
One Example Shown, Many Concepts Known!

Counterexample-Driven Conceptual Reasoning in Mathematical LLMs

Yinghui Li^{*1} Jiayi Kuang^{*2} Haojing Huang^{*1} Zhikun Xu^{*†34} Xinnian Liang⁵ Yi Yu³ Wenlian Lu³
Yangning Li¹⁶ Xiaoyu Tan^{†7} Chao Qu⁷ Ying Shen^{†28} Hai-Tao Zheng¹⁶ Philip S. Yu⁹

Abstract

Leveraging mathematical Large Language Models (LLMs) for proof generation is a fundamental topic in LLMs research. We argue that the ability of current LLMs to prove statements largely depends on whether they have encountered the relevant proof process during training. This reliance limits their deeper understanding of mathematical theorems and related concepts. Inspired by the pedagogical method of “proof by counterexamples” commonly used in human mathematics education, our work aims to enhance LLMs’ ability to conduct mathematical reasoning and proof through counterexamples. Specifically, we manually create a high-quality, university-level mathematical benchmark, COUNTERMATH, which requires LLMs to prove mathematical statements by providing counterexamples, thereby assessing their grasp of mathematical concepts. Additionally, we develop a data engineering framework to automatically obtain training data for further model improvement. Extensive experiments and detailed analyses demonstrate that COUNTERMATH is challenging, indicating that LLMs, such as OpenAI o1, have insufficient counterexample-driven proof capabilities. Moreover, our exploration into model training reveals that strengthening LLMs’ counterexample-driven conceptual reasoning abilities is crucial for improving their overall mathematical capabilities.

We believe that our work offers new perspectives on the community of mathematical LLMs.

1. Introduction

Mathematics, as a fundamental aspect of reasoning, has garnered significant research interest. Recent studies have demonstrated that Large Language Models (LLMs) exhibit strong mathematical reasoning abilities (OpenAI, 2023; Google, 2024; Yang et al., 2024; Shao et al., 2024; Ying et al., 2024; Chern et al., 2023; Luo et al., 2023; Yu et al., 2024a). Enhancing the mathematical reasoning capabilities of LLMs has become a prominent and fundamental topic within the LLMs research community.

Currently, there are two main paradigms for enhancing the mathematical reasoning capabilities of LLMs. *The first involves synthetic generation based on seed math questions* (Yu et al., 2023; Li et al., 2024a). For example, WizardMath (Luo et al., 2023) introduces a variety of math instructions to generate math questions of different complexities using GPT-3.5. *The second approach leverages formal mathematical languages to train LLM-based theorem provers*, such as Lean 4 (Moura & Ullrich, 2021). For instance, Draft-Sketch-Prove (Jiang et al., 2023), Hunyuan-Prover (Li et al., 2024c), and Lean-STaR (Lin et al., 2024a) interact with formal languages through informal proofs, automatic formalization, and natural language thoughts for theorem proving.

The two methods above enable LLMs to develop problem-solving skills either by training on massive similar problems, or by gaining proficiency through exposure to similar proof processes (Mirzadeh et al., 2024; Yu et al., 2024b). In both cases, these approaches enhance LLMs’ mathematical reasoning abilities through training, where proficiency is achieved through familiarity, akin to “drill-based” learning in human mathematics learning. However, relying solely on intensive-practice by inundating LLMs with math problems is neither sufficient nor essential for true mathematics learning. **In other words, drill-based learning alone does not foster a deep understanding of mathematical concepts in either humans or LLMs.**

^{*}Equal contribution.

¹Tsinghua University. E-mail: liyinghu20@mails.tsinghua.edu.cn

²Sun-Yat Sen University ³School of Mathematical Science, Fudan University ⁴ARC Lab, Arizona State University ⁵Bytedance Inc. ⁶Peng Cheng Laboratory ⁷INFLY TECH (Shanghai) Co., Ltd. ⁸Guangdong Provincial Key Laboratory of Fire Science and Intelligent Emergency Technology, ⁹University of Illinois Chicago. Correspondence to: Ying Shen, Xiaoyu Tan, Zhikun Xu <sheny76@mail.sysu.edu.cn, txywilliam1993@outlook.com, zhikunxu@asu.edu>.

Proceedings of the 42nd International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

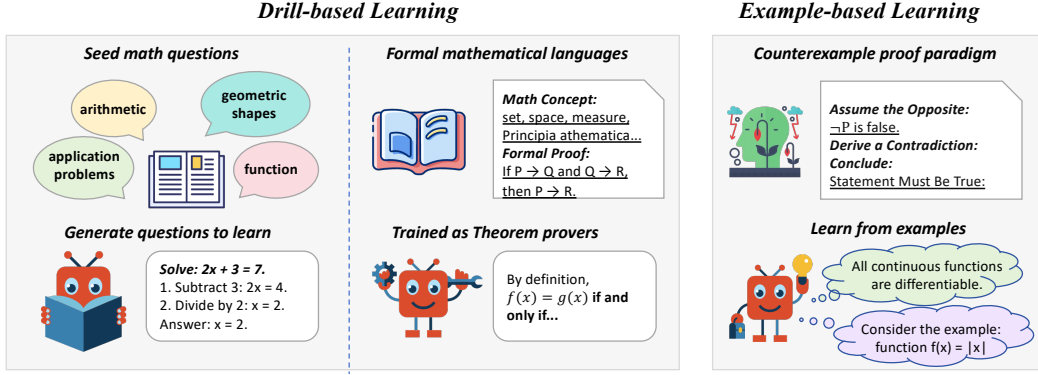


Figure 1. Comparison between drill-based learning and example-based learning. The first two math LLMs fail when confronted with advanced mathematics, and “Proving by examples” is a highly creative and concept-intensive mathematical skill.

As illustrated in Figure 1, for human mathematics learning, “example-based” learning is a more important strategy than drill-based learning. In particular, for mathematical proofs, “proof by counterexamples” is an indispensable approach. Inspired by the idea that counterexample-driven proofs better reflect a deep understanding of mathematical concepts, we propose COUNTERMATH¹, a counterexample-based mathematical reasoning benchmark. COUNTERMATH is designed to evaluate LLMs’ ability to distinguish subtle differences between mathematical terms and properties at university-level by providing examples. Specifically, we collect 1,216 statement-rationale pairs from mathematical textbooks, focusing on disproving certain statements under unusual conditions using counterexamples. In terms of difficulty, COUNTERMATH covers advanced mathematical knowledge similar to PutnamBench (Tsoukalas et al.) and Putnam-AXIOM (Gulati et al., 2024), both of which assess the depth of mathematical understanding in LLMs.

In addition to extensively evaluating various mainstream mathematics LLMs on COUNTERMATH, we also develop a framework for automatically acquiring counterexample-based mathematical reasoning data to enable further model training. Detailed analyses of both the evaluated LLMs and our trained LLMs reveal that:

- The contemporary LLMs including OpenAI o1 exhibit limited performance in determining whether a statement in COUNTERMATH is true or false, indicating significant room for improvement in higher-level mathematical conceptual reasoning.
- When analyzing the reasoning process of LLMs, many models still struggle with example-based reasoning. This demonstrates the limitations of drill-based learning and underscores the potential value of COUNTERMATH in advancing mathematical LLMs.

¹<https://github.com/THUKElab/COUNTERMATH>

- Lower performance is observed in topology and real analysis during our fine-grained evaluation, which indicates promising future research directions. Further studies on mathematical LLMs should explore these underrepresented areas of higher mathematics.
- Our fine-tuned model, trained with only 1,025 training samples, demonstrates strong performance on both our benchmark and OOD benchmarks. This confirms that learning counterexample-based reasoning is not only effective for our task but also has general significance for improving mathematical reasoning.

2. Related Work

Math Benchmarks. Recently, the number of math-related benchmarks has increased drastically (Amini et al., 2019; Yang & Deng, 2019; Zheng et al., 2022; Hendrycks et al., 2021; Cobbe et al., 2021; Frieder et al., 2023; Liu et al., 2024; He et al., 2024; Lu et al., 2024). The most influential ones are MATH (Hendrycks et al., 2021) and GSM8K (Cobbe et al., 2021), which focus on arithmetic reasoning at the high school competition level and grade school level, respectively. Moreover, other benchmarks such as MathBench (Liu et al., 2024) and OlympiadBench (He et al., 2024) are also blends of problems sets from various competitions and standard examinations, which are used to test human students’ abilities of utilizing the math knowledge and certain tricks to solve complex application-based problems. However, mathematicians are more expecting LLMs to help them in literature review, idea generation, proof-checking and collaborative writing as they focus on a broader spectrum of mathematical activities rather (Frieder et al., 2024). To better accommodate the true need for math research, some formal theorem proving benchmarks like PutnamBench (Tsoukalas et al.), CoqGym (Yang & Deng, 2019) and MiniF2F (Zheng et al., 2022) are also proposed recently in a combination of formal mathematical languages



Figure 2. Overview the construction process of COUNTERMATH. COUNTERMATH was first extracted from photocopied mathematical textbooks by crowd-sourced labelers with the OCR tool. For the next step, authors with bachelor degrees in applied mathematics as annotation experts would filter and correct improper statement-rationale pairs. Finally, GPT-4o was prompted to translate the validated data into English under experts’ supervision.

compilers (e.g. Coq, Lean), which could be viewed as the important math benchmarks for developing *Mathematics Mechanization* (Wu, 2001; Wu & Gao, 2007).

On the contrary, our benchmark COUNTERMATH focuses on conceptual reasoning among mathematical concepts and theorems. Specifically, we research certain math reasoning technique: *counterexamples in mathematics*, to check whether the models fully and correctly understand math concepts and theorems, which should be one of atomic abilities for what mathematician are expecting from LLMs compared to independently solving some simple math word problems.

Math Augmented LLMs. In contrast to general-purpose models such as GPT-4 (OpenAI, 2023) and Gemini (Google, 2024), several mathematics augmented LLMs have been developed using methods like data augmentation, pretraining, fine-tuning, and reinforcement learning with extensive mathematical corpora. For instance, WizardMath (Luo et al., 2023) employed tailored math prompts to generate seed data and then underwent RLHF and process supervision for training. Abel (Chern et al., 2023) utilized supervised fine-tuning with meticulous data processing, referred to as *Parental Oversight*. InternLM2-Math (Ying et al., 2024) enhanced mathematical reasoning with chain-of-thought (Wei et al., 2022), code interpreters, and Lean4 translation and theorem proving. NuminaMath (Li et al., 2024b), which recently secured first place in the Kaggle AIMO competition², leveraged *tool-integrated reasoning (TIR)* to generate math questions with fine-grained solutions. Qwen2.5-Math (Yang et al., 2024), initialized with general-purpose Qwen2.5 models, was trained on the undisclosed large-scale and high-quality mathematics-specific corpus. Deepseek-Math (Shao et al., 2024) focuses on data engineering during pretraining and efficient RL training.

²<https://www.kaggle.com/competitions/ai-mathematical-olympiad-prize/leaderboard>

Conceptual Reasoning. Conceptual reasoning is an ability to reason in abstract and high-level perspectives (Wang et al., 2024b; Huang et al., 2024; Li et al., 2024d; 2025). Recently, there are numerous studies where LLMs are reasoning on abstracted and conceptualized structures by analogy, deduction, induction, etc (Saparov et al., 2023; Li et al., 2023; Yasunaga et al., 2024; Xu et al., 2024; Wang et al., 2024a; Cheng et al., 2024; Zhou et al., 2024). Specifically in math, conceptual reasoning requires people to reason around math concepts and axioms at the play of math hypothesis, statements and problems (Simon, 2011). An example of this is ConceptMath (Wu et al., 2024), a math word problem benchmark in elementary school and middle school level, but the reasoning in solving these problems remains superficial as it just requires models to extract the correct variables and do basic arithmetic operations and it is also saturated with GPT models, which diminishes it from showing whether LLMs are truly mastering mathematics.

3. COUNTERMATH

3.1. Data Construction

Our dataset is constructed from a series of math textbooks focusing on counterexamples in different fields such as Algebra (Hu, 1983), Topology (Wang & Yang, 2000), Real Analysis (Wang, 1989) and Functional Analysis (Wang, 1994). **We have obtained the authors’ consent to use their publications solely for academic research purposes.** As the raw data sources are in Chinese, we also translate our dataset into English, creating a mathematical conceptual reasoning benchmark based on counterexamples, named as COUNTERMATH. Each data point includes a *statement*, *rationale*, *judgement* (i.e., whether the statement is True or False by its rationale), and *field*.

As illustrated in Figure 2, we first recruited several Chinese annotators from specific vendors to extract statement-rationale pairs from the aforementioned textbooks using an OCR tool, which yielded 1,274 statement-rationale pairs.

Next, the experts, among the authors, who have the bachelor’s degrees in applied mathematics manually checked all the data points from the previous stage, annotated each statement as True or False based on its rationale, and filtered out ambiguous pairs. This resulted in 1,216 data samples as the final version of our dataset. Additionally, since the data source was originally in Chinese, we prompted GPT-4o³ to translate the dataset into English. The authors then further validated the translated dataset to ensure its correctness and appropriateness. More details about the annotation process are provided in Appendix A.

3.2. Data Analysis

Overview Since the dataset has been constructed from textbooks in four different fields, the distribution of statement-rationale pairs is shown in Figure 3a. Moreover, the distribution of judgements is presented in Figure 3b. We observe that most statements are labeled as True. This may be because the data is sourced from mathematical textbooks, where most statements are phrased correctly to avoid misleading readers, especially novices in mathematics. In general, as shown in Figure 8, the statements often involve several college-level mathematical concepts or properties, focusing on nuanced understandings of mathematics. In real-world applications, due to the concise formulations, these statements are frequently used as interview questions in math-related graduate programs. This also contributes to **the diversity of mathematical testbeds for contemporary LLMs, fostering research in conceptual mathematical reasoning** (Klymchuk, 2010).

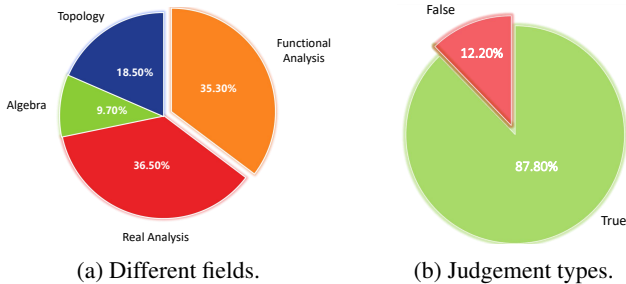


Figure 3. Data Distribution of COUNTERMATH.

Data Validation As mentioned in Section 3.1, the retention rate between the two annotation stages is 95.4%, demonstrating that most high-quality statement-rationale pairs were extracted from the textbooks. The filtered statement-rationale pairs often suffer from issues such as *irrelevance between statement and rationale*, *excessive typos*, and *trivial rationales*, such as *simply quoting conclusions from other statement-rationale pairs or even papers*.

³<https://openai.com/index/hello-gpt-4o/>

4. Benchmark Settings

Baselines We are testing following large language models with COUNTERMATH. For *open-weight models*, we use **Deepseek-Math-7B-RL** (Shao et al., 2024), **Eurus-2-7B-PRIME** (Cui et al., 2025), **Qwen2.5-Math-7B/72B-Instruct** (Yang et al., 2024), **NuminaMath-7B-TIR** (Li et al., 2024b), **InternLM2-Plus-7B/20B/Mixtral8x22B** (Ying et al., 2024), **Abel-7B/13B/70B** (Chern et al., 2023), **WizardMath-7B/70B** (Luo et al., 2023), **Mathstral-7B**⁴, **MetaMath-Mistral-7B** (Yu et al., 2023), **Xwin-Math-7B/13B/70B** (Li et al., 2024a), **Rho-math-7b-interpreter** (Lin et al., 2024b), **MAMmoTH2-7B/8x7B-Plus** (Yue et al., 2024), **QwQ-32B-Preview**⁵. For *proprietary models*, we use **GPT-4o**, **OpenAI o1**⁶, **Qwen-Max**⁷, **Deepseek-R1**⁸, **Claude3.7-sonnet**⁹, and **Gemini2.5-pro**¹⁰. The selection of baseline models tries to cover as many representative models as possible, which considers various perspectives, including data processing, training paradigms, base models, and developers (i.e., academia or industry). The detailed summary of open-weight baseline models is in Appendix B.

Prompts To maximize the performance of LLMs on COUNTERMATH, we use the default **CoT** prompts for each LLM, which are typically mentioned in their Huggingface model cards. These prompts include completion prompts, Alpaca-like prompts, and chat-template-based prompts. Additionally, we adopt a **Hint** prompt, designed to encourage LLMs to provide examples when attempting to solve problems. A summary of our used prompts is provided in Appendix B. Note that it is observed that using In-Context Learning (ICL) prompts do not significantly affect the performance of LLMs on COUNTERMATH. We believe this is due to the nature of counterexample-driven conceptual reasoning, where LLMs struggle to learn the ability to provide examples from a small number of demonstrations. In particular, mathematical subfields exhibit substantial variation in concepts and terminology. Therefore, we do not use ICL prompts in our experiments.

Evaluation Metrics Our evaluation metrics are two-fold. For efficiency, we use lexical matching such as *F1* to match

⁴<https://mistral.ai/news/mathstral/>

⁵<https://qwenlm.github.io/blog/qwq-32b-preview/>

⁶<https://cdn.openai.com/o1-system-card-20241205.pdf>

⁷<https://www.alibabacloud.com/help/en/model-studio/developer-reference/what-is-qwen-llm>

⁸<https://api-docs.deepseek.com/news/news250120>

⁹<https://www.anthropic.com/claude/sonnet>

¹⁰<https://deepmind.google/technologies/gemini/pro/>

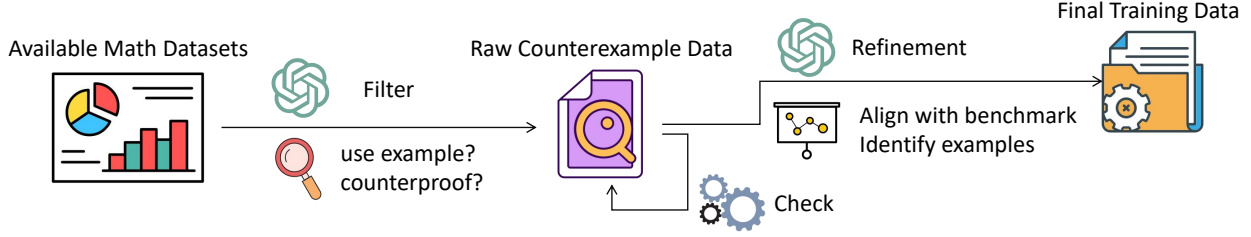


Figure 4. The overview of our training data engineering framework.

the judgements of the statements. The reason of not using *accuracy* is the imbalance of data distributions mentioned in Section 3.2. To assess whether the model has acquired the capability of solving mathematical problems through exemplification, we conducted a targeted evaluation in this section. We designed a systematic evaluation framework leveraging GPT-4o as an automated judge to perform the following tasks: **Example Extraction** automatically identifies and extracts instances where the model explicitly uses exemplification (i.e., generating or referencing specific counterexamples to justify the statement) during its reasoning process. **Alignment Assessment** evaluates whether each extracted example aligns with the reasoning approach of a predefined Reference Example in terms of logical structure, problem decomposition, and goal relevance. The evaluation prompts are detailed in Appendix B. We also designed three evaluation metrics to assess the models’ ability to solve problems by providing examples:

- **Proportion of Examples:** This metric calculates the proportion of problem-solving cases in which the model employs examples as part of its solution.
- **Strict Align:** This measures the percentage of the model’s provided examples that are fully consistent with the reference.
- **Loose Align:** This evaluates the proportion of instances where at least one example provided by the model aligns with the reference.

To further validate the reliability of the proposed LLM-based metrics, we conducted a comprehensive human evaluation. This evaluation focused primarily on two key aspects: (1) the accuracy of example extraction, and (2) the consistency of the evaluation results with human judgement. We selected 100 sample cases and manually reviewed them. First, the accuracy of the example extraction process, as performed by the model, was found to be 97%, demonstrating the robustness of our automatic extraction mechanism. Second, the evaluation of the extracted examples—assessed for alignment with reference examples—achieved an accuracy of 93.5% when compared to human judgements. These results indicate a strong alignment between the model’s automatic

evaluation process and human evaluators, confirming the reliability and validity of our methodology for assessing the use of exemplification in mathematical problem-solving.

5. Conceptual Finetuning

5.1. Training Data Engineering Framework

Filter-based Data Collection To validate our approach, we conduct supervised fine-tuning to enhance the model’s ability to provide examples for conceptual reasoning and incentivize its higher-level mathematical understanding. As shown in Figure 4, we propose an automatic training data engineering framework to obtain training data. Since most LLMs currently cannot provide satisfactory mathematical proofs based on examples, for data collection, we filter data from existing human-written datasets rather than directly generating counterexample data. Specifically, we apply a data filtering and refinement strategy based on strictly labeled datasets of propositional statements and proofs. We collect several high-quality mathematical proof datasets, such as ProofNet (Azerbayev et al., 2023) and NaturalProof (Welleck et al., 2021), ensuring that there is no overlap between these datasets and our COUNTERMATH. We use GPT-4o to filter data specifically involving proofs using counterexamples. Our designed data filtering prompt is presented in Appendix C. Inspired by AutoRace (Hao et al., 2024), which applies LLM as a judge, we design several criteria to evaluate whether the data utilize counter-reasoning, counter-examples for proof, or special examples. To maximize data retention, we employ a loose filtering criterion, retaining data if at least one criterion is met. Subsequently, we double-check the filtered data by assessing the completeness of statements, and the rigor and validity of proofs, while discarding incomplete or inconsistent entries. We ultimately obtain 1,025 filtered samples from an initial pool of over 30,000 data.

Training Data Refinement We further refine the rationale of the collected SFT data to improve the model’s ability to effectively provide examples during training. Compared with our COUNTERMATH, we observe that the SFT data feature longer proof processes and lack explicit illustrations of the

Table 1. Main evaluation results of various mainstream mathematical LLMs with the default CoT prompts on COUNTERMATH. The **Examples**, **Strict**, and **Loose** represent the three of our designed example-related evaluation metrics.

Models		Judgement F1 (macro)	Rationale Reasoning		
			Examples (%)	Strict (%)	Loose (%)
<i>Open source models</i>					
size = 7B	Deepseek-Math-7B-rl	32.2	65.9	18.9	20.6
	Eurus-2-7B-PRIME	37.5	64.8	28.5	32.0
	NuminaMath-7B-TIR	30.4	54.1	13.0	13.7
	InternLM2-Math-Plus-7B	33.9	36.6	9.0	9.5
	Abel-7B-002	34.4	66.1	16.0	17.9
	WizardMath-7B-v1.1	27.9	43.2	6.4	7.2
	Mathstral-7B-v0.1	28.2	38.9	7.5	7.9
	MetaMath-Mistral-7B	31.0	26.5	0.4	0.7
	Xwin-Math-7B-V1.0	28.1	31.3	1.2	1.7
	rho-math-7b-interpreter-v0.1	22.3	18.3	1.9	2.1
	MAmmoTH2-7B-Plus	32.3	54.2	10.7	12.1
	Qwen2.5-Math-7B-Instruct	38.3	74.2	30.2	33.2
7B < size < 70B	Abel-13B-001	22.4	24.4	0.8	0.8
	Xwin-Math-13B-V1.0	30.2	31.3	1.2	1.7
	InternLM2-Math-Plus-20B	18.4	28.8	8.4	9.5
	MAmmoTH2-8x7B-Plus	28.8	51.4	14.1	15.5
	QwQ-32B-Preview	39.9	70.0	38.6	43.8
size ≥ 70B	InternLM2-Math-Plus-Mixtral8x22B	37.3	63.2	21.5	23.1
	Xwin-Math-70B-V1.0	25.5	25.2	1.4	1.7
	Abel-70B-001	31.0	48.4	5.3	6.1
	WizardMath-70B-v1.0	24.2	52.9	6.3	7.4
	Qwen2.5-Math-72B-Instruct	41.8	76.6	38.9	41.6
<i>Commercial models</i>					
GPT-4o		59.0	44.7	19.7	21.3
OpenAI o1-preview		60.1	55.8	39.8	40.9
Qwen-max		58.9	61.8	30.4	33.9
Deepseek-R1		80.7	86.8	54.2	65.3
Claude3.7-sonnet		64.8	78.0	45.0	52.5
Gemini2.5-pro		77.0	90.8	65.1	75.7

provided examples. To address this, we employ GPT-4o to refine the rationales to better align with the feature distribution of COUNTERMATH. Specifically, we randomly select one example from each of the four fields in our COUNTERMATH as a reference, and provide three manually rewritten before-and-after comparisons to guide the model. Our designed data refinement prompt is presented in Appendix C. We emphasize that while reducing proof length, the rewriting process should preserve the original reasoning structure to maintain the integrity of the data. This refinement ensures that the SFT data closely aligns with the characteristics of COUNTERMATH and explicitly incorporates example usage to facilitate training.

5.2. Training and Evaluation Details

For model training, we select Qwen-2.5-Math-7B-Instruct, an open-source model known for its strong mathematical reasoning capabilities and general applicability. we fine-

tune Qwen2.5-Math-Instruct-7B using supervised LoRA training on $2 \times L20$ 48GB GPUs, with a learning rate of $1.0e-5$. After training, we evaluate the model on both our COUNTERMATH and several out-of-distribution (OOD) benchmarks, such as MATH (Hendrycks et al., 2021) and GSM8K (Cobbe et al., 2021), OlympiadBench (He et al., 2024), MMLU (Wang et al., 2024c), AIME 2024. We aim to demonstrate not only improved performance on the COUNTERMATH but also enhanced generalization to OOD benchmarks, thereby validating the effectiveness of our insight in improving the model’s overall conceptual reasoning capabilities by examples.

6. Analysis and Discussions

6.1. Evaluation Results without Finetuning

We selected a range of advanced mathematical LLMs with varying parameter sizes to evaluate their conceptual reason-

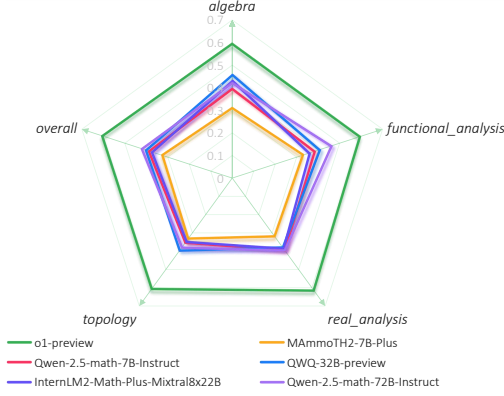


Figure 5. Fine-grained evaluation results of different fields in COUNTERMATH.

ing abilities on our benchmark. Table 1 summarizes their performance across various metrics. From the results, we derive the following findings and insights:

Judgement Performance The performance on the automatic evaluation metric F1 reflects the models’ fundamental conceptual reasoning abilities, specifically their capacity to correctly determine the truth or falsehood of a given statement. While open-source models exhibit some performance, their overall performance is relatively low around 30. Even the advanced Qwen-2.5-Math-72B-Instruct achieves only 41.8, falling behind commercial models. Notably, the math-reasoning-specific Deepseek-R1 model outperforms others, achieving an F1 score of 80.7. However, compared to benchmarks focused on elementary and high school mathematics, the overall performance of commercial models is relatively lower. This disparity underscores the challenges posed by our higher-level mathematics benchmark, aligning with our objective of exploring deeper conceptual reasoning. When comparing the performance of models of different sizes, we find that some small 7B models outperform the larger 72B model. For example, Deepseek-Math-7B-rl outperforms WizardMath-70B-v1.0. We believe this phenomenon is related to the training methods behind these models. As explained in Section 1, drill-based training in WizardMath-70B-v1.0 is not very effective in improving the ability to understand deep and complex mathematical concepts. In contrast, the superior performance of Deepseek-Math-7B-rl suggests that reinforcement learning may play a key role in enhancing counterexample-driven conceptual reasoning.

Conceptual Reasoning Ability with Examples Beyond assessing binary classification F1 in judgements, we also evaluate the reasoning process of the models. We hope that LLMs provide logical proofs or counterexamples rather than relying on random guessing or copying statements. The Qwen-series models demonstrate superior higher math-

ematics reasoning, accurately identifying statements suited for counterexample reasoning and surpassing even the commercial o1 model. However, many open-source models fail to generate meaningful examples. For instance, MetaMath and rho produce counterexamples in only 26.5% and 18.3% of cases, respectively, with consistency rates as low as 0.4% and 1.9%. This limitation likely originates from the “Practice lots of math problems” training strategies employed, where most data are derived from elementary and high school mathematics. Such training leaves these models ill-equipped to handle the abstraction and conceptual reasoning demands of higher-level mathematics.

Fine-grained Analysis Across Fields We further conduct a fine-grained analysis of four mathematical fields in COUNTERMATH, evaluating six advanced models of varying sizes. As shown in Figure 5, the o1 model consistently outperforms others across all fields, with relatively balanced performance in each. Among open-source models, we observe stronger performance in algebra and functional analysis but weaker results in topology and real analysis. This suggests that topology and real analysis pose greater challenges for LLMs, possibly due to their lower occurrence in training data. These findings highlight the need for future research to focus on developing LLMs capable of handling these underexplored fields of mathematical reasoning.

6.2. Results with Finetuning

Evaluation on Our Benchmark We evaluate the approach on our benchmark to verify whether counterexample reasoning effectively enhances a model’s conceptual reasoning capabilities. In addition to the base model, we include comparisons with the hint prompt, where explicit hints encouraging LLM to reason by example were provided. For a fair comparison, we use identical prompts to generate outputs during the experiments. The results indicate that, with just 1,025 training samples, the fine-tuned model outperforms base models in F1 score. Furthermore, the trained model demonstrates superior example-based conceptual reasoning abilities, showing improvements in both the quantity and quality of examples compared to base models. On the other hand, although the constructed training data are refined to align closely with our COUNTERMATH distribution, some discrepancies remain. Since our exploration involves only the simple SFT strategy on a limited dataset, the model performs slightly worse on some metrics, which can be considered an acceptable limitation.

Evaluation on Out-of-Distribution Benchmarks To assess the generalizability of the fine-tuned model, we further evaluate its performance on out-of-distribution (OOD) benchmarks. This aims to verify whether the model’s counterexample reasoning capability, which is valid on our

Table 2. The evaluation results on our COUNTERMATH.

Models	F1 (macro)	Examples(%)	Strict(%)	Loose(%)
<i>Base models</i>				
Qwen2.5-Math-7B-Instruct	38.3	74.2	30.2	33.2
Qwen2.5-Math-7B-Instruct + Hint prompt	39.4	79.0	33.1	36.4
<i>Our training model</i>				
Qwen2.5-Math-7B-Instruct-SFT	39.7	75.2	31.4	34.7
Qwen2.5-Math-7B-Instruct-SFT + Hint prompt	41.1	79.4	31.1	34.7

Table 3. The Out-of-distribution Evaluation Results.

Models	GSM8K	MATH	Olympaidbench-Math	MMLU-college-Math	AIME 2024
Qwen2.5-math-7B-Instruct	95.1	80.5	41.6	74.0	20.0
Qwen2.5-math-72B-Instruct	95.4	84.9	49.2	78.3	26.7
Qwen2.5-math-7B-Instruct +Countermath-SFT	95.6	87.9	46.4	80.0	30.0

COUNTERMATH, can transfer to other benchmarks and deliver broader performance improvements. Using identical prompts and configurations for fairness, we compare the base models and fine-tuned models on OOD benchmarks. The results reveal that the fine-tuned model outperforms the base models on both benchmarks, even surpassing larger models like the 72B-parameter model. This aligns well with our hypothesis: equipping the model with counterexample reasoning ability can enhance conceptual reasoning across the general mathematical domain.

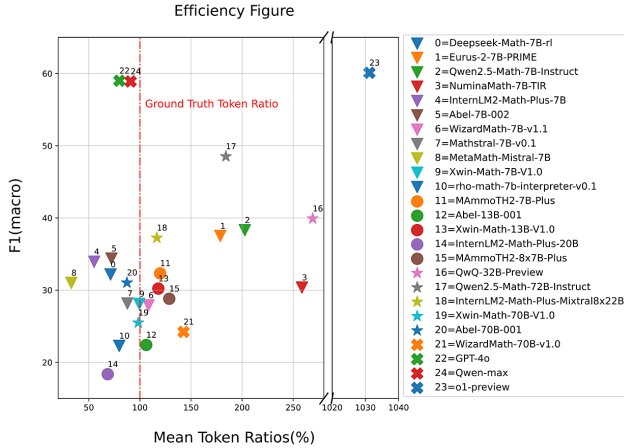


Figure 6. The relationship between Mean Token Ratios (%) and F1 (macro) scores for various models. The red dashed line represents the Ground Truth Token Ratio (100%), serving as an efficiency benchmark. Models closer to this line are more token-efficient, while those farther to the right consume significantly more tokens.

6.3. Used Tokens Analysis

The relationship between model performance and token usage efficiency is a critical factor in understanding the trade-offs inherent in model design. As depicted in Figure 6, we analyzed the connection between the mean token ratios (%), which represent the efficiency of token usage relative to ground truth, and the F1 (macro) score, which reflects the predictive performance of the model. The Mean Token Ratio (%) is calculated by dividing the number of tokens actually used by the model during inference by the number of tokens in the ground truth answer. Reasoning models such as "23=o1-preview" and "16=QwQ-32B-Preview" utilize a significantly large number of tokens during inference, but this extensive token consumption does not lead to a corresponding improvement in F1 (macro) scores. This suggests that the benchmark task is highly difficult, where simply increasing the length or detail of token reasoning does not necessarily enhance performance. Models like "24=Qwen-max" and "22=GPT-4o" demonstrate a commendable balance between token usage and performance. These models achieve relatively high F1 (macro) scores while maintaining token consumption close to the ground truth token ratio. This indicates their ability to perform accurate reasoning efficiently, without overly relying on additional token usage.

6.4. Case Study

In our case study, we observe that while the LLM correctly judges mathematical statements, it sometimes provides incorrect examples as shown in Figure 7, Case 1. In addition, we verify the validity of our evaluation metric of EXAMPLE. The LLM generates valid counterexamples that dif-

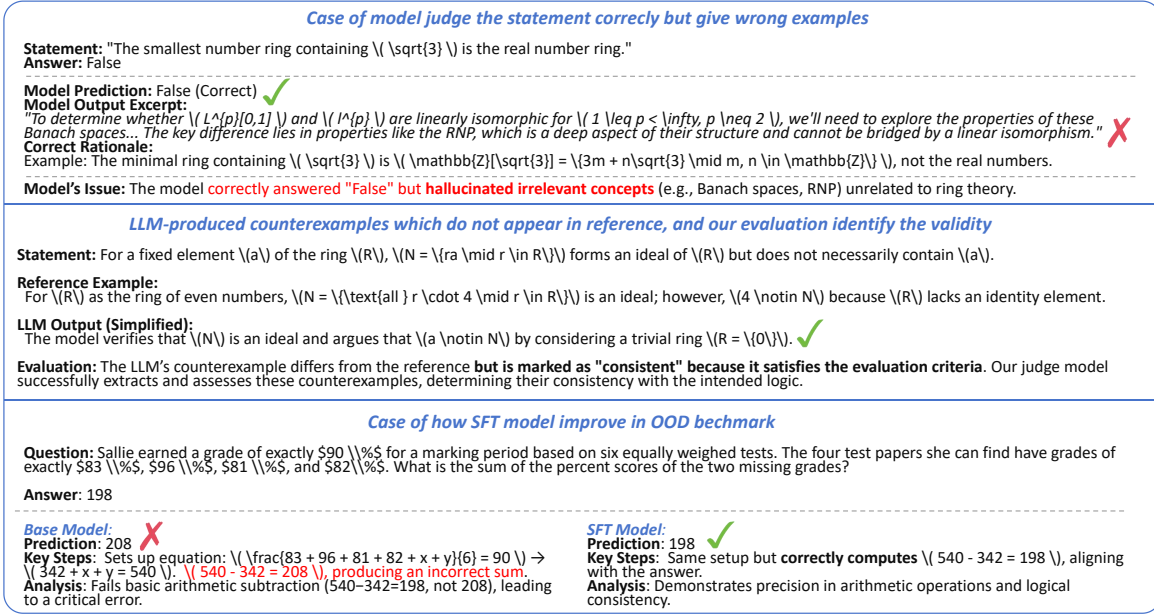


Figure 7. Our case studies of the performance of LLM include: (1) correct judgment with incorrect examples, (2) validation of novel but effective counterexamples, and (3) performance improvement on MATH dataset after counterexample training

fers from reference ones, yet are correctly validated by our judge model, highlighting the effectiveness of our evaluation method. Finally, we compare the improvement of MATH benchmark between base and SFT model, demonstrating the benefit of incorporating counterexample training to enhance mathematical reasoning capabilities.

7. Conclusion

In this work, we address the limitations of drill-based learning in mathematical LLMs by introducing COUNTERMATH, a counterexample-based reasoning benchmark. Unlike existing datasets, COUNTERMATH evaluates models on their ability to distinguish nuanced mathematical concepts through example-driven reasoning. Our key contributions include constructing a high-quality dataset with 1,216 university-level counterexample-based proofs, benchmarking state-of-the-art mathematical LLMs to reveal their conceptual reasoning gaps, and developing an automated framework for counterexample data generation and training. Experimental results show that current LLMs struggle with counterexample-based reasoning, particularly in topology and real analysis, highlighting areas for future research. Furthermore, our fine-tuned model, trained on only 1,025 examples, significantly outperforms baseline models, demonstrating the effectiveness and generalizability of counterexample-driven learning in mathematical reasoning.

Acknowledgements

This research is supported by National Natural Science Foundation of China (Grant No. 62276154), Research Center for Computer Network (Shenzhen) Ministry of Education, the Natural Science Foundation of Guangdong Province (Grant No. 2023A1515012914 and 440300241033100801770), Basic Research Fund of Shenzhen City (Grant No. JCYJ20210324120012033, JCYJ20240813112009013 and GJHZ20240218113603006), the Major Key Project of PCL (NO. PCL2024A08), the Key-Area Research and Development Program of Guangdong Province No.2024B1111100001. This work is supported in part by NSF under grants III-2106758, and POSE-2346158.

Impact Statement

This paper introduces a conceptual mathematical benchmark aimed at advancing research on Large Language Models (LLMs) in the realm of genuine mathematical reasoning. The dataset utilized in this work is derived from publications with copyrights reserved and authors' permission only for academic research purposes. It is important to acknowledge that our experiments and evaluations rely heavily on LLMs, but this study does not fully explore or mitigate potential biases inherent in their outputs. Addressing these biases and ensuring model alignment with social values remain critical challenges. This underscores the importance of conducting comprehensive evaluations that consider diverse dimensions of human society and their implications.

References

- Amini, A., Gabriel, S., Lin, P., Koncel-Kedziorski, R., Choi, Y., and Hajishirzi, H. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. *arXiv preprint arXiv:1905.13319*, 2019.
- Azerbaiyev, Z., Piotrowski, B., Schoelkopf, H., Ayers, E. W., Radev, D., and Avigad, J. Proofnet: Autoformalizing and formally proving undergraduate-level mathematics. *CoRR*, abs/2302.12433, 2023. doi: 10.48550/ARXIV.2302.12433.
- Chen, W., Ma, X., Wang, X., and Cohen, W. W. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Transactions on Machine Learning Research*, 2023.
- Cheng, K., Yang, J., Jiang, H., Wang, Z., Huang, B., Li, R., Li, S., Li, Z., Gao, Y., Li, X., et al. Inductive or deductive? rethinking the fundamental reasoning abilities of llms. *arXiv preprint arXiv:2408.00114*, 2024.
- Chern, E., Zou, H., Li, X., Hu, J., Feng, K., Li, J., and Liu, P. Generative ai for math: Abel. <https://github.com/GAIR-NLP/abel>, 2023.
- Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.
- Cui, G., Yuan, L., Wang, Z., Wang, H., Li, W., He, B., Fan, Y., Yu, T., Xu, Q., Chen, W., Yuan, J., Chen, H., Zhang, K., Lv, X., Wang, S., Yao, Y., Peng, H., Cheng, Y., Liu, Z., Sun, M., Zhou, B., and Ding, N. Process reinforcement through implicit rewards, 2025.
- Frieder, S., Pinchetti, L., Chevalier, A., Griffiths, R.-R., Salvatori, T., Lukasiewicz, T., Petersen, P. C., and Berner, J. Mathematical capabilities of chatgpt, 2023.
- Frieder, S., Berner, J., Petersen, P., and Lukasiewicz, T. Large language models for mathematicians, 2024.
- Google, G. T. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of contexts, 2024.
- Gou, Z., Shao, Z., Gong, Y., Yang, Y., Huang, M., Duan, N., Chen, W., et al. Tora: A tool-integrated reasoning agent for mathematical problem solving. In *The Twelfth International Conference on Learning Representations*, 2024.
- Gulati, A., Miranda, B., Chen, E., Xia, E., Fronsda, K., de Moraes Dumont, B., and Koyejo, S. Putnam-AXIOM: A functional and static benchmark for measuring higher level mathematical reasoning. In *The 4th Workshop on Mathematical Reasoning and AI at NeurIPS’24*, 2024.
- Hao, S., Gu, Y., Luo, H., Liu, T., Shao, X., Wang, X., Xie, S., Ma, H., Samavedhi, A., Gao, Q., et al. Llm reasoners: New evaluation, library, and analysis of step-by-step reasoning with large language models. *arXiv preprint arXiv:2404.05221*, 2024.
- He, C., Luo, R., Bai, Y., Hu, S., Thai, Z. L., Shen, J., Hu, J., Han, X., Huang, Y., Zhang, Y., Liu, J., Qi, L., Liu, Z., and Sun, M. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems, 2024.
- Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D., and Steinhardt, J. Measuring mathematical problem solving with the math dataset. *NeurIPS*, 2021.
- Hu, C. *Counterexamples in Algebra*. Shaanxi Science and Technology Press, 1983.
- Huang, S., Ma, S., Li, Y., Huang, M., Zou, W., Zhang, W., and Zheng, H. Lateval: An interactive llms evaluation benchmark with incomplete information from lateral thinking puzzles. In Calzolari, N., Kan, M., Hoste, V., Lenci, A., Sakti, S., and Xue, N. (eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pp. 10186–10197. ELRA and ICCL, 2024. URL <https://aclanthology.org/2024.lrec-main.889>.
- Jiang, A. Q., Welleck, S., Zhou, J. P., Lacroix, T., Liu, J., Li, W., Jamnik, M., Lample, G., and Wu, Y. Draft, sketch, and prove: Guiding formal theorem provers with informal proofs. In *The Eleventh International Conference on Learning Representations*, 2023.
- Klymchuk, S. *Counterexamples in Calculus*. 01 2010. ISBN 9780883857656. doi: 10.1090/clrm/034.
- Li, C., Wang, W., Hu, J., Wei, Y., Zheng, N., Hu, H., Zhang, Z., and Peng, H. Common 7b language models already possess strong math capabilities. *arXiv preprint arXiv:2403.04706*, 2024a.
- Li, J., Beeching, E., Tunstall, L., Lipkin, B., Soletskyi, R., Huang, S., Rasul, K., Yu, L., Jiang, A. Q., Shen, Z., et al. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. *Hugging Face repository*, 13:9, 2024b.
- Li, Y., Xu, Z., Chen, S., Huang, H., Li, Y., Jiang, Y., Li, Z., Zhou, Q., Zheng, H.-T., and Shen, Y. Towards real-world writing assistance: A chinese character checking benchmark with faked and misspelled characters. *arXiv preprint arXiv:2311.11268*, 2023.

- Li, Y., Du, D., Song, L., Li, C., Wang, W., Yang, T., and Mi, H. Hunyuanprover: A scalable data synthesis framework and guided tree search for automated theorem proving. *arXiv preprint arXiv:2412.20735*, 2024c.
- Li, Y., Zhou, Q., Luo, Y., Ma, S., Li, Y., Zheng, H.-T., Hu, X., and Philip, S. Y. When llms meet cunning texts: A fallacy understanding benchmark for large language models. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024d.
- Li, Y., Ma, S., Chen, S., Huang, H., Huang, S., Li, Y., Zheng, H.-T., and Shen, Y. Correct like humans: Progressive learning framework for chinese text error correction. *Expert Systems with Applications*, 265:126039, 2025.
- Lin, H., Sun, Z., Yang, Y., and Welleck, S. Lean-star: Learning to interleave thinking and proving. *arXiv preprint arXiv:2407.10040*, 2024a.
- Lin, Z., Gou, Z., Gong, Y., Liu, X., Shen, Y., Xu, R., Lin, C., Yang, Y., Jiao, J., Duan, N., et al. Rho-1: Not all tokens are what you need. *arXiv preprint arXiv:2404.07965*, 2024b.
- Liu, H., Zheng, Z., Qiao, Y., Duan, H., Fei, Z., Zhou, F., Zhang, W., Zhang, S., Lin, D., and Chen, K. Mathbench: Evaluating the theory and application proficiency of llms with a hierarchical mathematics benchmark, 2024.
- Lu, P., Bansal, H., Xia, T., Liu, J., Li, C., Hajishirzi, H., Cheng, H., Chang, K.-W., Galley, M., and Gao, J. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*, 2024.
- Luo, H., Sun, Q., Xu, C., Zhao, P., Lou, J., Tao, C., Geng, X., Lin, Q., Chen, S., and Zhang, D. Wizard-math: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *arXiv preprint arXiv:2308.09583*, 2023.
- Mirzadeh, I., Alizadeh, K., Shahrokhi, H., Tuzel, O., Bengio, S., and Farajtabar, M. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models. *arXiv preprint arXiv:2410.05229*, 2024.
- Moura, L. d. and Ullrich, S. The lean 4 theorem prover and programming language. In Platzer, A. and Sutcliffe, G. (eds.), *Automated Deduction – CADE 28*, pp. 625–635, Cham, 2021. Springer International Publishing. ISBN 978-3-030-79876-5.
- OpenAI. Gpt-4 technical report, 2023.
- Saparov, A., Pang, R. Y., Padmakumar, V., Joshi, N., Kazemi, M., Kim, N., and He, H. Testing the general deductive reasoning capacity of large language models using OOD examples. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.
- Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.
- Simon, M. A. Studying mathematics conceptual learning: Student learning through their mathematical activity. *North American Chapter of the International Group for the Psychology of Mathematics Education*, 2011.
- Tsoukalas, G., Lee, J., Jennings, J., Xin, J., Ding, M., Jennings, M., Thakur, A., and Chaudhuri, S. Putnambench: Evaluating neural theorem-provers on the putnam mathematical competition. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- Wang, L. *Counterexamples in Real Analysis*. Higher Education Press, 1989.
- Wang, L. *Counterexamples in Functional Analysis*. Higher Education Press, 1994.
- Wang, L. and Yang, F. *Counterexamples in Topology*. Science Press, 2000.
- Wang, R., Zelikman, E., Poesia, G., Pu, Y., Haber, N., and Goodman, N. Hypothesis search: Inductive reasoning with language models. In *The Twelfth International Conference on Learning Representations*, 2024a.
- Wang, W., Fang, T., Shi, H., Xu, B., Ding, W., Zhang, L., Fan, W., Bai, J., Li, H., Liu, X., et al. On the role of entity and event level conceptualization in generalizable reasoning: A survey of tasks, methods, applications, and future directions. *arXiv preprint arXiv:2406.10885*, 2024b.
- Wang, Y., Ma, X., Zhang, G., Ni, Y., Chandra, A., Guo, S., Ren, W., Arulraj, A., He, X., Jiang, Z., et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290, 2024c.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.
- Welleck, S., Liu, J., Bras, R. L., Hajishirzi, H., Choi, Y., and Cho, K. Naturalproofs: Mathematical theorem proving in natural language. In Vanschoren, J. and Yeung, S. (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS*

- Datasets and Benchmarks 2021, December 2021, virtual*, 2021.
- Wu, W. and Gao, X.-S. Mathematics mechanization and applications after thirty years. *Frontiers of Computer Science in China*, 1:1–8, 02 2007. doi: 10.1007/s11704-007-0001-8.
- Wu, W.-t. *Mathematics Mechanization: Mechanical Geometry Theorem-Proving, Mechanical Geometry Problem-Solving and Polynomial Equations-Solving*. Mathematics and Its Applications. Springer Netherlands, 2001. ISBN 9780792358350.
- Wu, Y., Liu, J., Bu, X., Liu, J., Zhou, Z., Zhang, Y., Zhang, C., Bai, Z., Chen, H., Ge, T., et al. Conceptmath: A bilingual concept-wise benchmark for measuring mathematical reasoning of large language models. *arXiv preprint arXiv:2402.14660*, 2024.
- Xu, Z., Li, Y., Ding, R., Wang, X., Chen, B., Jiang, Y., Deng, X., Ma, J., Zheng, H.-T., Lu, W., et al. Let llms take on the latest challenges! a chinese dynamic question answering benchmark. *arXiv preprint arXiv:2402.19248*, 2024.
- Yang, A., Yang, B., Hui, B., Zheng, B., Yu, B., Zhou, C., Li, C., Li, C., Liu, D., Huang, F., et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.
- Yang, K. and Deng, J. Learning to prove theorems via interacting with proof assistants. In *International Conference on Machine Learning*, pp. 6984–6994. PMLR, 2019.
- Yasunaga, M., Chen, X., Li, Y., Pasupat, P., Leskovec, J., Liang, P., Chi, E. H., and Zhou, D. Large language models as analogical reasoners. In *The Twelfth International Conference on Learning Representations*, 2024.
- Ying, H., Zhang, S., Li, L., Zhou, Z., Shao, Y., Fei, Z., Ma, Y., Hong, J., Liu, K., Wang, Z., et al. Internlm-math: Open math large language models toward verifiable reasoning. *arXiv preprint arXiv:2402.06332*, 2024.
- Yu, L., Jiang, W., Shi, H., Yu, J., Liu, Z., Zhang, Y., Kwok, J. T., Li, Z., Weller, A., and Liu, W. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*, 2023.
- Yu, L., Jiang, W., Shi, H., Jincheng, Y., Liu, Z., Zhang, Y., Kwok, J., Li, Z., Weller, A., and Liu, W. Metamath: Bootstrap your own mathematical questions for large language models. In *The Twelfth International Conference on Learning Representations*, 2024a.
- Yu, X., Zhou, B., Cheng, H., and Roth, D. Reasonagain: Using extractable symbolic programs to evaluate mathematical reasoning. *arXiv preprint arXiv:2410.19056*, 2024b.
- Yue, X., Zheng, T., Zhang, G., and Chen, W. Mammoth2: Scaling instructions from the web. *arXiv preprint arXiv:2405.03548*, 2024.
- Zheng, K., Han, J. M., and Polu, S. minif2f: a cross-system benchmark for formal olympiad-level mathematics. In *International Conference on Learning Representations*, 2022.
- Zhou, B., Zhang, H., Chen, S., Yu, D., Wang, H., Peng, B., Roth, D., and Yu, D. Conceptual and unbiased reasoning in language models. *arXiv preprint arXiv:2404.00205*, 2024.

A. Details for Data Curation

Annotation Cost In the first-stage annotation, there were three annotators recruited who have at least bachelor’s degrees in engineering or science-related majors. On average, each of them has annotated about 416 statement-rationale pairs from the given textbooks. They were working through the vendor’s provided platform with the provided OCR tool and annotation examples. The cost for the first-stage crowd-sourced annotation was about \$1306.

Annotation Process The annotation process consists of two stages. In the first stage, the recruited annotator were asked to annotate `statement`, `rationale` (i.e. the response for supporting the statement, typically examples or counterexamples), `field` (i.e. algebra, topology, real analysis or functional analysis), and `text` (i.e. mappings to original annotations for the validation use). During annotation, they were provided with several annotation examples created by authors, demonstrating the annotation targets. Moreover, we also asked annotators to focus on statements related to *proving or disproving* and *existence of certain mathematical objects* and ignore those statements without clear or complete answers such as definitions for certain advanced and complicated concepts. For the second stage, authors with at least bachelor’s degrees in applied mathematics were checking the LaTeX formats and typos in statements and rationales, keeping the rationales that correct and clearly support or deny the statements with examples or counterexamples. Besides, we also added another element in annotated data points, `judgement`, to show whether the statement is true or false by its rationale. One full annotation example is shown in Figure 8. **It should be noted that we have modified some statements (less than 5%), which are easily revised to be the reverse, to make them False for diversity because nearly all statements are stated as True in the original textbooks.** In all, we tried to make sure that all statements and rationales were concise and related.

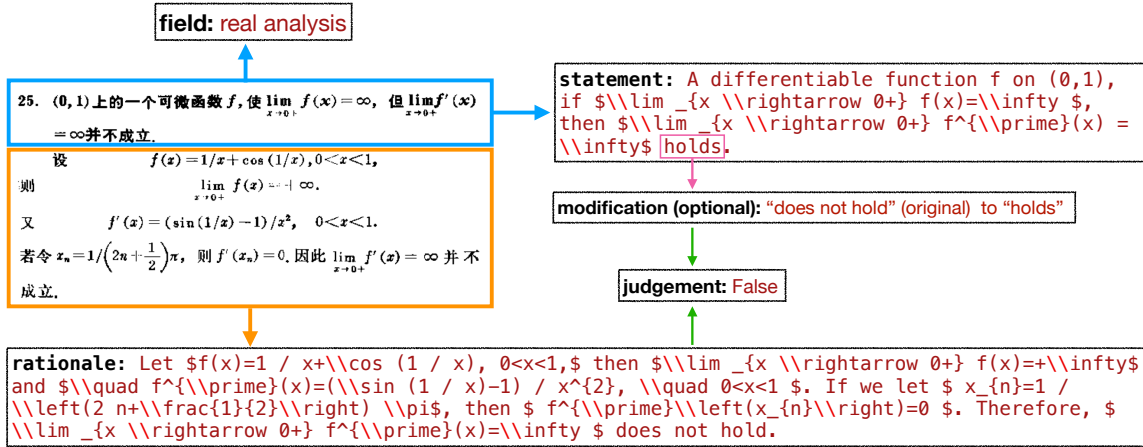


Figure 8. An annotation example from *Counterexamples in Real Analysis*. (Wang, 1989).

B. Details for Experimental Settings

Summary for Open-weight Baselines The summary of the open-weight baseline models is shown in the following Table 4. From Table 4, it is evident that current math-focused LLMs are built on a variety of base models, with **Mistral** being the most commonly used, followed by **Llama2**, **Qwen2-Math**, and **Deepseek-Math**. Notably, most models utilize data generation and augmentation strategies centered around **MATH** and **GSM8K**, which has accelerated the saturation of these benchmarks and highlighted the limitations of current “mathematical reasoning” capabilities in LLMs.

Furthermore, most academia-developed models are often constrained to supervised fine-tuning (SFT) only due to limited computation resources. Recent efforts, such as **Eurus-2-PRIME** and **Rho-Math**, have begun to explore advanced pre-training and post-training techniques, which have been (implicitly) validated by companies as effective in enhancing mathematical reasoning. However, the opacity surrounding engineering details in technical reports (OpenAI, 2023; Shao et al., 2024; Yang et al., 2024) hinders progress toward genuine mathematical reasoning in LLMs, which is critical for helping math researchers with true mathematical research.

Consequently, our proposed conceptual mathematical benchmark, COUNTERMATH, is a timely and significant

contribution to fostering advancements in genuine mathematical reasoning of LLMs.

Details of Prompts The used prompts are summarized as follows. We follow the corresponding tokenizers for *chat-template-based prompts* with the instruction as system prompt and statement as user input.

Completion Prompt

```
{statement}
Please reason step by step about whether the above statement is True or False, and put your final answer within \\boxed{}
```

Alpaca Prompt

Below is an instruction that describes a task. Write a response that appropriately completes the request.

Instruction:

```
{statement}
Please reason step by step about whether the above statement is True or False, and put your final answer within \\boxed{}
```

Response:

Chat-Template-Based Prompt

```
<BOS >system
Please reason step by step about whether the statement is True or False, and put your final answer within \\boxed{}.<EOS>
<BOS >user
{statement}<EOS>
```

Prompt with Hint

```
{statement}
Please reason by giving examples about whether the above statement is True or False, and put your final answer within \\boxed{}
```

Details of Evaluation Prompt The evaluation prompts used are as follows:

Evaluation Prompt

Role

You are a professional evaluator responsible for assessing the correctness of students' use of proof by contradiction to solve mathematical proof problems.

Objective

- Extract all examples in the reference solution.
- Extract all examples presented in the student's proof.
- Compare and determine whether the examples provided by the student's proof are semantically equivalent to those in the reference solution.

Workflow

1. Read and understand the given mathematical statement and its truth value.
2. Extract all examples from the reference solution.
3. Extract all examples from the student's proof. If the student does not provide any examples during the proof process, please output "None" in the ### <Examples in Student's Proof>.
4. Compare each example provided by the student's proof with the examples in the reference solution, determining whether they convey the same meaning. If they are semantically equivalent, output CONSISTENCY; otherwise, output UNCONSISTENCY.

Constraints

- Focus only on counterexamples used within the proof by contradiction, ignoring other proof methods.
- Ensure the accuracy of the comparison to avoid misinterpreting the student's intended meaning.

Output Format

When a student provides examples in a proof, for instance by stating "Let's consider an example to illustrate that", what follows may lead to the presentation of an example:

<Examples in Reference>

Example 1:

Example 2:

Example 3:

<Examples in Student's Proof>

Example 1:

Example 2:

<Each Result in Examples in Student's Proof>

Example 1: CONSISTENCY

Example 2: UNCONSISTENCY

When the student does not provide any examples:

<Examples in Reference>

Example 1:

Example 2:

Example 3:

<Examples in Student's Proof>

None

<Each Result in Examples in Student's Proof>

None

Input

Question:

Please judge whether the following statement is True or False:

{statement}

Reference Answer:

{answer}

Reference Proof:

{rationale}

Student Proof:

{model output}

Models	Scale	Base Models	Training Data	Training Paradigms
Deepseek-Math-RL	7B	Deepseek-Math-Base	Math corpus from Common Crawl, CoT/PoT/TIR mathematical instruction data, CoT data related to MATH and GSM8K	<i>CP, SFT, GRPO</i>
Qwen2.5-Math-Instruct	7B/72B	Qwen2.5-Math-Base	Math corpus including Common Crawl and synthetic data, synthetic CoT/TIR math data	<i>CP, SFT, GRPO</i>
InternLM2-Math-Plus	7B/20B/8x22B	InterLM2-Base/Mixtral	Math corpus from Common Crawl, in-house data and synthetic data, Open math-related instruction data	<i>CP, SFT</i>
WizardMath	7B/70B	Mistral/Llama2	Data augmentation based on MATH and GSM8K	<i>SFT, PPO</i>
Abel	7B/13B/70B	Llama-2	Data augmentation by Parental Oversight	<i>SFT</i>
Xwin-Math	7B/13B/70B	Llama-2	Data augmentation based on MATH and GSM8K	<i>SFT</i>
MAmmoTH2-Plus	7B/8x7B	Mistral/Mixtral	WebInstruct + additional instruction tuning datasets, including Math-Plus, Code-Feedback, etc.	<i>SFT</i>
Eurus-2-PRIME	7B	Qwen2.5-Math-Base	Reasoning-related instruction data from open datasets, Curated RL data from NuminaMath, Codeforces, etc.	<i>SFT, PRIME</i>
Numina-Math-TIR	7B	Deepseek-Math-Base	NuminaMath	<i>SFT</i>
Mathstral	7B	Mistral	NuminaMath	<i>SFT</i>
MetaMath-Mistral	7B	Mistral	Data augmentation based on MATH and GSM8K	<i>SFT</i>
Rho-math-interpreter	7B	Mistral	OpenWebMath& general corpora, ToRA(TIR data)	<i>SLM, SFT</i>
QwQ-Preview	32B	Qwen2.5*	?	?

Table 4. Summary of open-weight baseline models. *CP* stands for Continue Pretrain. *SFT* stands for Supervised Fine-Tuning. *GRPO* refers to a variant of PPO, which replaces the value network with the group average (Shao et al., 2024). PoT (Chen et al., 2023) and TIR (Gou et al., 2024) stand for Program-of-Thought and Tool-Integrated Reasoning, respectively. *PRIME* stands for using ORM as PRM by DPO-like rewards (Cui et al., 2025). *SLM* stands for Selective Language Modeling (Lin et al., 2024b). * only stands for the same model architecture.

C. Details for Training Data Engineering Framework

In constructing our training dataset, we utilized GPT to filter and refine the data. Below, we outline the specific prompt designs used in this process:

Data Filtering We developed multiple criteria for the LLM to evaluate whether the original data qualified as the counterexample data required for our task. These criteria ensured that the selected data met the conceptual and structural requirements of counterexample reasoning. The specific criteria are as follows:

Data Filtering Prompt

```
# Instruction
Please assess the following mathematical proof based on these criteria to determine if it employs counterexamples.

# Criterion

## Criterion 1:
Does the proof involve assuming the negation of the conclusion and deriving a contradiction from it? This indicates the use of proof by contradiction.

## Criterion 2:
Is there a specific instance provided that directly refutes a general assertion? This suggests the use of a counterexample.

## Criterion 3:
Does the explanation include specially chosen cases whose characteristics are essential for illustrating concepts or validating arguments? This points towards the use of special cases.

If at least one criterion is satisfied, consider it as meeting the condition and mark it as True. Remember, even if only one criterion applies, please retain this data.

# Input
The rationale you need to judge:

{statement}

{rationale}
```

Data Refinement We design prompts for the LLM to rewrite the data, making the training dataset more closely aligned with the benchmark distribution. This refinement process also highlights examples more explicitly, enabling the model to better learn counterexample-based conceptual reasoning. The details of our prompt design are provided below:

Data Refinement Prompt

Task:

You are tasked with modifying the proof processes in given rationales.

Instructions:

Annotate Examples:

Identify and annotate any specific examples used within the proof process.

Follow Reference Examples:

Use the reference examples and modified before-and-after examples provided below as a guideline for how to modify the text.

Preserve Proof Content:

Focus on reducing the length of the proof without altering its core content or logical flow.

Example References

- Field algebra: [Example A]

- Field real analysis: [Example B]

- Field functional analysis: [Example C]

- Field topology: [Example D]

Modified Before-and-After Examples:

Example 1:

Before:

After:

Example 2: *Before:*

After:

Example 3:

Before:

After:

Input:

{rationale}