Extended Thinking Time Improves Language Model Accuracy in Competition-Level Mathematics

Assignee Research

June 7, 2026

Abstract

This report synthesises findings from 4 peer-reviewed papers addressing the following research question: How does extended thinking time affect language model accuracy on competition-level mathematics v20. 16 claims were extracted from source literature; 0 were independently verified against retrieved documents. An automated multi-reviewer quality assessment produced a score of 2.8/10. This report is a machine-generated literature synthesis and does not constitute original research.

1 Introduction

This paper examines: Measuring Mathematical Problem Solving With the MATH Dataset. Research question: How does extended thinking time affect language model accuracy on competition-level mathematics v20.

2 Methodology

Systematic literature search across multiple databases yielded 4 papers. Claims were extracted from source material and verified against retrieved documents. An independent multi-reviewer assessment produced a quality score of 2.8/10.

3 Results

4 papers retrieved. 16 claims extracted; 0 independently verified. Quality review score: 2.8/10.

4 Limitations

This report is a machine-generated literature synthesis and does not constitute original research. Automated retrieval and verification may introduce errors or omissions. Review scores reflect automated assessment, not human peer review. Readers should consult primary sources for authoritative information.

5 Extracted Claims

Claim	Verified	Confidence
Accuracy remains low even for the best models on the MATH dataset.	×	0.14
Accuracy is increasing very slowly with model size on the MATH dataset.	×	0.08
Pretraining on AMPS enables a small 0.1B parameter model to perform similarly to a large fine-tuned 13B parameter model.	×	0.04
Having models generate their own step-by-step solutions before producing an answer actually degrades accuracy.	×	0.08
Providing partial ground truth step-by-step solutions can improve performance.	×	0.03
Providing models with step-by-step solutions at training time also increases accuracy.	×	0.05
GPT-2 models tokenize numbers so that one digit is processed at a time.	×	0.03
T5's tokenizer removes many LATEX symbols, leading to non-competitive performance.	×	0.01
Models pretrain on AMPS for one epoch using AdamW with a batch size of 128 and weight decay of 0.05.	×	0.01
During pretraining, Khan Academy data is upsampled by a factor of 5 and Mathematica data is downsampled by a factor of 2	×	0.01
Models are trained with 8 A100 GPUs, each requiring less than a day.	×	0.01
GPT-2 uses beam search with a beam size of 20 for final answer generation and a beam size of 10 for full step-by-step so	×	0.03
GPT-2 0.1B model achieves an average accuracy of 5.4% on the MATH dataset.	×	0.09
GPT-2 1.5B model achieves an average accuracy of 8.8% on the MATH dataset.	×	0.09
GPT-3 13B model achieves an average accuracy of 5.8% on the MATH dataset.	×	0.09
GPT-3 175B model achieves an average accuracy of 3.0% on the MATH dataset.	×	0.09

References

- http://arxiv.org/abs/2506.09162v1
- http://arxiv.org/abs/2305.17306v1
- http://arxiv.org/abs/2103.03874v2