Review Article

Open Access Journal of Mathematical and Computer Science

V-MATH and Veri-Math: Step-Level Verification for Enhancing Mathematical Reasoning in LLMs

Rugaba John Paul[1,2] and Tang Shengbing[2*]

[1]*Faculty of science and technology, Victoria University, Uganda*
[2]*Central China Normal University, Wollongong Joint Institute, China*

[*]**Corresponding authors**
Tang Shengbing, Central China Normal University, Wollongong Joint Institute, China

Received: July 24, 2026; **Accepted:** August 06, 2026; **Published:** August 13, 2026

Abstract

Recent advances in large language models (LLMs) have led to notable progress in automated problem solving; however, mathematical reasoning remains a persistent point of weakness, particularly in settings that demand long chains of logically dependent steps. Errors introduced early in a solution frequently propagate unnoticed, yet most existing benchmarks evaluate only the final answer and therefore provide little insight into where or why a model's reasoning breaks down. This paper addresses this gap through two complementary contributions: the V-MATH dataset and the Veri-Math step-level verification model. V-MATH comprises more than 260,000 problems ranging from elementary material to advanced university and competition topics. Each problem is accompanied by a detailed, stepwise solution annotated with explicit correctness labels (✓, ✗, ⊖), allowing the precise identification of logical failures within the solution process. Constructing a dataset at this scale highlights several challenges, including maintaining consistent standards for step granularity across disparate mathematical domains and preserving annotation quality in the presence of diverse problem structures and reasoning styles.

Building on this dataset, this paper introduces Veri-Math, a model trained to assess the validity of individual reasoning steps rather than final answers. The verifier employs a dual-objective training strategy that combines step classification with ranking, enabling it to discriminate between sound reasoning and subtle logical inconsistencies. Evaluations on GSM8K, MATH, MathPile, and PRM-800K demonstrate notable gains when existing language models are paired with Veri-Math: for example, LLaMA 3-8B achieves an average improvement of 10.9%, Qwen improves by 15.4%, and Gemma 2- 2B achieves a 19.9% increase on PRM-800K. On the V-MATH test set, the verifier attains 84.4% accuracy in step-correctness classification.

Taken together, V-MATH and Veri-Math establish a framework for examining mathematical reasoning at a finer level of detail than has previously been feasible. While the work exposes ongoing challenges such as the difficulty of detecting highly nuanced logical errors and the inherent complexity of constructing consistent step-level annotations across heterogeneous topics, it provides a foundation for building more reliable and transparent computational reasoning systems suited to use in education, scientific inquiry, and formal problem solving.

Keywords: Large Language Models, Mathematical Reasoning, Step-Level Verification, Verifier Models, Contrastive Learning

Introduction

Large language models (LLMs), including GPT-3.5, GPT-4, PaLM-2, and LLaMA-2, have achieved exceptional performance across natural language processing tasks such as translation, summarization, and question answering. However, mathematical reasoning remains a persistent weakness. Unlike standard language tasks, mathematical problem-solving demands precise symbolic manipulation, sequential logical inference, and multi-stage computation where each step depends on the correctness of previous ones.

The autoregressive nature of LLMs is a core architectural limitation: once a token is generated, the model cannot

Citation: Rugaba John Paul, Tang Shengbing. V-MATH and Veri-Math: Step-Level Verification for Enhancing Mathematical Reasoning in LLMs.. Open Access J Math Comput Sci. 2026. 1(1): 1-6. DOI: doi.org/10.61440/OAJMCS.2026.v1.03

retroactively verify or revise it. Errors introduced early in a reasoning chain propagate silently, ultimately corrupting the final answer. Existing benchmark datasets such as GSM8K and MATH evaluate only final-answer correctness, providing no intermediate step supervision [1,2]. This gap in training and evaluation signal means LLMs receive little feedback about the location or nature of their logical failures.

Prompting strategies such as Chain-of-Thought (CoT) and Self-Consistency have improved the structural coherence of generated solutions, but neither method formally verifies the logical validity of individual reasoning steps [3,4]. Hallucinations fluent but logically incorrect stepsremain common, especially in complex multi-step derivations involving algebra, calculus, and probability.

This paper addresses these challenges through two complementary contributions: (1) V-MATH, a large-scale dataset of 260,000+ stepwise-annotated mathematical problems, and (2) Veri-Math, a step-level verification model trained via contrastive learning to discriminate between valid and flawed reasoning steps. Together, they establish a verifier-guided framework for fine-grained mathematical reasoning evaluation and improvement.

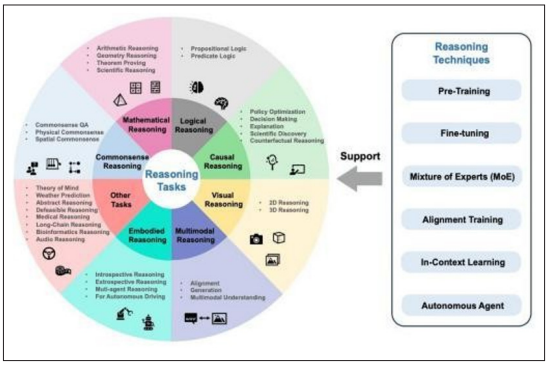


Figure 1: Overview of LLM reasoning tasks and techniques [5].

Related work

Mathematical Reasoning Benchmarks

GSM8K [1] provides ~8,500 grade-school arithmetic word problems widely used for Chain-of-Thought evaluation. The MATH dataset offers 12,500 competition-level problems spanning algebra, geometry, calculus, and number theory [2]. AQuA-RAT contains ~100,000 algebraic multiple-choice problems, while SVAMP tests model robustness via structural problem variations [6,7]. The DeepMind Mathematics Dataset covers over 2 million procedurally generated questions [8]. Critically, all these datasets provide only final-answer supervision with no intermediate step annotations, limiting their utility for verifier training.

Table 1: Summary of Mathematical Reasoning Datasets

Dataset	Level	Size	Step Labels
GSM8K	Elementary	~8.5k	✗
MATH	HS Competition	12.5k	✗
AQuA-RAT	Algebra	~100k	✗
SVAMP	Arithmetic	~1k	✗
V-MATH (ours)	Elem–University	260k+	✓ ✗ ⊖

Reasoning Strategies

Chain-of-Thought (CoT) prompting elicits intermediate reasoning steps before the final answer, improving transparency on datasets like GSM8K and MATH [3]. Self-Consistency decoding aggregates multiple CoT paths via majority vote [4]. Tree-of-Thought (ToT) treats reasoning as a decision tree enabling exploratory search [9]. ReAct interleaves reasoning with external tool actions [10]. Reflexion introduces a self-critique loop [11]. Despite their progress, these methods are generative: they do not formally verify the logical validity of intermediate steps, remaining susceptible to hallucinations.

Verifier Architectures

GRACE pioneered contrastive learning for reasoning chain discrimination [12]. FG-PRM applied fine-grained process reward modeling [13]. DERA and R-STAR introduced step-level classifiers for error detection [14]. Recent works WizardMath, MAmmoTH, DeepSeekMath have advanced reinforcement learning for mathematical reasoning [15-17]. OpenAI o1 demonstrated that inference-time computation substantially improves reasoning quality [18]. Veri-Math extends these foundations with a scalable, automatically labeled step-level verification framework.

V-Math Dataset Construction

Design Principles and Objectives

The V-MATH dataset was constructed with three core objectives: (1) stepwise supervision enabling fine-grained evaluation of reasoning chains through per-step correctness labels; (2) diversity and difficulty stratification spanning elementary, secondary, and university-level mathematics; and (3) support for contrastive training—including both valid and flawed reasoning paths to enable discriminative learning.

Each problem in V-MATH has a maximum of seven reasoning steps, ensuring uniform structure across difficulty levels. Solutions exceeding this limit are condensed through hierarchical grouping and redundancy removal. The dataset contains 260,000 problems with over 13.7 million annotated reasoning steps stored in tier-specific JSONL format.

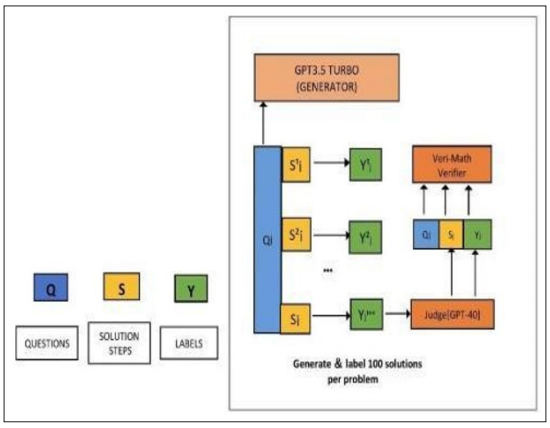


Figure 2: V-MATH Dataset Generation, Judgment, and Annotation Pipeline

Tier Structure and Problem Sources

Table 2: V-MATH Dataset Tier Structure

Tier	Topics	Steps	Problems	%
Elementary	Arithmetic, geometry	2–3	110,000	42.3%

Grade 12	Algebra, trig, calculus	3–5	127,000	48.8%
University	Calculus, LA, proofs	4–6+	23,000	8.8%
Total	—	—	260,000	100%

Problem sources were carefully selected by educational tier. Elementary problems were drawn from Math-Drills and Illustrative Mathematics. Grade 12 problems sourced from Khan Academy, ACT practice, GSM8K, MATH, and Uganda Certificate of Education materials. University-level problems came from Art of Problem Solving and Wyzant resources.

Solution Generation and Annotation

For each problem, GPT-3.5 Turbo generated five distinct reasoning paths using prompt rephrasing to maximize diversity. Filtering mechanisms removed incomplete and off-topic solutions. GPT-4o then evaluated each step, assigning one of three labels: ✓ (correct—mathematically sound and contextually valid), ✗ (incorrect—containing logical, computational, or symbolic error), or ⊖ (neutral—vague, redundant, or non-contributing).

Labeling prompts included few-shot examples and structured rubrics to ensure annotation consistency. A dataset-wide rule applies: if any step in a solution is labeled ✗ and the final answer is incorrect, the entire solution set is marked incorrect.

Dataset Statistics

Table 3: Dataset Breakdown by Mathematical Topic

Topic	Problems	%
Linear Algebra	52,000	20%
Calculus	52,000	20%
Arithmetic	78,000	30%
Probability	26,000	10%
Polynomial Functions	26,000	10%
Measurement	26,000	10%

Error analysis reveals that solution accuracy declines sharply with problem complexity. Elementary-level problems show near-100% solution accuracy; Grade 12 introduces misapplied formulas and algebraic manipulation errors; university-level problems exhibit a 40% error rate (2 of 5 generated solutions contain errors), primarily due to incomplete logical justification and misapplication of advanced theorems.

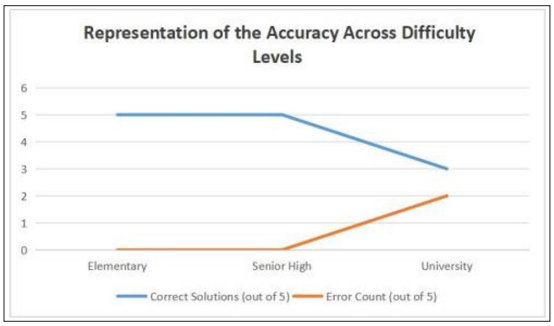


Figure 3: Accuracy Across v-math Difficulty Levels. University-Level Problems Show Approximately 40% Error Rate

Veri-Math: Architecture and Training Framework Overview

Veri-Math is a step-level discriminator built on the Qwen2.5-Math-1.5B-Instruct foundation. It evaluates the logical validity of individual reasoning steps within multi-step mathematical solutions, assigning correctness scores that enable verifier-guided decoding. Unlike post-hoc methods, Veri-Math interacts with the generation process in real time, scoring candidate steps and reweighting the LLM's decoding preferences toward logically coherent paths.

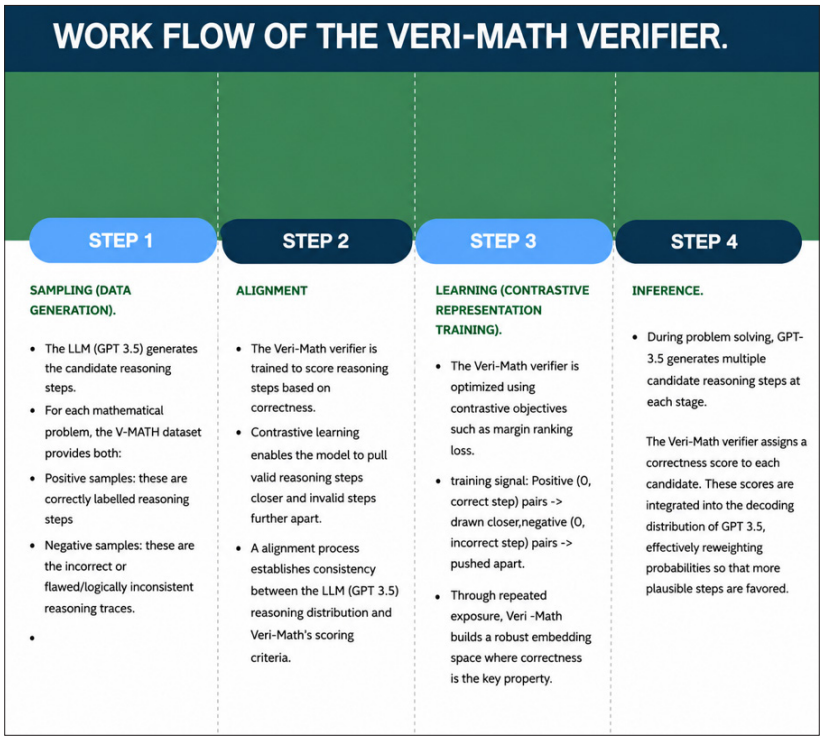


Figure 4: Veri-Math verifier workflow: candidate steps generated by LLM are scored and filtered

Table 4: Comparison of Decoding Strategies

Feature	Greedy	Self-Consist	Veri-Math
Step validation	✗	✗ Implicit	☑ Per-step
Error correction	✗	✗ Final only	☑ Iterative
Adaptivity	✗ Static	✗ Aggregate	☑ Dynamic

Scoring Mechanism

The verifier model $f\theta(q, s)$ takes a question q and candidate solution path s as input and outputs a scalar logit z. The sigmoid function converts this to a correctness probability $p = \sigma(f\theta(q, s)) \in [0,1]$. Scores close to 1 indicate logical validity; scores close to 0 indicate reasoning errors. For a complete solution of n steps, per-step scores are summed before applying the sigmoid: high totals (e.g., $5.0 \rightarrow \sigma \approx 0.993$) reflect confident correctness; negative totals (e.g., $-1.4 \rightarrow \sigma \approx 0.198$) indicate likely errors. The best candidate path is selected as $s^* = \text{argmax}_j\, p_j$.

Training Objectives

Training combines two objectives: contrastive loss (LNCE) and binary cross-entropy loss (LBCE) with a mixing weight α: $L = \alpha LNCE + (1-\alpha)\, LBCE$. The contrastive loss encourages correct reasoning paths to attain higher similarity to the question representation than incorrect ones. The binary loss supervises the verifier to assign probabilities close to 1 for correct and 0 for incorrect reasoning chains. This dual-objective approach enables both relational understanding of reasoning quality and calibrated probability estimation.

Training Configuration

The verifier was trained on 80% of V-MATH problems (stratified by tier) with 20% held for evaluation. GPT-3.5 generated up to 100 solution attempts per problem (maximum 5 steps each). Contrastive pairs consisted of one fully-correct path (all ✓ steps) and one negative path (at least one ✗ step). Chains with fewer than three steps or >50% neutral labels were discarded.

Table 5: Veri-Math Training Configuration

Parameter	Value
Optimizer	AdamW
Learning Rate	1×10^{-5}
Batch Size	16 contrastive pairs
Epochs	50
Loss Function	Margin-based contrastive
Dataset Split	80% train / 20% val

Results and Evaluation

OpenELM (1.1B), Microsoft Phi-2 (2.7B), and Google Gemma 2-2B (2B). Experiments ran on Google Colab Pro+ with NVIDIA A100 GPU (40GB VRAM). For each model and dataset, 1,000 problems were evaluated with the verifier frozen during evaluation to ensure generalization assessment.

Table 7: LLaMA 3-8B Results (Final Answer Accuracy %)

Dataset	Without Verifier	With Veri-Math	Gain
GSM8K	65.7%	78.7%	+13.0%
MathPile	67.4%	74.7%	+7.3%
PRM-800K	72.4%	82.3%	+9.9%
MATH	62.7%	72.4%	+9.7%

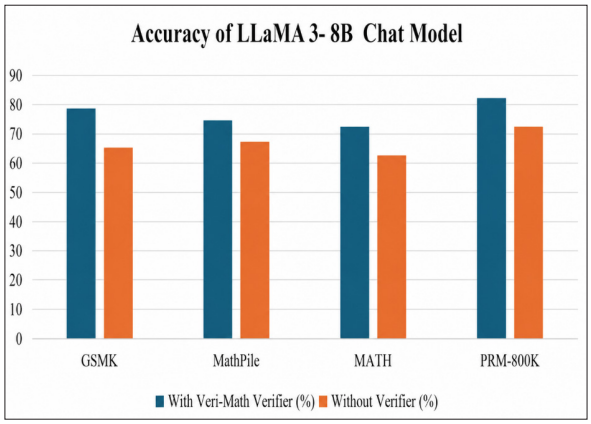


Figure 6: LLaMA 3-8B final answer accuracy with and without Veri-Math

Table 8: Qwen 2 Results (Final Answer Accuracy %)

Dataset	Without Verifier	With Veri-Math	Gain
GSM8K	60.5%	74.7%	+14.2%
MathPile	53.5%	72.2%	+18.7%
PRM-800K	62.4%	77.7%	+15.3%
MATH	65.8%	82.0%	+16.2%

Verifier Classification Accuracy

Evaluated on the V-MATH test set (13.7M+ annotated tokens), Veri-Math achieves 84.4% step-level classification accuracy. This result demonstrates strong generalization across the multi-tier dataset spanning elementary to university mathematics. Error analysis reveals category-specific recall: arithmetic errors (90.1%), sign/symbol errors (85.7%), equation manipulation errors (82.3%), logic jumps (79.4%), hallucinated reasoning (78.6%), and incorrect assumptions (71.4%).

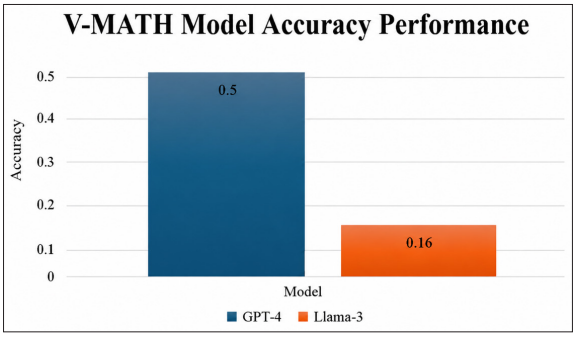


Figure 5: Veri-Math step classification accuracy on the V-MATH test set. Overall accuracy: 84.4%

Table 6: Accuracy Comparison Across Mathematical Topics (V-MATH)

Domain	LLM Baseline	Veri-Math	Gain
Algebra	72.3%	84.1%	+11.8%
Calculus	78.2%	89.4%	+11.2%
Arithmetic	79.1%	85.9%	+6.8%

Number Theory	73.5%	79.8%	+6.3%
Geometry	65.0%	69.7%	+4.7%

Impact on LLM Reasoning Performance

Five LLMs were evaluated across four benchmark datasets with and without the Veri-Math verifier: LLaMA 3-8B Chat (8B), Qwen 2 (1.5B)

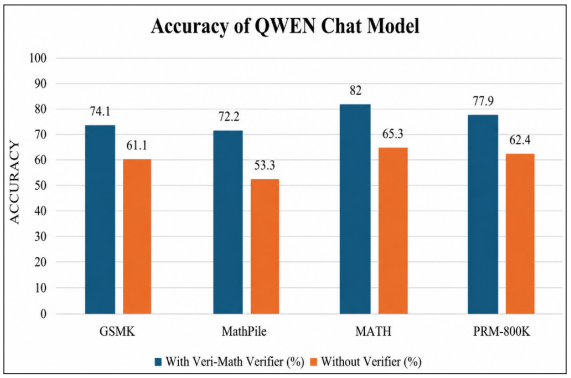


Figure 6: Qwen 2 final answer accuracy with and without Veri-Math

Gemma 2-2B demonstrated the largest single-dataset gain: +19.9% on PRM-800K (62.4% → 82.3%), consistent with the hypothesis that verifier-guided decoding provides the greatest relative benefit to models with lower baseline reasoning capacity. OpenELM showed gains of up to +12.9% on PRM-800K; Microsoft Phi-2 improved by up to +13.7% on MATH. Across all five models and four datasets, Veri-Math consistently improved final answer accuracy.

Table 9: Summary of Average Improvements (All Models, All Datasets)

Model	Params	Avg. Gain
LLaMA 3-8B	8B	+10.9%
Qwen 2	1.5B	+15.4%
OpenELM	1.1B	+10.7%
Microsoft Phi-2	2.7B	+11.6%
Google Gemma 2-2B	2B	+11.6%

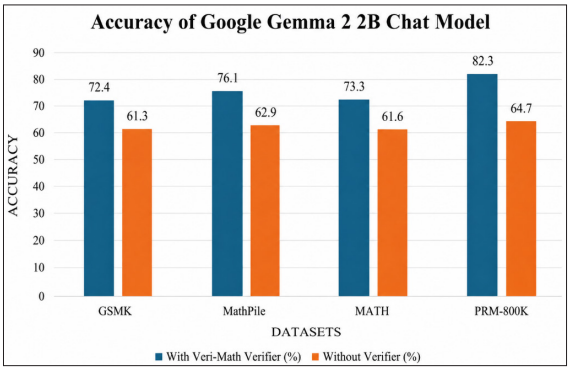


Figure 7: Gemma 2-2B accuracy with and without Veri-Math. Largest gain (+19.9%) on PRM-800K

Discussion

The 84.4% step classification accuracy demonstrates that Veri-Math generalizes well across diverse problem types and abstraction levels. The consistent improvement in final answer accuracy—across all five tested LLMs and four benchmark datasets—supports the central hypothesis: most LLM mathematical reasoning failures stem from insufficient intermediate validation, and a step-level verifier can systematically compensate for this gap by selecting the most coherent reasoning path from multiple candidates.

Notably, smaller models (Qwen 1.5B, OpenELM 1.1B) benefit more proportionally than larger ones, suggesting that the verifier acts as a reasoning amplifier that is particularly valuable for resource-constrained deployments. The verifier also fosters reasoning consistency over surface fluency—critical for symbolic tasks where syntactically correct steps can embed deep semantic flaws.

Limitations include sensitivity to GPT-4o annotation noise, potential over-filtering of valid but non-standard reasoning paths, and limited cross-domain evaluation beyond English mathematical content. Future work should explore uncertainty-aware training, multi-domain fine-tuning, and integration of the verifier directly into the decoding loop for real-time step-level guidance.

Conclusion

This paper presented V-MATH and Veri-Math: a large-scale step-annotated mathematical dataset and a contrastively trained step-level verification model. V-MATH provides 260,000+ problems with fine-grained ✓/✗/⊖ labels across elementary, secondary, and university mathematics, filling a critical gap in step-level supervision. Veri-Math leverages this dataset to discriminate between valid and flawed reasoning chains, achieving 84.4% step classification accuracy and improving final answer accuracy by 10–20% across five LLMs on four benchmarks.

Beyond benchmarks, the Veri-Math framework has direct applications in intelligent math tutoring (step-level formative feedback), formal theorem proving (hybrid proof verification), and compliance-critical reasoning systems (auditable intermediate-step validation). By elevating step-level correctness as a first-class evaluation criterion, this work establishes a foundation for more reliable, transparent, and interpretable AI reasoning systems.

References

1. Cobbe K, Kosaraju V, Bavarian M, Chen M, Jun H, et al. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168. 2021.
2. Hendrycks D, Burns C, Kadavath S, Arora A, Basart S, et al. Measuring mathematical problem solving with the math dataset. arXiv preprint arXiv:2103.03874. 2021.
3. Wei J, Wang X, Schuurmans D, Bosma M, Xia F, et al. Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems. 2022. 35: 24824-24837.
4. Wang X, Wei J, Schuurmans D, Le Q, Chi E, et al. Self-consistency improves chain of thought reasoning in language models. arXiv preprint arXiv:2203.11171. 2022.
5. Sun J, Zheng C, Xie E, Liu Z, Chu R, et al. A survey of reasoning with foundation models: Concepts, methodologies, and outlook. ACM Computing Surveys. 2025. 57: 1-43.

6. Amini A, Gabriel S, Lin S, Koncel-Kedziorski R, Choi Y et al. "MathQA: Towards Interpretable Math Word Problem Solving," ACL-NAACL. 2019. 1: 2357-2367.
7. Gaur V, Saunshi N. Reasoning in large language models through symbolic math word problems. InFindings of the Association for Computational Linguistics: ACL 2023. 2023. 5889-5903.
8. Saxton D. "Teaching Autoregressive Language Models Complex Tasks By Demonstration". 2021.
9. Besta M, Memedi F, Zhang Z, Gerstenberger R, Blach N. et al. "Demystifying Chains, Trees, and Graphs of Thoughts". 2024.
10. Yao S. "ReAct: Synergizing Reasoning and Acting in Language Models." 2023.
11. Shinn N, Cassano F, Gopinath A, Narasimhan K, Yao S. Reflexion: Language agents with verbal reinforcement learning. Advances in neural information processing systems. 2023. 36: 8634-8652.
12. Khalifa M, Logeswaran L, Lee M, Lee H, Wang L. Grace: Discriminator-guided chain-of-thought reasoning. InFindings of the Association for Computational Linguistics: EMNLP 2023. 2023 Dec. 15299-15328.
13. Li R, Luo Z, Du X. Fg-prm: Fine-grained hallucination detection and mitigation in language model mathematical reasoning. arXiv preprint arXiv:2410.06304. 2024.
14. Lightman H, Kosaraju V, Burda Y, Edwards H, Baker B, et al. Let's verify step by step. InInternational Conference on Learning Representations. 2024. 2024: 39578-39601.
15. Luo H, Sun Q, Xu C, Zhao P, Lou JG, et al. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. InInternational Conference on Learning Representations. 2025 2025: 49573-49609.
16. Yue X. "MAmmoTH: Mathematical Reasoning via Hybrid CoT and PoT Instruction Tuning". 2024.
17. Guo L. "DeepSeekMath: Enhancing Mathematical Reasoning via Group Relative Policy Optimization". 2024.
18. OpenAI. "o1 Technical Report," OpenAI. 2025.

Copyright: © 2026 Tang Shengbing, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.