Benchmarking and Enhancing Rule Knowledge-Driven Reasoning of Large Language Models

Zijie Xu[1], Wenjun Ke[1, 2], Peng Wang[1, 2*], Guozheng Li[1], Qingjian Ni[1*], Jiajun Liu[1], Ziyu Shang[1], Jing Zhou[1]

[1]School of Computer Science and Engineering, Southeast University
[2]Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications (Southeast University), Ministry of Education, China
{zijiexu, kewenjun, pwang, gzli, nqj, jiajliu, ziyus1999, zhoujing0201}@seu.edu.cn

Abstract

Large Language Models (LLMs) have demonstrated strong capabilities across diverse tasks under the **example-driven learning paradigm**. However, in high-stakes domains such as emergency response and industrial safety, historical incidents are scarce, confidential, or both, while concise *rule books* are abundant. We formalize this underexplored setting as **rule knowledge–driven reasoning** and ask: *Can LLMs reason reliably when rules are plentiful but examples are nearly absent?* To study this question, we introduce **RULER**, an automatic benchmark that generates 32K rigorously verified questions from 1K expert-curated emergency response rules to probe three core abilities: *rule memorization*, *single-rule application*, and *multi-rule complex reasoning*. RULER is further equipped with a hallucination-aware evaluation suite and novel relational metrics. A comprehensive empirical study of five representative LLMs and five enhancement strategies shows that, even when models achieve reliable performance on rule memorization and single-rule application, multi-rule complex reasoning plateaus at 5.4 on a 10-point scale. To address this limitation, we propose **RAMPS**, a **R**ule knowledge-**A**ware **M**onte Carlo Tree Search **P**rocess-reward **S**upervision framework. RAMPS injects rule knowledge priors into MCTS, distills 12K step-level traces without human annotation, and trains an advantage-based reward model that scores candidate reasoning paths during beam search inference. Experimental results show that RAMPS significantly improves multi-rule complex reasoning performance to 7.7.

Introduction

Large language models (LLMs) have achieved impressive success across various domains, including medicine, law, mathematics, and code generation (Peng et al. 2023; Cui et al. 2023; Uesato et al. 2023; Tong and Zhang 2024). These advances are largely driven by scaling both model parameters and training data under the *example-driven learning paradigm* (Brown et al. 2020; Yang et al. 2025b). However, in high-stakes domains such as emergency response and industrial safety, real-world examples are inherently scarce and often confidential due to privacy and security constraints. This scarcity is not an incidental issue but a fundamental characteristic of these domains. Nevertheless, these fields are not knowledge deserts: human experts have established comprehensive rule knowledge systems that serve as actionable guidance. Motivated by human learning strategies—first mastering expert-curated knowledge, then validating and testing their abilities to solve practical problems through targeted examples—we explore a critical question: **Can LLMs deliver reliable, trustworthy reasoning in new domains solely by leveraging expert rule knowledge, without explicit in-domain examples for training?**

We formalize this question as a novel task: *rule knowledge-driven reasoning*, in which LLMs are given domain-specific expert rules but no training examples and must generalize to unseen reasoning tasks. This setting complements the prevailing *example-driven learning paradigm*. Advancing research in this direction requires dedicated evaluation frameworks. However, existing knowledge-based reasoning evaluations (Yu et al. 2024; Sun et al. 2024; Li, Wang, and Ke 2023) have three critical limitations: (1) coarse metrics that do not disentangle underlying reasoning abilities; (2) inadequate quantification of hallucination behavior; and (3) reliance on extensive human annotation or large-scale corpora, which is misaligned with example-scarce domains.

To this end, we introduce **RULER**, an automatic, large-scale benchmark for **RULE** knowledge-driven open-ended **R**easoning. Starting from 1K expert-curated emergency response rules, a question generator (o3 (OpenAI 2025)) produces draft questions, and an independent verifier (QwQ-32B (QwenLM 2025)) challenges them via multi-turn cross-verification (Cohen et al. 2023). This pipeline yields 32K high-quality, open-ended questions without human annotation. We further normalize rules into triplets $\{Subject, State, Action\}$ for fine-grained assessment. For example, *The occurrence of a major earthquake disaster* $(State)$ $\Rightarrow$ *The provincial government activates a Level-II earthquake emergency response*$(Action)$. Leveraging this representation, we construct three complementary question sets that decompose the LLMs' reasoning process into three core abilities: **rule memorization**, **single-rule application**, and **multi-rule complex reasoning**.

To address the limitations in previous work (Shang et al. 2024, 2025; Liu et al. 2024b; Li et al. 2025), we develop a *hallucination-aware* evaluation suite that assesses both quality and trustworthiness. First, we disentangle two types of hallucination: **intrinsic hallucination**, where an answer

*Corresponding authors.

contradicts explicit information in the question content (e.g., altering casualty numbers or misjudging the stated response levels); and **extrinsic hallucination**, where the model fabricates entities (e.g., agencies, actions, or clauses) absent from the rule knowledge set. Second, adopting the LLM-as-a-Judge paradigm (Zheng et al. 2023), we prompt an independent judge (o3) to assign three ten-point scores: intrinsic, extrinsic, and composite. Finally, to reveal mutual influences that coarse metrics obscure, we define two relational metrics: **Reliability**, which measures whether application or complex reasoning is supported by high memorization scores; and **Reasoning Capability**, which quantifies complex reasoning when all relevant rules are memorized.

Empirical experiments on five popular LLMs and five representative enhancement methods reveal a critical bottleneck: despite adequate rule memorization and single-rule application, multi-rule complex reasoning stalls at 5.4/10, significantly limiting practical deployment. To mitigate this gap, we devise **RAMPS**, a **R**ule knowledge-**A**ware **M**onte Carlo Tree Search **P**rocess-Reward **S**upervision framework. In contrast to prior process-reward models (PRMs) that target mathematics or code, RAMPS tackles open-ended, rule-driven reasoning where finding the earliest erroneous step is non-trivial. Intuitively, steps whose prerequisite rules are weakly memorized or misapplied are more likely to derail the reasoning chain. Therefore, we give exploration priority to these error-prone steps and then apply binary search to locate the first potential error step (Luo et al. 2025b). RAMPS then automatically generates 12K step-level annotations and uses them to train an advantage-based PRM, effectively reducing hallucinations and significantly boosting complex reasoning performance to 7.7 (+2.3). Extensive experiments demonstrate that rule-aware process supervision substantially advances rule knowledge–driven reasoning.

In summary, our contributions are four-fold:

- We formalize *rule knowledge–driven reasoning*, where LLMs rely on expert rules without training examples, complementing the prevailing example-driven paradigm.
- We release **RULER**, an automatic benchmark of 32K questions with hallucination-aware, relational metrics for rule memorization, application, and complex reasoning.
- We conduct a comprehensive empirical study on RULER revealing substantial deficits in multi-rule complex reasoning and offering practical insights for future research.
- We propose **RAMPS**, a rule-aware Monte Carlo Tree Search process-supervision framework that reduces hallucinations, and markedly improves complex reasoning.

Related Work

Rule-based reasoning. Classic association rule mining (Agrawal and Srikant 1994; Galárraga et al. 2013) lays the data-driven foundation and then adapts to the knowledge graph (Lu et al. 2022; Luo et al. 2024, 2025a). Although recent efforts translate natural language rules into logic via templates (Zhang et al. 2023; Servantez et al. 2024; Xu et al. 2024), their domain-specific grammars and reliance on symbolic engines hinder scalability.

Evaluating LLMs. General benchmarks (MMLU, BIG-bench, HELM) (Hendrycks et al. 2021; Srivastava et al. 2023; Liang et al. 2023) and domain probes for math, medicine, and law (Liu et al. 2024a; Singhal et al. 2023; Li et al. 2024) report single aggregate scores, rarely separating abilities or quantifying hallucination. RULER instead isolates reasoning abilities, and introduces hallucination-aware and relational metrics for trust-sensitive diagnosis.

Enhancing LLM reasoning. Chain-of-Thought and Self-Consistency (Wei et al. 2022; Wang et al. 2023) boost reasoning with outcome-level signals, while process-reward models give step-level feedback for math (PRM800K, MATH-SHEPHERD, MIPS) (Lightman et al. 2023; Wang et al. 2024a,b). RAMPS generalizes PRM to open-ended, rule knowledge-driven reasoning.

Preliminaries

Rule Knowledge. Rule knowledge describes prescriptive domain expertise in natural language: if a certain *state* occurs, a specified *action* should be taken within a domain-specific context. To formalize rule knowledge, we adopt the concept of *atomic knowledge* from the commonsense knowledge graph ATOMIC (Shen, Wu, and Xia 2023), and represent each rule as a triplet:

$$r = \{\textit{Subject}, \textit{State}, \textit{Action}\}, \tag{1}$$

where *Subject* denotes the domain (e.g., *Earthquake*); *State* indicates the triggering condition (e.g., "The occurrence of a major earthquake disaster"); and *Action* is the prescribed response (e.g., "The provincial government activates a Level-II emergency response"). Given a specific domain, the rule knowledge set is denoted by $\mathcal{R} = \{r_i\}_{i=1}^{|\mathcal{R}|}$.

Rule Knowledge-Driven Reasoning. We formulate *rule knowledge–driven reasoning* as a type of open-ended reasoning, in which a model $\mathcal{M}$ receives a domain-specific rule knowledge set $\mathcal{R}$, without any explicit examples for training. Given limited indirect feedback through a small development set $D_{\text{dev}} = \{(q_i, a_i)\}_{i=1}^{|D_{\text{dev}}|}$, where the gold references $\{a_i\}$ are not exposed to $\mathcal{M}$, the goal is to develop an enhancement method $f_{\boldsymbol{\theta}}$ that generalizes the reasoning ability of $\mathcal{M}$ to a large-scale unseen test set $D_{\text{test}} = \{(q_j, a_j)\}_{j=1}^{|D_{\text{test}}|}$.

Formally, we optimize $f_{\boldsymbol{\theta}}$ by maximizing the expectation of evaluation score assigned by an automatic judge model $\mathcal{J}$ (following the LLM-as-a-Judge paradigm (Zheng et al. 2023)):

$$f_{\boldsymbol{\theta}}^{*} = \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{(q,a)\sim D_{\text{dev}}}\Big[\mathcal{J}\big(a,\ f_{\boldsymbol{\theta}}(\mathcal{M}(q), \mathcal{R})\big)\Big]. \tag{2}$$

The optimized enhancement method $f_{\boldsymbol{\theta}}^{*}$ is subsequently evaluated directly on the unseen test set D_{test}, which clearly distinguishes our *rule knowledge–driven reasoning* setting from the conventional *example-driven learning paradigm*.

RULER: Benchmark Construction

Rule Knowledge Set Construction. As illustrated in Figure 1 (Stage 1), we initially collected 34 authoritative emergency response documents from official government portals,

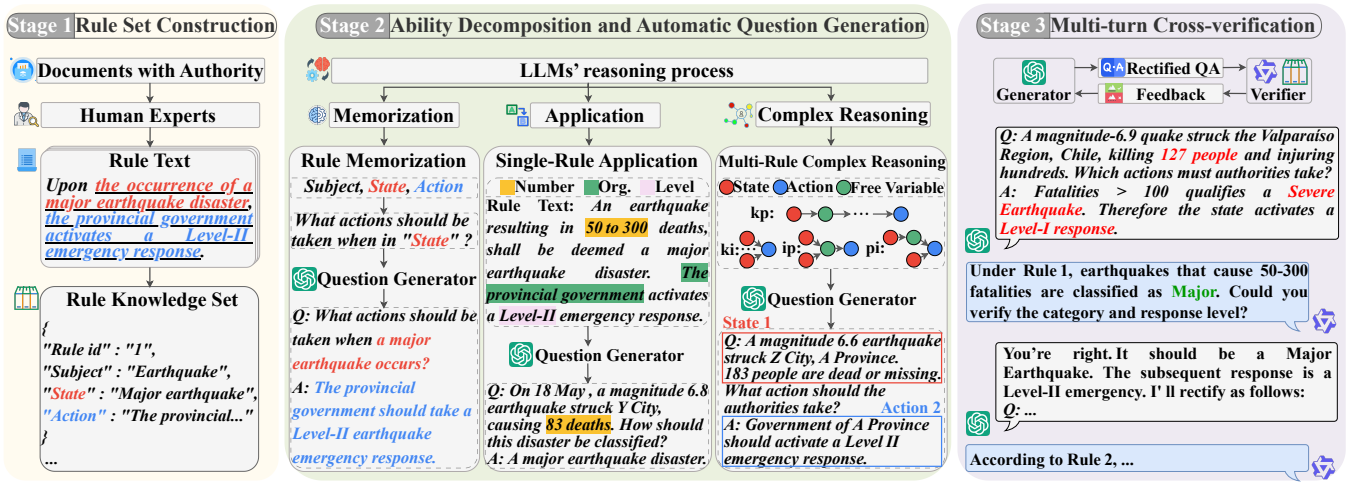

Figure 1: The overall workflow of RULER construction.

covering 21 distinct domains such as earthquakes, fires, traffic accidents, industrial safety, and cybersecurity incidents. These documents were rigorously distilled by human experts into 1K well-structured natural language rules, each formalized into a triplet $r = \{Subject, State, Action\}$, forming the comprehensive rule set $\mathcal{R}$.

Ability Decomposition and Question Generation. For fine-grained evaluation, we decompose LLM reasoning into three core abilities (Figure 1, Stage 2):

Rule Memorization assesses whether $\mathcal{M}$ can precisely recall a given rule $r_i \in \mathcal{R}$, formulated as: $\mathcal{M}(q_m) \equiv_{\text{sem}} a_m$, where $\equiv_{\text{sem}}$ denotes semantic equivalence (Tarski 1956). The memorization question q_m is generated by rewriting the *State* of rule r_i into a formal interrogative sentence, and the *Action* is directly used as the gold reference a_m.

Single-Rule Application measures whether $\mathcal{M}$ can understand and apply a single rule to a specific, contextually enriched problem. Formally: $\mathcal{M}(q_a) \equiv_{\text{sem}} a_a$. To ensure consistency with the original rules, we use the open-source entity extraction toolkit Stanza (Qi et al. 2020) on the rule texts, guiding the LLM to embed specific details such as locations, organizations, and numbers into question contents.

Multi-Rule Complex Reasoning evaluates the capability of integrating multiple logically related rules $\{r_i\}_{i=1}^{C}$ to solve complex reasoning tasks: $\mathcal{M}(q_c(\{r_i\}_{i=1}^{C})) \equiv_{\text{sem}} a_c$. Inspired by complex logical queries (Ren, Hu, and Leskovec 2020), we construct multi-hop reasoning chains based on query structures (*kp*, *ki*, *ip*, and *pi*) with two logical operators—projection and intersection. For example, if $Action_i \equiv_{\text{sem}} State_j$, rules r_i and r_j form a projection chain (2*p* query); if $Action_i \equiv_{\text{sem}} Action_j$, they constitute an intersection chain (2*i* query). Question generation begins from an anchor (Figure 1, *State 1* of the first rule), sequentially examines intermediate states and actions as free variables, and concludes at the final *Action* (*State 2* of the last rule).

Multi-turn Cross-verification. To ensure the quality of generated questions (Figure 1, Stage 3), we devise a multi-turn cross-verification (Cohen et al. 2023). Specifically, a verifier (QwQ-32B) challenges the generator (o3) with clarifying questions referencing the corresponding rules. The generator iteratively responds, allowing the verifier to identify factual or logical flaws. Rectification proceeds in repeated dialogue rounds until all issues are resolved (e.g., correcting an incorrect response level in Figure 1).

Benchmark Statistics and Analysis. From the 1,000-rule set, we constructed a benchmark comprising 32K questions: 11K for rule memorization, 11K for single-rule application, and 10K for multi-rule complex reasoning. The benchmark is not used for parametric training: it is partitioned into a 3K development set D_{dev} (used only for optimizing $f_{\boldsymbol{\theta}}$) and a 29K test set D_{test}. To validate question quality, we conduct human verification on 1K randomly sampled questions per category. Results indicate a very low error rate: 0.1%, 1.2%, and 3.7%, respectively, highlighting the benchmark's reliability and scalability without manual annotation.

Evaluation Metrics

Hallucination-aware Evaluation. We further distinguish two types of hallucinations: **Intrinsic Hallucination**, which refers to answers that explicitly conflict with the provided question content q (e.g., altering casualty numbers), and **Extrinsic Hallucination**, which involves fabricated entities or actions beyond the rule knowledge set $\mathcal{R}$. Given the open-ended nature of RULER, we adopt the LLM-as-a-Judge evaluation paradigm (Zheng et al. 2023). According to rigorous criteria, an independent model $\mathcal{J}$ (a separate o3) assigns **Intrinsic**, **Extrinsic**, and **Composite** scores on a 1–10 scale to each answer generated by the LLM $\mathcal{M}$. Given a question q, the enhancement method $f_{\boldsymbol{\theta}}$ together with $\mathcal{M}$ and $\mathcal{R}$ produces an answer $\hat{a}(q)$, and we denote its composite score by $\mathcal{J}_{\text{comp}}(q) := \mathcal{J}_{\text{comp}}(\hat{a}(q))$.

Reliability. To assess if $\mathcal{M}$ reasoning reliably utilizes memorized rules, we define Reliability for each single-rule

Model	Method	Memorization			Application			Complex Reasoning			Reliability	Reasoning	Average
		Intr	Extr	Comp	Intr	Extr	Comp	Intr	Extr	Comp			
InternLM2.5-7B	Vanilla	4.4	5.0	4.2	4.9	4.8	4.5	3.8	3.6	3.4	4.7	3.4	4.2
	Few-Shot	4.8	4.9	4.5	5.0	4.7	4.7	3.9	3.5	3.6	4.6	3.6	4.3
	RAG	7.0	7.4	6.6	5.4	6.4	5.4	4.5	4.9	4.3	6.9	4.3	5.6
	RAG + Few-Shot	7.3	7.6	6.9	5.8	6.7	5.8	4.6	5.0	4.4	7.1	4.5	5.8
	Self-Training	6.9	7.2	6.5	5.6	6.2	5.8	3.9	4.6	4.1	6.6	4.1	5.6
	Self-Training + DPO	7.4	7.6	6.8	5.8	6.9	6.3	4.2	4.8	4.5	7.2	4.6	6.0
Qwen3-8B	Vanilla	4.9	5.4	4.5	5.3	5.1	4.8	4.1	4.0	3.7	5.1	3.8	4.6
	Few-Shot	5.4	5.2	4.8	5.4	4.9	4.9	4.3	3.7	3.9	4.8	4.0	4.7
	RAG	7.7	8.0	7.3	5.9	7.0	5.9	4.7	5.2	4.6	7.5	4.7	6.2
	RAG + Few-Shot	8.0	8.1	7.5	6.4	7.2	6.3	4.9	5.3	4.7	7.6	4.9	6.4
	Self-Training	7.5	7.7	7.1	6.1	6.8	6.0	4.1	5.2	4.5	7.1	4.7	6.1
	Self-Training + DPO	7.9	8.0	7.4	6.3	7.5	6.8	4.8	5.4	4.9	7.7	5.3	6.6
GLM-Z1-9B	Vanilla	4.8	5.2	4.4	5.2	5.0	4.7	4.0	3.8	3.6	4.9	3.8	4.5
	Few-Shot	5.1	5.0	4.6	5.4	4.9	4.9	4.2	3.7	3.8	4.9	3.9	4.6
	RAG	7.2	7.6	6.8	5.7	6.7	5.8	4.6	5.0	4.4	7.0	4.5	5.9
	RAG + Few-Shot	7.5	7.8	7.1	6.1	6.9	6.2	4.8	5.1	4.6	7.2	4.7	6.2
	Self-Training	6.9	7.2	6.5	5.9	6.6	6.0	4.0	4.7	4.2	6.7	4.4	5.8
	Self-Training + DPO	7.4	7.7	7.0	6.1	7.4	6.7	4.3	4.9	4.7	7.6	5.0	6.2
Qwen3-14B	Vanilla	5.4	5.8	5.0	5.8	5.5	5.2	4.4	4.3	4.1	5.7	4.2	5.0
	Few-Shot	5.9	5.7	5.3	5.9	5.3	5.4	4.6	4.0	4.2	5.6	4.5	5.1
	RAG	8.2	8.4	7.8	6.3	7.4	6.2	5.0	5.6	4.9	7.9	5.2	6.7
	RAG + Few-Shot	8.6	8.7	8.1	6.8	7.6	6.6	5.2	**5.8**	5.1	8.2	5.6	6.9
	Self-Training	7.9	8.1	7.5	6.5	7.3	6.4	4.9	5.3	4.8	7.5	5.2	6.6
	Self-Training + DPO	8.3	8.4	7.9	**6.9**	**8.1**	**7.5**	**5.3**	5.7	**5.4**	**8.4**	**5.9**	**7.1**
GPT-4o	Vanilla	5.7	6.1	5.3	5.9	5.6	5.4	4.3	4.2	4.0	6.0	4.1	5.1
	Few-Shot	6.2	6.0	5.6	6.0	5.4	5.5	4.6	4.1	4.2	5.9	4.5	5.3
	RAG	8.5	8.8	8.3	6.2	7.5	6.2	4.9	5.5	4.8	7.8	5.1	6.7
	RAG + Few-Shot	**8.9**	**9.0**	**8.4**	6.5	7.7	6.5	5.0	**5.8**	5.0	8.3	5.5	7.0

Table 1: Main results. **Intr/Extr**: Intrinsic/Extrinsic hallucination; **Comp**: Composite score. Bold scores denote the best results and underlined scores indicate the second-best results. All answers are assessed fully automatically by the judge model (o3).

application or multi-rule complex reasoning question q_j as:

$$\text{Reliability}(q_j) = \mathbb{I}\big(\mathcal{J}_{\text{comp}}(q_j) \geq \tau_{\text{comp}}\big) \cdot \frac{1}{|\mathcal{R}_j|} \sum_{r_i \in \mathcal{R}_j} \mathcal{J}_{\text{comp}}\big(q_m(r_i)\big), \quad (3)$$

where τ_{comp} is a composite score threshold (e.g., 6), $\mathbb{I}(\cdot)$ is the indicator function, $\mathcal{R}_j$ represents the referenced rules of q_j, and $q_m(r_i)$ is the memorization question for rule r_i. Higher Reliability indicates that the answer to q_j is supported by strong memorization of all its referenced rules.

Reasoning Capability. To isolate and evaluate the LLM's genuine reasoning ability beyond memorization, we define Reasoning Capability for multi-rule question $q_c(\mathcal{R}_c)$ as:

$$\text{ReasoningCapability}(q_c) = \frac{\sum_{r_i \in \mathcal{R}_c} \mathbb{I}\left(\mathcal{J}_{\text{comp}}\big(q_m(r_i)\big) \geq \tau_{\text{comp}}\right) \cdot \mathcal{J}_{\text{comp}}(q_c(\mathcal{R}_c))}{\sum_{r_i \in \mathcal{R}_c} \mathbb{I}\left(\mathcal{J}_{\text{comp}}\big(q_m(r_i)\big) \geq \tau_{\text{comp}}\right)}, \quad (4)$$

where $\mathcal{R}_c = \{r_i\}_{i=1}^{C}$ is the set of referenced rules of q_c. A high Reasoning Capability score indicates effective utilization of memorized rule knowledge for complex reasoning.

Empirical Study and Analysis

Experimental Setup

We conduct extensive empirical experiments on five popular LLMs (InternLM2.5 (Wu et al. 2024), GLM-Z1 (GLM et al. 2024), Qwen3-8B, Qwen3-14B (Yang et al. 2025a), and GPT-4o (Hurst et al. 2024)). Beyond the vanilla setting, we assess five representative enhancement methods: **Few-Shot** (leveraging minimal demonstrations), **RAG** (retrieving top-K relevant rules as context), **RAG + Few-Shot**, **Self-Training** (fine-tuning on self-generated QA pairs), and **Self-Training + DPO** (refining via Direct Preference Optimization (Rafailov et al. 2023) on high-quality self-generated QA pairs filtered via self-consistency (Wei Jie et al. 2024)). The RULER benchmark dataset and evaluation scripts are available at https://github.com/TheoryRhapsody/RULER.

Experimental Results

Experimental results on RULER are summarized in Table 1. We emphasize the following empirical findings.

Vanilla models underperform across all categories: Without enhancement, the composite scores for Memorization (4.2–5.3), Application (4.5–5.4), and Complex Reason-

ing (3.4–4.1) remain low, with severe intrinsic and extrinsic hallucinations (mean: 4.9). This highlights the benchmark's challenge and the limitations of vanilla models, even for GPT-4o, indicating that pre-training alone is insufficient.

Non-parametric methods reduce hallucinations to a certain extent: Few-Shot modestly improves intrinsic hallucination scores (mean gain: +0.3) but leaves extrinsic hallucinations relatively unchanged. RAG significantly mitigates extrinsic hallucinations (mean gain: +1.9) and improves response reliability (mean gain: +2.1). However, its effectiveness decreases from Memorization (max: 8.3) to Complex Reasoning (max: 4.9), reflecting the higher retrieval burden in the latter, where essential rule-based knowledge is not explicitly referenced in the question. Notably, RAG + Few-Shot achieves strong scores in Memorization (max: 8.4) and Application (max: 6.6), validating the feasibility of rule knowledge-driven reasoning with minimal examples.

Parametric methods exhibit diminishing returns on complex tasks: Self-Training moderately alleviates intrinsic hallucinations (mean gain: +1.1) and extrinsic hallucinations (mean gain: +1.6), excelling mainly in Memorization (max: 7.5). However, performance declines sharply for Application (max: 6.4) and Complex Reasoning (max: 4.8), due to propagation of inherent hallucinations in the self-generated data. Self-Training + DPO partially corrects these biases, further enhancing extrinsic hallucination scores (mean gain over Self-Training: +0.5) and Reliability (mean gain: +0.8), yet intrinsic hallucination scores (mean: 6.2) and Reasoning Capability (mean: 5.2) remain suboptimal.

Current methods struggle significantly with Complex Reasoning: Despite improvements, all models and methods remain unsatisfactory in complex reasoning tasks, with a mean composite score of 4.4. Even the best-performing model (Qwen3-14B, Self-Training + DPO) achieves only 5.4. Additionally, Reasoning Capability scores remain consistently below 6.0, indicating severe limitations in utilizing memorized knowledge for multi-hop reasoning.

Empirical Analysis

Empirical insights are illustrated in Figure 2 and Figure 3.

Reliability strongly correlates with extrinsic hallucination in complex reasoning: Figure 2 shows that Reliability strongly correlates with Complex Reasoning extrinsic hallucination ($\rho = 0.971$) and composite score ($\rho = 0.922$). Low Reliability directly indicates insufficient internalized rule knowledge, leading to higher extrinsic hallucinations and reduces performance.

Reasoning Capability strongly correlates with intrinsic hallucination in complex reasoning: Reasoning Capability presents strong correlation with intrinsic hallucination ($\rho = 0.971$) and composite score ($\rho = 0.885$), implying intrinsic hallucinations arise primarily from logical errors in reasoning, which significantly reduce overall quality.

Single-hop reasoning accuracy is foundational for complex reasoning: Reasoning Capability exhibits a pronounced association with Application composite score ($\rho = 0.951$), underscoring that accurate single-step reasoning underpins effective multi-hop reasoning, emphasizing stepwise correctness to enhance global reasoning competence.

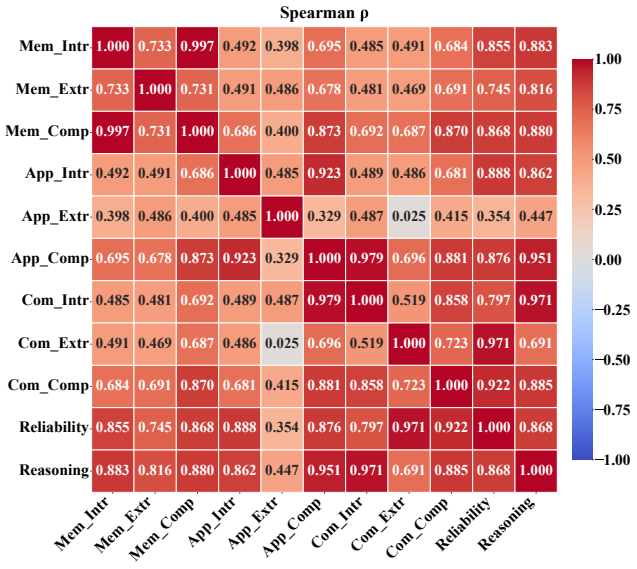

Figure 2: Spearman rank correlation (ρ) heatmap of metrics.

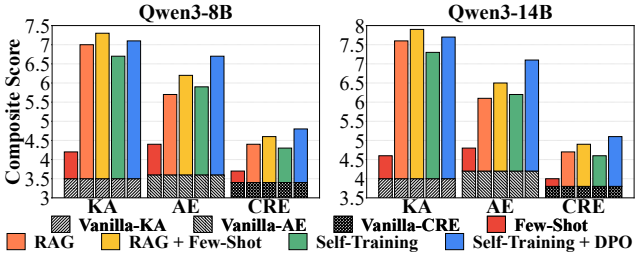

Figure 3: Evaluation of the Knowledge Acquisition (KA), Application Enhancement (AE), and Complex Reasoning Enhancement (CRE) on "hard" items (vanilla score ≤ 5).

Existing methods have limited enhancement on complex reasoning: To mitigate data contamination—models exploit pre-training priors rather than rule knowledge, we quantify enhancement gains over vanilla on "hard" items (vanilla composite $\leq$ 5) across three question categories: Knowledge Acquisition (Memorization), Application Enhancement (Application), and Complex Reasoning Enhancement (Complex Reasoning). Figure 3 illustrates substantial gains in Knowledge Acquisition (max gain: +4.0) and Application Enhancement (max gain: +3.1), yet their efficacy drops sharply on Complex Reasoning Enhancement (max gain: +1.4). This underscores a fundamental limitation of existing enhancement strategies for complex reasoning.

RAMPS: Rule Knowledge-Aware MCTS Process-Reward Supervision

Rule-Aware MCTS for Process Annotation. Existing methods perform well on memorization and application

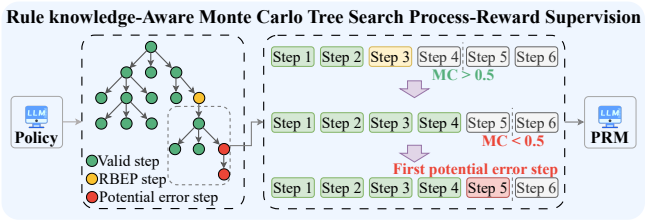

Figure 4: Overview of RAMPS framework.

tasks but consistently fail at complex reasoning. Motivated by recent advances in Test-Time Scaling (Snell et al. 2024), we propose a rule knowledge-aware MCTS to automatically annotate reasoning steps. Specifically, given a complex reasoning question $q_c(\mathcal{R}_c)$ with reasoning path $x_{1:L}$, we leverage the previous observation in the empirical study to identify *Rule-Based Error-Prone (RBEP)* steps — steps whose referenced rules obtain memorization or application composite scores ≤ 5. Using Self-Training + DPO optimized Qwen3-8B as the policy model, we generate an initial corpus by sampling reasoning paths on the multi-rule complex reasoning subset of the dev set (1K Complex Reasoning).

To efficiently locate the first potential error step (sufficient for annotation), we integrate a binary search into MCTS (Figure 4). Given a state $s = (q_c, x_{1:t})$ and the current reasoning step x_t, we perform k rollouts from x_t: $\Psi_k(x_{1:t}) = \{\psi^{(i)}\}_{i=1}^{k}$ to estimate the expected composite score MC, and devise the rule knowledge-aware prior p_s to evaluate Memorization or Application abilities in state s:

$$MC(q_c, x_{1:t}) = \sigma_{T_s,\tau_s}\left(\frac{1}{k}\sum_{i=1}^{k}\psi^{(i)}\right), \tag{5}$$

$$p_s = \sigma_{T_k,\tau_k}\left(\frac{1}{|\mathcal{R}_c|}\sum_{r_i\in\mathcal{R}_c}\mathcal{J}_{\text{comp}}(q(r_i))\right), \tag{6}$$

where σ is the sigmoid function, τ_s is the composite score threshold (e.g., 6), and T_s is the temperature. The advantage function capturing future gains is defined as:

$$\hat{A}(q_c, x_{1:t}) = \text{clip}\left(MC(q_c, x_{1:t+1}) - MC(q_c, x_{1:t}), -c, c\right), \tag{7}$$

where $\text{clip}(\cdot, -c, c) = \max(-c, \min(\cdot, c))$. Combining advantage $\hat{A}(q_c, x_{1:t})$ and rule knowledge prior p_s, the PUCT algorithm selects a rollout ψ for binary search:

$$Q(x_{1:t}, \psi) = \alpha^{1-MC(q_c,x_{1:t})}(1+\gamma p_s)e^{\hat{A}(q_c,x_{1:t})}, \tag{8}$$

$$(x_{1:t}, \psi) = \arg\max_{(x_{1:t},\psi)}\left[Q(x_{1:t},\psi) + c_{\text{puct}}\frac{\sqrt{\sum_i N(s_i)}}{1+N(x_{1:t},\psi)}\right], \tag{9}$$

where $Q(x_{1:t}, \psi)$ denotes the rollout value, $N(x_{1:t}, \psi)$ counts visits. The factor $\alpha^{1-MC(q_c,x_{1:t})}$ prioritizes the low-scoring rollouts from states with a high expected composite score MC, steering the search toward error-prone regions worthy of annotation. Hyper-parameters γ, c_{puct} control the exploration-exploitation balance. The binary search efficiently ($\mathcal{O}(k \log N)$ complexity) localizes the earliest error step, yielding 12K high-quality automatic annotations.

Advantage-based PRM Training. We further train a process-reward model (PRM) initialized from the policy model to guide globally optimal solutions. Instead of predicting $MC(q_c, x_{1:t})$, the PRM models advantage signals $\hat{A}(q_c, x_{1:t})$. The training objective combines a logistic loss with a consistency regularizer to enhance generalization:

$$\mathcal{L} = \sum_{(q_c,x_{1:t})}\left[\log(1+e^{-\kappa\hat{A}(q_c,x_{1:t})z_\theta(q_c,x_{1:t})}) + \lambda\mathbb{E}_{\xi\sim\mathcal{Q}}\text{KL}(\pi_\xi|\bar{\pi})\right], \tag{10}$$

where $z_\theta(q_c, x_{1:t})$ is the PRM output logit, κ controls loss steepness, λ balances the consistency regularizer, $\mathcal{Q}$ denotes a perturbation distribution, and π_ξ and $\bar{\pi}$ are the PRM-induced predictive distributions under perturbation ξ and their average, respectively. During inference, RAMPS employs beam search with beam width B, selecting the top-B partial reasoning paths at each step. Each candidate expands into k successor steps, which are scored by the PRM and iteratively extended until completion.

Experiments

Experimental Setup

We further evaluate RAMPS using two strong vanilla models, Qwen3-8B and Qwen3-14B, on challenging Complex Reasoning tasks. We compare RAMPS with two test-time scaling approaches: Majority Voting (MV), selecting the most frequent reasoning path among N samples via self-consistency; and Outcome Reward Model (ORM), ranking and selecting the best of N samples by outcome scores.

Main Results

Test-Time Scaling consistently boosts performance: As shown in Table 2, the composite scores improves consistently: MV achieves slight gains (4.1–6.0), ORM yields further improvement (4.4–6.4), and RAMPS delivers the highest results (4.9–7.7), demonstrating the effectiveness of leveraging test-time scaling for enhancing multi-rule complex reasoning without *example-driven learning paradigm*.

MV provides limited gains with minimal cost: Despite its simplicity, MV marginally improves intrinsic and extrinsic hallucinations (mean gains: +0.4 and +0.3), limiting the enhancement of the Reasoning Capability (mean: 5.3).

ORM achieves moderate improvement via outcome-based feedback: By providing feedback on the final answer, ORM reduces extrinsic hallucinations (4.5–6.7) and improves Reasoning Capability (4.6–7.0). However, intrinsic hallucinations remained a bottleneck (max: 6.5), restricting overall reasoning performance (max: 6.4).

RAMPS substantially improves reasoning through rule-aware process supervision: Leveraging step-level detailed annotations from Rule-Aware MCTS, RAMPS effectively pinpoints rule-based error-prone steps, significantly mitigating intrinsic hallucinations (max: 7.6) and substantially boosting Reasoning Capability (max: 8.3). Consequently, RAMPS substantially improves the best Composite score from 5.4 to 7.7, clearly outperforming MV and ORM.

Model	Method	Intr	Extr	Comp	Reasoning
	Vanilla	4.1	4.0	3.7	3.8
	Few-Shot	4.3	3.7	3.9	4.0
	+ MV	4.4	4.1	4.1 (+0.2↑)	4.3
	+ ORM	4.8	4.5	4.4 (+0.5↑)	4.6
	+ RAMPS	5.3	4.9	4.9 (+1.0↑)	5.1
	RAG	4.7	5.2	4.6	4.7
	+ MV	4.9	5.3	4.8 (+0.2↑)	4.9
	+ ORM	5.3	5.6	5.2 (+0.6↑)	5.3
	+ RAMPS	5.8	6.1	5.9 (+1.3↑)	6.0
	RAG + Few-Shot	4.9	5.3	4.7	4.9
Qwen3-8B	+ MV	5.1	5.4	4.8 (+0.1↑)	5.0
	+ ORM	5.5	5.8	5.4 (+0.7↑)	5.5
	+ RAMPS	6.1	6.3	6.2 (+1.5↑)	6.3
	Self-Training	4.1	5.2	4.5	4.7
	+ MV	4.6	5.3	4.8 (+0.3↑)	5.1
	+ ORM	5.1	5.4	5.2 (+0.7↑)	5.6
	+ RAMPS	5.8	6.0	5.9 (+1.4↑)	6.4
	Self-Training + DPO	4.8	5.4	4.9	5.3
	+ MV	5.4	5.6	5.5 (+0.6↑)	5.9
	+ ORM	5.8	5.9	5.8 (+0.9↑)	6.3
	+ RAMPS	6.6	6.9	6.8 (+1.9↑)	7.2
	Vanilla	4.4	4.3	4.1	4.2
	Few-Shot	4.6	4.0	4.2	4.5
	+ MV	5.0	4.7	4.6 (+0.4↑)	4.8
	+ ORM	5.5	5.1	5.0 (+0.8↑)	5.3
	+ RAMPS	5.9	5.4	5.3 (+1.1↑)	5.6
	RAG	5.0	5.6	4.9	5.2
	+ MV	5.4	5.9	5.3 (+0.4↑)	5.5
	+ ORM	5.9	6.2	5.7 (+0.8↑)	6.0
	+ RAMPS	6.4	6.8	6.4 (+1.5↑)	6.6
	RAG + Few-Shot	5.2	5.8	5.1	5.6
Qwen3-14B	+ MV	5.5	6.0	5.4 (+0.3↑)	5.8
	+ ORM	6.1	6.4	6.0 (+0.9↑)	6.4
	+ RAMPS	6.8	7.2	6.9 (+1.8↑)	7.2
	Self-Training	4.9	5.3	4.8	5.2
	+ MV	5.5	5.6	5.3 (+0.5↑)	5.7
	+ ORM	5.9	6.0	5.7 (+0.9↑)	6.2
	+ RAMPS	6.7	6.8	6.6 (+1.8↑)	7.1
	Self-Training + DPO	5.3	5.7	5.4	5.9
	+ MV	6.0	6.2	6.0 (+0.6↑)	6.5
	+ ORM	6.5	6.7	6.4 (+1.0↑)	7.0
	+ RAMPS	**7.6**	**7.9**	**7.7** (+2.3↑)	**8.3**

Table 2: Main results on Multi-Rule Complex Reasoning. **Intr/Extr/Comp**: Intrinsic/Extrinsic/Composite score. Numbers with ↑ indicate absolute improvements.

Further Analysis

Ablation Study. To verify RAMPS, we compare two variants: (1) **MC**: replacing advantage signals $\hat{A}(q_c, x_{1:t})$ with expected score $MC(q_c, x_{1:t})$ for PRM training; and (2) **Best-of-N**: substituting beam search with Best-of-N, equalizing total sample counts for fairness. Figure 5 shows performance declines of 0.2–0.6 points when using $MC(q_c, x_{1:t})$, confirming advantage signals better avoid local optima. Nevertheless, MC variant remains superior to ORM. Best-of-N slightly reduces performance (mean degradation: -0.4), as iterative expansion via beam search better aligns with RAMPS's scaling strategy. Note that Best-of-N inference is over twice as efficient, suitable for latency-critical scenarios.

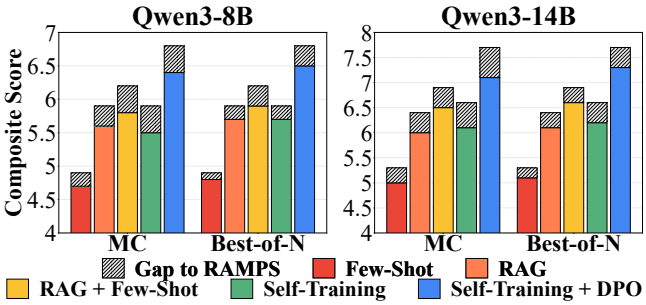

Figure 5: Comparison of RAMPS with two variants.

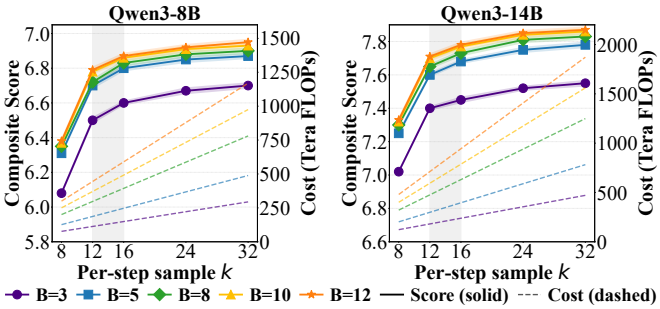

Figure 6: Cost-performance trade-off of Beam Search.

Cost-performance Trade-off. To balance the cost and performance of the beam search, we further investigate the influence of per-step sample k and beam width B on reasoning enhancement and the corresponding decoding cost per question. We vary $k \in \{8, 12, 16, 24, 32\}$ and $B \in \{3, 5, 8, 10, 12\}$, and adopt Qwen3-8B and Qwen3-14B with Self-Training + DPO + RAMPS combinations for exploration. Figure 6 overlays *Composite Score* (solid lines) with the *cost* in TFLOPs (dashed lines) for each $\langle k, B \rangle$ setting, making the efficiency frontier visually explicit.

For Qwen3-8B, increasing k from 8 to 12 improves by 0.4 on average while the cost grows only $\sim$1.5×, enlarging to $k = 16$ adds achieves slight gains (mean gains: +0.1) for another $\sim$1.3× cost, and further doubling to $k = 32$ yields $+0.07$ yet almost doubles cost again. A similar elbow appears for beam width: moving from $B = 3$ to 5 increases 0.2 on average at moderate cost (76.7 → 126.2 TFLOPs); each step beyond $B = 8$ achieve negligible gains < 0.05 while nearly doubling TFLOPs. Qwen3-14B exhibits the same trend with absolute cost $\sim$1.6× higher owing to larger size. Therefore, we recommend empirical settings of $k =$ 12–16, $B =$ 5–8 as cost-effective for complex reasoning.

Conclusion

We introduce RULER and RAMPS for benchmarking and enhancing LLMs on rule-knowledge-driven reasoning. RULER auto-generates 32K high-quality questions with fine-grained metrics, and RAMPS provides rule-aware step-level rewards that significantly enhance complex reasoning.

Acknowledgments

We thank the anonymous reviewers for their insightful comments. This work was supported by National Science Foundation of China (Grant Nos.62376057), the Start-up Research Fund of Southeast University (RF1028623234), and the Big Data Computing Center of Southeast University.

References

Agrawal, R.; and Srikant, R. 1994. Fast Algorithms for Mining Association Rules in Large Databases. In *VLDB*.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language Models are Few-Shot Learners. In *NeurIPS*.

Cohen, R.; Hamri, M.; Geva, M.; and Globerson, A. 2023. LM vs LM: Detecting Factual Errors via Cross Examination. In *EMNLP*.

Cui, J.; Li, Z.; Yan, Y.; Chen, B.; and Yuan, L. 2023. Chatlaw: Open-source legal large language model with integrated external knowledge bases. *CoRR*.

Galárraga, L. A.; Teflioudi, C.; Hose, K.; and Suchanek, F. 2013. AMIE: association rule mining under incomplete evidence in ontological knowledge bases. In *WWW*.

GLM, T.; Zeng, A.; Xu, B.; Wang, B.; Zhang, C.; Yin, D.; Zhang, D.; Rojas, D.; Feng, G.; Zhao, H.; et al. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*.

Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2021. Measuring Massive Multitask Language Understanding. In *ICLR*.

Hurst, A.; Lerer, A.; Goucher, A. P.; Perelman, A.; Ramesh, A.; Clark, A.; Ostrow, A.; Welihinda, A.; Hayes, A.; Radford, A.; et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Li, G.; Wang, P.; and Ke, W. 2023. Revisiting large language models as zero-shot relation extractors. *arXiv preprint arXiv:2310.05028*.

Li, G.; Wang, P.; Ke, W.; Xu, Z.; Liu, J.; and Shang, Z. 2025. On the Consistency of Commonsense in Large Language Models. In *Findings of ACL*.

Li, H.; Chen, Y.; Ai, Q.; Wu, Y.; Zhang, R.; and Liu, Y. 2024. Lexeval: A comprehensive chinese legal benchmark for evaluating large language models. *NIPS*, 37: 25061–25094.

Liang, P.; Bommasani, R.; Lee, T.; Tsipras, D.; Soylu, D.; Yasunaga, M.; Zhang, Y.; Narayanan, D.; Wu, Y.; Kumar, A.; Newman, B.; Yuan, B.; Yan, B.; Zhang, C.; Cosgrove, C. A.; Manning, C. D.; Re, C.; Acosta-Navas, D.; Hudson, D. A.; Zelikman, E.; Durmus, E.; Ladhak, F.; Rong, F.; Ren, H.; Yao, H.; WANG, J.; Santhanam, K.; Orr, L.; Zheng, L.; Yuksekgonul, M.; Suzgun, M.; Kim, N.; Guha, N.; Chatterji, N. S.; Khattab, O.; Henderson, P.; Huang, Q.; Chi, R. A.; Xie, S. M.; Santurkar, S.; Ganguli, S.; Hashimoto, T.; Icard, T.; Zhang, T.; Chaudhary, V.; Wang, W.; Li, X.; Mai, Y.; Zhang, Y.; and Koreeda, Y. 2023. Holistic Evaluation of Language Models. *TMLR*.

Lightman, H.; Kosaraju, V.; Burda, Y.; Edwards, H.; Baker, B.; Lee, T.; Leike, J.; Schulman, J.; Sutskever, I.; and Cobbe, K. 2023. Let's verify step by step. In *ICLR*.

Liu, H.; Zheng, Z.; Qiao, Y.; Duan, H.; Fei, Z.; Zhou, F.; Zhang, W.; Zhang, S.; Lin, D.; and Chen, K. 2024a. MathBench: Evaluating the Theory and Application Proficiency of LLMs with a Hierarchical Mathematics Benchmark. In *ACL*.

Liu, J.; Ke, W.; Wang, P.; Wang, J.; Gao, J.; Shang, Z.; Li, G.; Xu, Z.; Ji, K.; and Li, Y. 2024b. Fast and Continual Knowledge Graph Embedding via Incremental LoRA. In *IJCAI*.

Lu, S.; Liu, B.; Mills, K. G.; JUI, S.; and Niu, D. 2022. R5: Rule Discovery with Reinforced and Recurrent Relational Reasoning. In *ICLR*.

Luo, L.; Ju, J.; Xiong, B.; Li, Y.-F.; Haffari, G.; and Pan, S. 2025a. ChatRule: Mining Logical Rules with Large Language Models for Knowledge Graph Reasoning. In *PAKDD*.

Luo, L.; Liu, Y.; Liu, R.; Phatale, S.; Guo, M.; Lara, H.; Li, Y.; Shu, L.; Meng, L.; Sun, J.; and Rastogi, A. 2025b. Improve Mathematical Reasoning in Language Models with Automated Process Supervision. In *ICLR*.

Luo, L.; Zhao, Z.; Haffari, G.; Li, Y.-F.; Gong, C.; and Pan, S. 2024. Graph-constrained reasoning: Faithful reasoning on knowledge graphs with large language models. *arXiv preprint arXiv:2410.13080*.

OpenAI. 2025. Introducing OpenAI o3 and o4-mini. https://openai.com/index/introducing-o3-and-o4-mini/. Accessed: 2025-04-16.

Peng, C.; Yang, X.; Chen, A.; Smith, K. E.; PourNejatian, N.; Costa, A. B.; Martin, C.; Flores, M. G.; Zhang, Y.; Magoc, T.; et al. 2023. A study of generative large language model for medical research and healthcare. *NPJ digital medicine*.

Qi, P.; Zhang, Y.; Zhang, Y.; Bolton, J.; and Manning, C. D. 2020. Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. In *ACL*.

QwenLM. 2025. QwQ-32B: Embracing the Power of Reinforcement Learning. https://qwenlm.github.io/blog/qwq-32b/. Accessed: 2025-03-06.

Rafailov, R.; Sharma, A.; Mitchell, E.; Manning, C. D.; Ermon, S.; and Finn, C. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. In *NeurIPS*.

Ren, H.; Hu, W.; and Leskovec, J. 2020. Query2box: Reasoning over Knowledge Graphs in Vector Space Using Box Embeddings. In *ICLR*.

Servantez, S.; Barrow, J.; Hammond, K.; and Jain, R. 2024. Chain of Logic: Rule-Based Reasoning with Large Language Models. In *ACL*.

Shang, Z.; Ke, W.; Xiu, N.; Wang, P.; Liu, J.; Li, Y.; Luo, Z.; and Ji, K. 2024. Ontofact: Unveiling fantastic fact-skeleton of llms via ontology-driven reinforcement learning. In *AAAI*.

Shang, Z.; Liu, J.; Luo, Z.; Wang, P.; Ke, W.; Liu, J.; Xu, Z.; and Li, G. 2025. Acquisition and Application of Novel Knowledge in Large Language Models. In *ACL*.

Shen, X.; Wu, S.; and Xia, R. 2023. Dense-ATOMIC: Towards Densely-connected ATOMIC with High Knowledge Coverage and Massive Multi-hop Paths. In *ACL*.

Singhal, K.; Azizi, S.; Tu, T.; Mahdavi, S. S.; Wei, J.; Chung, H. W.; Scales, N.; Tanwani, A.; Cole-Lewis, H.; Pfohl, S.; et al. 2023. Large language models encode clinical knowledge. *Nature*.

Snell, C.; Lee, J.; Xu, K.; and Kumar, A. 2024. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*.

Srivastava, A.; Rastogi, A.; Rao, A.; Shoeb, A. A. M.; and et al. 2023. Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models. *TMLR*.

Sun, J.; Huang, W.; Wu, J.; Gu, C.; Li, W.; Zhang, S.; Yan, H.; and He, C. 2024. Benchmarking Chinese Commonsense Reasoning of LLMs: From Chinese-Specifics to Reasoning-Memorization Correlations. In *ACL*.

Tarski, A. 1956. *Logic, Semantics, Metamathematics: Papers from 1923 to 1938*. Clarendon Press.

Tong, W.; and Zhang, T. 2024. CodeJudge: Evaluating Code Generation with Large Language Models. In *ACL*.

Uesato, J.; Kushman, N.; Kumar, R.; Song, H. F.; Siegel, N. Y.; Wang, L.; Creswell, A.; Irving, G.; and Higgins, I. 2023. Solving Math Word Problems with Process-based and Outcome-based Feedback. In *ICLR*.

Wang, P.; Li, L.; Shao, Z.; Xu, R.; Dai, D.; Li, Y.; Chen, D.; Wu, Y.; and Sui, Z. 2024a. Math-Shepherd: Verify and Reinforce LLMs Step-by-step without Human Annotations. In *ACL*.

Wang, X.; Wei, J.; Schuurmans, D.; Le, Q. V.; Chi, E. H.; Narang, S.; Chowdhery, A.; and Zhou, D. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *ICLR*.

Wang, Z.; Li, Y.; Wu, Y.; Luo, L.; Hou, L.; Yu, H.; and Shang, J. 2024b. Multi-step Problem Solving Through a Verifier: An Empirical Analysis on Model-induced Process Supervision. In *EMNLP*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *NIPS*.

Wei Jie, Y.; Ferdinan, T.; Kazienko, P.; Satapathy, R.; and Cambria, E. 2024. Self-training Large Language Models through Knowledge Detection. In *EMNLP*.

Wu, Z.; Huang, S.; Zhou, Z.; Ying, H.; Wang, J.; Lin, D.; and Chen, K. 2024. Internlm2. 5-stepprover: Advancing automated theorem proving via expert iteration on large-scale lean problems. *arXiv preprint arXiv:2410.15700*.

Xu, Z.; Wang, P.; Ke, W.; Li, G.; Liu, J.; Ji, K.; Chen, X.; and Wu, C. 2024. Incorporating Schema-Aware Description into Document-Level Event Extraction. In *IJCAI-24*.

Yang, A.; Li, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Gao, C.; Huang, C.; Lv, C.; et al. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Yang, W.; Lin, Y.; Zhou, J.; and Wen, J.-R. 2025b. Distilling rule-based knowledge into large language models. In *COLING*.

Yu, J.; Wang, X.; Tu, S.; Cao, S.; Zhang-Li, D.; Lv, X.; Peng, H.; Yao, Z.; Zhang, X.; Li, H.; Li, C.; Zhang, Z.; Bai, Y.; Liu, Y.; Xin, A.; Yun, K.; GONG, L.; Lin, N.; Chen, J.; Wu, Z.; Qi, Y.; Li, W.; Guan, Y.; Zeng, K.; Qi, J.; Jin, H.; Liu, J.; Gu, Y.; Yao, Y.; Ding, N.; Hou, L.; Liu, Z.; Bin, X.; Tang, J.; and Li, J. 2024. KoLA: Carefully Benchmarking World Knowledge of Large Language Models. In *ICLR*.

Zhang, H.; Huang, J.; Li, Z.; Naik, M.; and Xing, E. 2023. Improved Logical Reasoning of Language Models via Differentiable Symbolic Programming. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *ACL*.

Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E.; Zhang, H.; Gonzalez, J. E.; and Stoica, I. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *NeurIPS*.