hangkai Hu, and Pinyan Lu. SHAPE: Stage-aware hierarchical advantage via potential estimation for llm reasoning. *arXiv preprint arXiv:2604.06636*, 2026.

[Arora and Zanette, 2026] Daman Arora and Andrea Zanette. Training language models to reason efficiently. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

[Booch *et al.*, 2021] Grady Booch, Francesco Fabiano, Lior Horesh, Kiran Kate, Jonathan Lenchner, Nick Linck, Andreas Loreggia, Keerthiram Murgesan, Nicholas Mattei, et al. Thinking fast and slow in ai. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

[Chang *et al.*, 2024] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 2024.

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

[Chen *et al.*, 2025] Yilong Chen, Junyuan Shang, Zhenyu Zhang, Yanxi Xie, Jiawei Sheng, Tingwen Liu, Shuohuan Wang, Yu Sun, Hua Wu, and Haifeng Wang. Inner thinking transformer: Leveraging dynamic depth scaling to foster adaptive internal thinking. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025.

[Chiang and Lee, 2024] Cheng-Han Chiang and Hung-yi Lee. Over-reasoning and redundant calculation of large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2024.

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

[Feng *et al.*, 2025] Xiachong Feng, Longxu Dou, and Lingpeng Kong. Reasoning does not necessarily improve role-playing ability. *arXiv preprint arXiv:2502.16940*, 2025.

[Gallegos *et al.*, 2024] Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K Ahmed. Bias and fairness in large language models: A survey. *Computational Linguistics*, 2024.

[Gong *et al.*, 2026] Haisong Gong, Jing Li, Junfei Wu, Qiang Liu, and Shu Wu. Strive: Structured reasoning for self-improvement in claim verification. *Machine Intelligence Research*, 2026.

[Guo *et al.*, 2025] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 2025.

[Jiang *et al.*, 2025] Fengqing Jiang, Zhangchen Xu, Yuetai Li, Luyao Niu, Zhen Xiang, Bo Li, Bill Yuchen Lin, and Radha Poovendran. SafeChain: Safety of language models with long chain-of-thought reasoning capabilities. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025.

[Kahneman, 2011] Daniel Kahneman. *Thinking, fast and slow*. macmillan, 2011.

[Kumar *et al.*, 2025a] Abhinav Kumar, Jaechul Roh, Ali Naseh, Marzena Karpinska, Mohit Iyyer, Amir Houmansadr, and Eugene Bagdasarian. Overthink: Slowdown attacks on reasoning llms. *arXiv preprint arXiv:2502.02542*, 2025.

[Kumar *et al.*, 2025b] Komal Kumar, Tajamul Ashraf, Omkar Thawakar, Rao Muhammad Anwer, Hisham Cholakkal, Mubarak Shah, Ming-Hsuan Yang, Phillip HS Torr, Salman Khan, and Fahad Shahbaz Khan. Llm post-training: A deep dive into reasoning large language models. *arXiv preprint arXiv:2502.21321*, 2025.

[Li *et al.*, 2025a] Dacheng Li, Shiyi Cao, Tyler Griggs, Shu Liu, Xiangxi Mo, Eric Tang, Sumanth Hegde, Kourosh Hakhamaneshi, Shishir G Patil, Matei Zaharia, et al. Llms can easily learn to reason from demonstrations structure, not content, is what matters! *arXiv preprint arXiv:2502.07374*, 2025.

[Li *et al.*, 2025b] Xiyun Li, Tielin Zhang, Chenghao Liu, Shuang Xu, and Bo Xu. Theory of mind inspired large reasoning language model improved multi-agent reinforcement learning algorithm for robust and adaptive partner modelling. *Machine Intelligence Research*, 2025.

[Li *et al.*, 2025c] Zhong-Zhi Li, Duzhen Zhang, Ming-Liang Zhang, Jiaxin Zhang, Zengyan Liu, Yuxuan Yao, Haotian Xu, Junhao Zheng, Pei-Jie Wang, Xiuyi Chen, et al. From system 1 to system 2: A survey of reasoning large language models. *arXiv preprint arXiv:2502.17419*, 2025.

[Liu *et al.*, 2025] Wei Liu, Ruochen Zhou, Yiyun Deng, Yuzhen Huang, Junteng Liu, Yuntian Deng, Yizhe Zhang, and Junxian He. Learn to reason efficiently with adaptive length-based reward shaping. *arXiv preprint arXiv:2505.15612*, 2025.

[Miao *et al.*, 2020] Shen-yun Miao, Chao-Chun Liang, and Keh-Yih Su. A diverse corpus for evaluating and developing English math word problem solvers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

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

[Muennighoff *et al.*, 2025] Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025.

[Nie *et al.*, 2026] Shuaiyi Nie, Siyu Ding, Wenyuan Zhang, Linhao Yu, Tianmeng Yang, Yao Chen, Tingwen Liu, Weichong Yin, Yu Sun, and Hua Wu. Attnpo: Attention-guided process supervision for efficient reasoning. *arXiv preprint arXiv:2602.09953*, 2026.

[OpenAI, 2024] OpenAI. Learning to reason with LLMs. https://openai.com/index/learning-to-reason-with-llms/, September 2024.

[Qu *et al.*, 2025] Xiaoye Qu, Yafu Li, Zhaochen Su, Weigao Sun, Jianhao Yan, Dongrui Liu, Ganqu Cui, Daizong Liu, Shuxian Liang, et al. A survey of efficient reasoning for large reasoning models: Language, multimodality, and beyond. *arXiv preprint arXiv:2503.21614*, 2025.

[Singhal *et al.*, 2025] Soumye Singhal, Jiaqi Zeng, Alexander Bukharin, Yian Zhang, Gerald Shen, Ameya Sunil Mahabaleshwarkar, Bilal Kartal, Yoshi Suhara, Akhiad Bercovich, Itay Levy, et al. Llama-nemotron: Efficient reasoning models. In *The Exploration in AI Today Workshop at ICML 2025*, 2025.

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

[Tu *et al.*, 2025] Songjun Tu, Jiahao Lin, Qichao Zhang, Xiangyu Tian, Linjing Li, Xiangyuan Lan, and Dongbin Zhao. Learning when to think: Shaping adaptive reasoning in r1-style models via multi-stage rl. In *Advances in Neural Information Processing Systems*, 2025.

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

[Wen *et al.*, 2025] Liang Wen, Yunke Cai, Fenrui Xiao, Xin He, Qi An, Zhenyu Duan, Yimin Du, Junchen Liu, Lifu Tang, Xiaowei Lv, Haosheng Zou, Yongchao Deng, Shousheng Jia, and Xiangzheng Zhang. Light-R1: Curriculum SFT, DPO and RL for long COT from scratch and beyond. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, 2025.

[Yan *et al.*, 2025] Kai Yan, Yufei Xu, Zhengyin Du, Xuesong Yao, Zheyu Wang, Xiaowen Guo, and Jiecao Chen. Recitation over reasoning: How cutting-edge language models can fail on elementary school-level reasoning problems? In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, 2025.

[Yang *et al.*, 2024] Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Shaochen Zhong, Bing Yin, and Xia Hu. Harnessing the power of llms in practice: A survey on chatgpt and beyond. *ACM Transactions on Knowledge Discovery from Data*, 2024.

[Yang *et al.*, 2025] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[Ye *et al.*, 2025] Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, et al. LIMO: Less is more for reasoning. In *Second Conference on Language Modeling*, 2025.

[Yeo *et al.*, 2025] Edward Yeo, Yuxuan Tong, Xinyao Niu, Graham Neubig, and Xiang Yue. Demystifying long chain-of-thought reasoning in LLMs. In *ICLR 2025 Workshop on Deep Generative Model in Machine Learning: Theory, Principle and Efficacy*, 2025.

[Yi *et al.*, 2025] Jingyang Yi, Jiazheng Wang, and Sida Li. Shorterbetter: Guiding reasoning models to find optimal inference length for efficient reasoning. In *Advances in Neural Information Processing Systems*, 2025.

[Zhang *et al.*, 2025a] Jiajie Zhang, Nianyi Lin, Lei Hou, Ling Feng, and Juanzi Li. AdaptThink: Reasoning models can learn when to think. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025.

[Zhang *et al.*, 2025b] Wenyuan Zhang, Tianyun Liu, Mengxiao Song, Xiaodong Li, and Tingwen Liu. SOTOPIA-Ω: Dynamic strategy injection learning and social instruction following evaluation for social agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025.

[Zhang *et al.*, 2026] Wenyuan Zhang, Xinghua Zhang, Haiyang Yu, Shuaiyi Nie, Bingli Wu, Juwei Yue, Tingwen Liu, and Yongbin Li. ExpSeek: Self-triggered experience seeking for web agents. *arXiv preprint arXiv:2601.08605*, 2026.

[Zheng *et al.*, 2023] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.