Linguistic Generalizability of Test-Time Scaling in Mathematical Reasoning

Guijin Son[1,2,4] **Jiwoo Hong**[3] **Hyunwoo Ko**[2,4] **James Thorne**[3]

Yonsei University[1] OneLineAI[2] KAIST AI[3] MODULABS[4]
spthsrbwls123@yonsei.ac.kr

Abstract

Scaling pre-training compute has proven effective for achieving multilinguality, but does the same hold for test-time scaling? In this work, we introduce **MCLM**, a multilingual math benchmark featuring competition-level problems in 55 languages. We test three test-time scaling methods—Outcome Reward Modeling (ORM), Process Reward Modeling (PRM), and Budget Forcing (BF)—on both Qwen2.5-1.5B Math and **MR1-1.5B**, a multilingual LLM we trained for extended reasoning. Our experiments show that using Qwen2.5-1.5B Math with ORM achieves a score of 35.8 on MCLM, while BF on MR1-1.5B attains 35.2. Although "thinking LLMs" have recently garnered significant attention, we find that their performance is comparable to traditional scaling methods like best-of-N once constrained to similar levels of inference FLOPs. Moreover, while BF yields a 20-point improvement on English AIME, it provides only a 1.94-point average gain across other languages—a pattern consistent across the other test-time scaling methods we studied—highlighting that test-time scaling may not generalize as effectively to multilingual tasks. To foster further research, we release MCLM, MR1-1.5B, and evaluation results. [1]

1 Introduction

Large Language Models (LLMs) have achieved impressive gains across a wide range of tasks by scaling compute during pre-training (Thoppilan et al., 2022; Smith et al., 2022). Contrary to early concerns about a so-called "curse of multilinguality," (Conneau et al., 2020; Pfeiffer et al., 2022) which suggested that training in diverse languages would degrade overall performance, sufficiently large decoder-only architectures have demonstrated strong multilingual capabilities (Dubey et al., 2024; Aryabumi et al., 2024b).

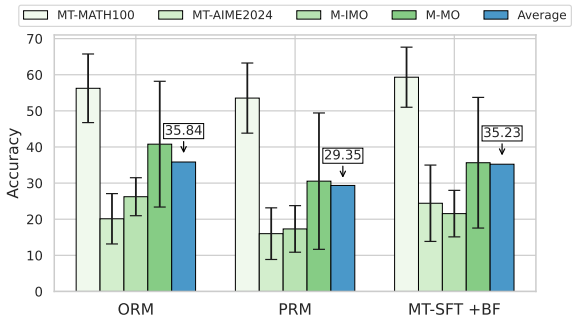

Figure 1: **Performance of Qwen2.5-1.5B-Math with different test-time scaling strategies.**——Once configured to use comparable inference FLOPs, all three methods (Outcome Reward Modeling, Process Reward Modeling, and Budget Forcing) achieve similar performance.

Yet as further scaling becomes increasingly difficult—due to data scarcity (Longpre et al., 2024), diminishing returns, or prohibitive costs (Achiam et al., 2023)—researchers have begun exploring test-time scaling methods that expand a model's reasoning or generation capacity at test time. An intriguing question arises: Does test-time scaling confer the same cross-lingual benefits we see at train-time scaling during pre-training?

Early studies demonstrated that chain-of-thought prompting (Wei et al., 2022) and scratchpads (Nye et al., 2021) can significantly boost model performance—particularly in mathematics (Lewkowycz et al., 2022; Azerbayev et al., 2024) and code (Le et al., 2022; Chae et al., 2024). Building on this, recent work proposes "test-time scaling," which further lengthens the chain-of-thought (Snell et al., 2025; Muennighoff et al., 2025). While such methods have proven effective for puzzles like Sudoku (Jellyfish042, 2024) and Hex (Jones, 2021), where action spaces are limited, mathematical reasoning remains relatively unexplored, largely due to its exponentially larger search space. To address this challenge, researchers have investigated external verifiers—such as best-of-N selection (Wang et al., 2022), Monte Carlo Tree Search (Guan et al., 2025; Tian et al., 2024; Wan et al., 2024), and

[1] https://github.com/gauss5930/MCLM

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 14333–14368
July 27 - August 1, 2025

process/outcome reward modeling (Zhang et al., 2025; Liu et al., 2025). Meanwhile, state-of-the-art LLMs (OpenAI, 2024, 2025) are capable of self-correction—often referred to as "system 2" reasoning (Xiang et al., 2025)—without explicit external verification. While longer chains of reasoning provide more room for in-depth thinking, they may also amplify the risk of error propagation (Bengio et al., 2015; Arora et al., 2022; Holtzman et al., 2020), making them more susceptible to out-of-domain disturbances such as language variation (Zhao et al., 2023; Chen et al., 2024b). From this vein, it remains unclear whether these strategies robustly generalize to new questions (SRI_Lab, 2025), languages, or domains.

In this work, we investigate the linguistic generalizability of test-time scaling methods by proposing a fine-grained multilingual complex reasoning benchmark, showing that test-time scaling alone does not yield robust multilingual performance. We build MCLM (**M**ultilingual **C**ompetition **L**evel **M**ath), a math reasoning dataset composed of four subsets varying source covering 55 languages.

We analyze three test-time scaling methods, outcome reward modeling (Wang et al., 2022, ORM), process reward modeling (Zhang et al., 2025, PRM), and budget forcing (Muennighoff et al., 2025, BF). We examine (1) *accuracy* to determine whether models retain overall performance across languages and (2) *consistency* to observe whether models can solve the same questions in different languages. While **ORM** and **PRM** provide clear gains on relatively easy datasets, the improvements are marginal for challenging tasks and inconsistent across the languages. Meantime, **BF** delivers noticeable gains only in English for tougher questions, with minimal impact on other languages. These findings underscore that while test-time scaling can enhance accuracy under certain conditions, it does not guarantee robust or consistent performance across multiple languages.

Finally, we introduce **MR1-1.5B**, an open multilingual thinking LLM trained on Deepseek-R1-1.5B using 100k R1-distilled instances translated by GPT-4o. Despite having only 1.5B parameters, **MR1** achieves performance on par with GPT-4o-Mini in multilingual mathematical reasoning.

2 Multilingual Competition Level Math

In this section, we introduce Multilingual Competition Level Math (**MCLM**), a multilingual math reasoning benchmark with challenging competition-level questions in 55 languages.

Models	MGSM
Gemma2-9B	78.37
Qwen2.5-14B-Instruct	82.27
Qwen2.5-72B-Instruct	88.16
Mistral-Large	89.01
GPT-4o-mini	87.36
o3-mini	**89.30**

Table 1: **MGSM performance of different models.** The 2025-01-31 version is used for o3-mini, remaining scores were sourced from the Yang et al. (2024b).

Going beyond math word problems A translated version of GSM8K (Cobbe et al., 2021), MGSM (Shi et al., 2023), has been widely used to assess the mathematical reasoning skills of multilingual LLMs (Anil et al., 2023; Shao et al., 2024; Aryabumi et al., 2024a). However, in Table 1, we observe that recent LLMs saturate MGSM. This implies the limitations of simple math word problems in accurately assessing the math reasoning capabilities of LLMs and necessitates a higher degree of complexity in reasoning benchmarks.

Assessing complex reasoning capabilities In this vein, recent studies have evaluated the reasoning capabilities of LLMs using competition-level math questions (MAA, 2024; Gao et al., 2025). While these benchmarks address limitations in simple math word problems, they are largely restricted to English and Chinese, limiting their ability to study multilingualism at scale.

2.1 Curating the MCLM benchmark

Machine-translated reasoning We select AIME and MATH-500 (Lightman et al., 2024), two widely used mathematical benchmarks, as the main source of complex math questions. For 100 questions randomly sampled from MATH-500 and full AIME datasets, we translate both benchmarks with GPT-4o (OpenAI et al., 2024), as shown to be proficient in translating mathematical contexts (Chen et al., 2024a; Lai et al., 2023). We then verified that the answers and equations remained unchanged after translation, removing one sample from MATH500 due to translation inconsistencies. Both subsets consist of questions with numerical answers only. For further details on the machine translation and sampling process, see Appendix A.2.

Subset	Source Benchmark	Languages	Sample Size per Language	Evaluation Method
MT-MATH100	Math-500	55	100	Rule-based verifier
MT-AIME2024	AIME 2024	55	30	Rule-based verifier
M-IMO	IMO (2006, 2024)	38	22–27	LLM-as-a-Judge
M-MO	Domestic/Regional Olympiads	11	28–31	LLM-as-a-Judge

Table 2: **Overview of benchmark subsets**: source benchmarks, language coverage (full lists in the appendix), sample sizes, and evaluation methods. Please see Appendix A.1 for the full list of languages.

Human-annotated reasoning To mitigate the potential biases in machine-translated data from translation artifacts (Plaza et al., 2024; Son et al., 2025), we also include human-translated or originally written questions.

First, we manually review 114 International Mathematical Olympiad (IMO, 2024, IMO) questions from 2006 to 2024 in English, excluding proof-based and image-heavy problems, resulting in a final set of 27. We then collect their official translations in 38 languages. Where official translations are unavailable, we do not substitute machine-generated versions, leaving those entries missing.

Second, we gather problems from various domestic and regional mathematical Olympiads worldwide. These contests originate in multiple languages, providing valuable data for multilingual mathematical reasoning. For English, Chinese, and Korean—where competition-level benchmarks already exist (He et al., 2024; Ko et al., 2025)—we incorporate existing datasets rather than recollecting data. While we exclude proof-based questions for simplicity, the final dataset still features a diverse range of answer formats (e.g., numerical, Boolean, descriptive) and spans 11 languages. We use GPT-4o-mini[1] for evaluation. An overview is provided in Table 2, additional details are in Appendix A.3.

3 Experimental Settings

In this section, we provide an overview of the test-time scaling methods evaluated (Section 3.1), compare the inference budgets in terms of FLOPs across different scaling techniques (Section 3.2), and describe the evaluation metrics used to assess their performance (Section 3.3).

3.1 Baselines: Test-Time Scaling Strategies

In this work, we evaluate three test-time scaling strategies using Qwen2.5-1.5B and 7B instruct models (Yang et al., 2024a) as baselines (Figure 2). We selected these model sizes because they offer a balanced trade-off between reasoning capacity and computational efficiency. Models smaller than 1.5B lack the capacity to solve complex problems, while larger models can be prohibitively expensive to scale (Biderman et al., 2023).

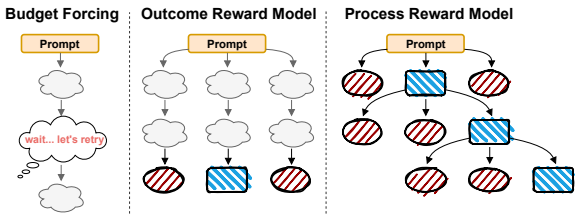

Figure 2: **Comparison of different inference-time scaling strategies.** Blue boxes represent selected outputs, while red boxes indicate rejected ones.

Outcome Reward Modeling We generate N responses per instance and use Qwen2.5-Math-72B-RM (Yang et al., 2024b) to evaluate them, selecting the highest-scoring answer as the final output.

Process Reward Modeling In contrast to outcome reward modeling, this strategy integrates the reward model during inference to guide the generation process. We employ Qwen2.5-Math-72B-PRM (Zhang et al., 2025); the model generates c candidate continuations at each step and selects the best one. For both ORM and PRM, the generator and reward model are served on separate servers, thereby avoiding the overhead of repeatedly on- and off-loading model weights.

Budget Forcing Recent LLMs, such as R1 (Guo et al., 2025) and O1 (OpenAI, 2024), are designed to generate longer chain-of-thoughts with in-context exploration and correction, allowing them to naturally scale during inference. However, this approach lacks controllability. To mitigate this, we adopt the budget-forcing method proposed by Muennighoff et al. (2025). In budget forcing, these thinking models are truncated and required to output an answer if they exceed a predefined budget. Conversely, if they fall short of the budget, they are prompted to generate additional reasoning steps,

[1] 2024-07-18 version (OpenAI et al., 2024)

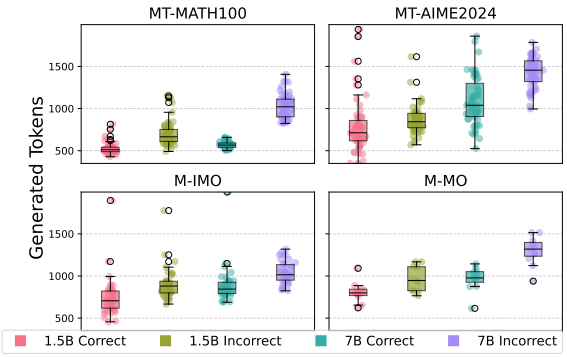

Figure 3: **# of generated tokens for 1.5B and 7B models in a greedy setting, divided by correctness.** Languages are represented as scatter plots, overlaid on box plots.

encouraging further exploration and correction.

3.2 Calculating Inference FLOPs

For our experiments, we first establish a unified inference budget based on two key estimates: the generator's cost is approximated as $2N_G D$ (Kaplan et al., 2020), where N_G is the number of parameters in the generation model and D is the total number of tokens generated per instance. The verifier's (or reward model's) cost is estimated as $4N_V$, where N_V is the number of parameters in the verifier. Here, the multiplier of 4 for the reward model's cost reflects a base cost of $2N_V$ that is doubled to account for the additional overhead incurred when invoking the reward model during inference (Snell et al., 2025). In the ORM, we generate k responses per instance, leading to a total inference cost of

$$k \times \Big(2N_G D + 4N_V\Big). \quad (1)$$

In Figure 3, under greedy generation, we observe that models tend to generate longer responses once they produce an error, particularly on harder benchmarks. Additionally, the 7B model generally produces longer outputs than the 1.5B model. However, no systematic trends emerge across the 55 languages—there is no clear pattern, such as longer outputs for low-resource languages. Although token counts vary with configuration, these differences are negligible compared to the dominant effect of model size on inference FLOPs. Consequently, we use an average of 921 tokens per question to estimate cost.

Outcome Reward Modeling In ORM with $k = 2$ responses, the inference cost is approximated as

$$\text{ORM FLOPs} \approx 2\Big(2N_G \times 921 + 4N_V\Big). \quad (2)$$

k	(S, c)	BF
2	(3, 3)	$\approx$ 2048 tokens
4	(4, 5)	$\approx$ 4096 tokens
8	(5, 8)	$\approx$ 8192 tokens

Table 3: **Selected configurations for PRM and BF**. Each S, c, and BF is set so that the inference FLOPs match ORM.

Assuming $N_G = 1.5 \times 10^9$ and $N_V = 72 \times 10^9$, this configuration results in an estimated cost of approximately 6.10×10^{12} FLOPs per instance.

Process Reward Modeling In PRM, at each generation step, the model produces c candidates, with each candidate generating x tokens (we fix $x = 128$ in our experiments). The total inference cost over S steps is given by

$$\text{PRM FLOPs} = S\,c\Big(x \cdot 2N_G + 4N_V\Big). \quad (3)$$

For PRM configurations, S and c, our preliminary experiments indicated that scaling one parameter in isolation produced suboptimal performance: a high S with a low c failed to explore sufficient alternatives, while an excessively low S prevented the generation process from completing. Therefore, we opted to proportionally scale both S and c to achieve a balanced search during generation.

Budget Forcing In contrast, BF relies solely on the generator, so its inference cost is given by

$$\text{SC FLOPs} = 2N_G \cdot BF, \quad (4)$$

where BF denotes the effective number of tokens that may be generated during the inference with budget forcing. In Table 3, we select the PRM parameters S and c and adjust the BF token budget BF so that the overall inference cost of each method matches that of ORM for $k = 2, 4$, and 8.

3.3 Evaluation Metrics

We evaluate our models using two primary metrics that capture performance at multiple levels: (1) *accuracy* and (2) *cross-lingual consistency*.

Accuracy We measure accuracy at a surface level to determine whether a single model achieves comparable performance across different languages.

Cross-Lingual Consistency To examine whether the model tends to solve (or fail) the same questions across languages, we compute **Fleiss' kappa** (Fleiss, 1971), which is originally designed to measure agreement among multiple annotators. In our

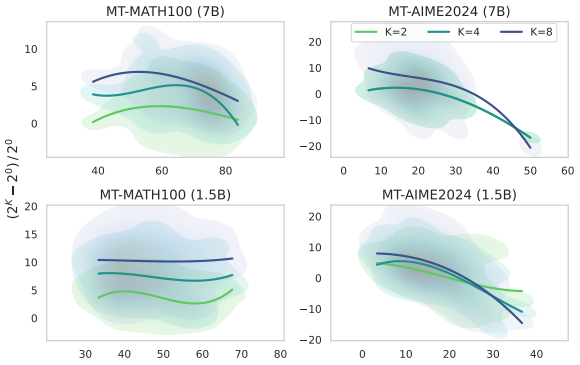

Figure 4: **Gains of ORM compared to a greedy-decoding baseline.** The semi-transparent "cloud" indicates the 2D data distribution via a KDE density plot, and the overlaid lines are third-order polynomial regressions modeling how each ORM setting scales with the baseline score.

setup, however, we treat each language as an "annotator'': for each problem, each "annotator'' (i.e., each language version of the model) provides either a correct or incorrect label. We then define consistency through Fleiss' kappa as:

$$\kappa = \frac{\bar{P} - \bar{P}_e}{1 - \bar{P}_e} \tag{5}$$

$$\bar{P} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{n(n-1)}\sum_{j=1}^{k} n_{ij}(n_{ij}-1) \tag{6}$$

$$\bar{P}_e = \sum_{j=1}^{k} p_j^2, \quad p_j = \frac{1}{Nn}\sum_{i=1}^{N} n_{ij}, \tag{7}$$

where N is the number of problems, n is the number of languages, and n_{ij} is the count of how many times language j gives a particular label (correct or incorrect) for problem i. In this formulation, $\bar{P}$ is the observed agreement (i.e., the proportion of problems for which all languages concur on correctness or incorrectness), and $\bar{P}_e$ is the expected agreement by chance. A high Fleiss' kappa indicates that the model responds consistently across languages (solving the same problems), not merely achieving similar overall accuracy by chance.

4 Result 1: ORM and PRM

In this section, we assess the multilingual robustness of both ORM and PRM. We find that, while each approach can boost performance, these gains do not consistently generalize across different languages and levels of difficulty.

4.1 Outcome Reward Modeling

For ORM, we use Qwen2.5-Math-1.5B and 7B-Instruct models to generate K samples per query and then apply Qwen2.5-Math-72B to score each sample, selecting the one with the highest score.

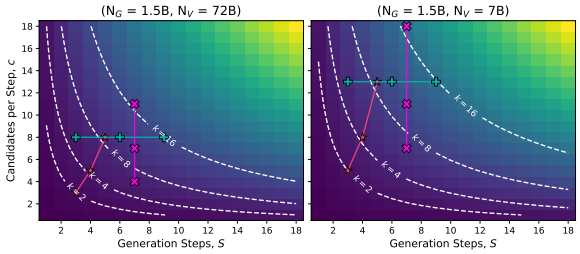

Figure 5: **PRM inference FLOPs as a function of generation steps S and candidates per step c.** The left panel uses a verifier size of 72B, while the right panel uses a 7B RM, displaying adjusted configurations to yield similar costs.

Limited gains at scale in non-English settings In Figure 4, we plot each model's baseline performance (averaged across 55 languages) on the x-axis versus the relative gain of each ORM setting (with $K \in \{2, 4, 8\}$) on the y-axis. On the MT-MATH100 dataset, both the 1.5B and 7B models show consistent improvement as K increases. However, on the more challenging MT-AIME2024 dataset, the gains for different K values are largely indistinguishable and, in some cases, even negative. This trend is comparable to English, which shows steady improvements also on MT-AIME2024—for instance, the 1.5B model rises from 16.67 to 26.67 to 36.67 as K increases, while the 7B model goes from 20.00 to 26.67 to 36.67.

Overall, while ORM is a viable scaling strategy in English, it yields limited returns in many other languages—possibly due to the models' difficulty in generating high-quality candidates. With few plausible options available, the reward model cannot effectively identify an improved solution.

4.2 Process Reward Modeling

Along with the configurations mentioned in Table 3, we experiment with additional setups to study how PRMs scale. Details are provided in Figure 5. In general, we evaluate two approaches: fixing S while increasing C (pink) and fixing C while scaling S (green). Additionally, we compare the efficacy of a 7B verifier against the original 72B. Due to cost constraints, these configurations are tested only on 14 languages of MT-MATH100 using Qwen2.5-Math-1.5B-Instruct.

No scalable gains for variance or consistency Figure 6 shows that, in PRM, the average performance of Qwen2.5-Math-1.5B-Instruct increases

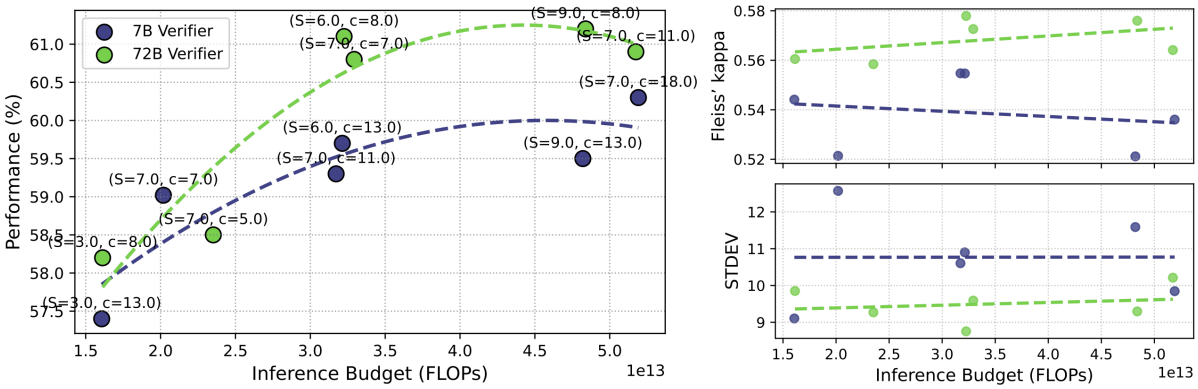

Figure 6: **Inference FLOPs versus PRM performance and consistency.** (Left) Second-degree polynomial regressions for average performance on 14 languages, comparing the 7B (blue) and 72B (green) reward models. (Right) Fleiss' kappa (top) and standard deviation (bottom) plotted against the same FLOPs budget; the fitted curves reveal no clear monotonic trend.

steadily with the inference budget. Even though the 7B reward model provides a larger search space, the 72B reward model achieves better outcomes under comparable compute. From a hardware standpoint, it can be more effective to run fewer steps and rely on a larger verifier. However, no clear pattern emerges for Fleiss' kappa or the standard deviation of individual scores, suggesting that adding more budget does not necessarily improve model consistency or reduce variance. In practical terms, while accuracy may scale with compute for PRM, cross-lingual consistency does not appear to follow, even for a relatively easier dataset like MT-MATH100.

4.3 ORM over PRM

As discussed earlier, both ORM and PRM exhibit unstable multilingual performance growth, with greater variance and lower Fleiss' Kappa scores at higher inference FLOPs. However, despite this instability, as shown in Figure 7, ORM consistently outperforms PRM in average accuracy, suggesting that, in general, it may be the more reliable choice. This is especially true given that, despite being assigned the same inference FLOPs, PRM invokes the reward model more frequently, requiring iterative back-and-forth interactions between the generator and evaluator, leading to higher latency.

5 Result 2: Budget Forcing

LLMs with "system 2" reasoning (Xiang et al., 2025)—such as OpenAI's newest O-series models (OpenAI, 2024), Google's Gemini Thinking[1], and Deepseek R1 (Guo et al., 2025)—are emerging as a test-time scaling approach. By generating and refining responses within a single inference (without external verifiers), these models seek to dramatically expand the inference budget for improved performance. In this section, we examine their effectiveness as a test-time scaling strategy. Because proprietary solutions remain largely opaque, we train our own LLMs with system 2 thinking. Here, we describe our training methods (Section 5.1), report performance on MCLM (Section 5.2) and scaling affects from budget forcing(Section 5.3).

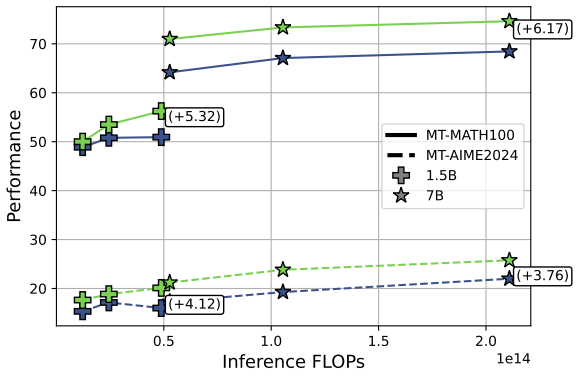

Figure 7: **Comparison of PRM vs. ORM performance on MATH (solid lines) and AIME (dashed lines).** 1.5B models are shown with plus markers, 7B models with stars. Blue lines represent PRM, green lines represent ORM. White box annotations indicate the performance difference (ORM PRM) at the highest compute setting for each line.

5.1 Inducing System 2 Thinking

A number of concurrent works propose diverse strategies for developing LLMs with long thinking. Broadly, these approaches fall into two main categories: (1) online reinforcement learning with verifiable outputs (Luo et al., 2025; Ye et al., 2025a), and (2) supervised fine-tuning on the "thinking trajectories" of proprietary LLMs (Muennighoff et al.,

[1] https://deepmind.google/technologies/gemini/flash-thinking/

Models	MT-MATH100	MT-AIME2024	M-IMO	M-MO	Average
Qwen2.5-Math-1.5B-Instruct	42.32 ± 8.61	16.36 ± 6.89	12.23 ± 6.02	25.00 ± 19.10	23.98
Deepseek-R1-1.5B	49.40 ± 8.84	17.21 ± 6.69	21.94 ± 6.75	26.77 ± 19.83	28.83
GPT-4o-Mini	70.30 ± 3.68	20.18 ± 6.83	13.33 ± 5.36	30.81 ± 15.80	33.66
o3-Mini	**84.89** ± 2.80	**45.33** ± 5.35	**29.75** ± 6.86	**51.42** ± 16.94	**52.85**
Qwen2.5-Math-1.5B + SFT	37.47 ± 7.56	14.85 ± 6.69	10.50 ± 5.16	18.40 ± 14.92	20.30
Qwen2.5-Math-1.5B + MT-SFT	42.02 ± 7.46	16.67 ± 7.31	10.52 ± 4.63	19.92 ± 12.68	22.28
Deepseek-R1-1.5B + MT-SFT	**55.61** ± 10.93	**19.94** ± 8.10	**19.20** ± 6.24	**28.97** ± 16.64	**30.93**

Table 4: **Model performance across MCLM.** Best model highlighted in **bold** for each panel. For results per language see Appendix C.

2025; Ye et al., 2025b). However, Luo et al. (2025) reports requiring 3,800 A100 GPU hours to induce such behavior, making it prohibitively expensive for our setting. Instead, we opt for supervised fine-tuning using thinking trajectories distilled from R1. Below are three training configurations.

Qwen2.5-Math-1.5B + SFT Following Muennighoff et al. (2025) and Huang et al. (2024), we randomly sample 50K thinking trajectories generated by R1 from the OpenR1-220K [1] dataset and fine-tune our model for three epochs.

Qwen2.5-Math-1.5B + SFT with Translated Data While training exclusively on English math problems has been shown to be effective and can generalize to new languages to some extent (Liu et al., 2024)—likely due to the universal nature of mathematical logic—we explore whether translating the data helps training. Following Ko et al. (2025); Zhang et al. (2023), we translate the problem and solution components into 14 languages[2] using GPT-4o, while keeping the reasoning process in English to leverage the model's strong proficiency in English-based logical reasoning. For further details on the training dataset, see Appendix B.

Deepseek-R1-1.5B + SFT with Translated Data Finally, we try initializing the translated SFT from Deepseek-R1-1.5B (hereafter MR1-1.5B). Since the model is already proficient in generating extended reasoning, we observe that longer training leads to performance degradation. To mitigate this, we terminate training at 0.5 epochs.

5.2 Performance of trained models

Translated data improves cross-lingual performance. Table 4 compares Qwen2.5-Math-1.5B and Deepseek-R1-1.5B under various fine-tuning regimes, revealing two key trends. First, incorporating translated data into Qwen2.5-Math-1.5B delivers a modest +1.98% improvement over an English-only setup, indicating that relying exclusively on English data is insufficient for robust cross-lingual performance (Liu et al., 2024). As shown in Figure 8, models trained on multilingual inputs begin with lower accuracy—likely due to increased entropy—but soon surpass their English-only counterparts and maintain lower variance across languages. Second, initiating fine-tuning from Deepseek-R1-1.5B, already adept at extended chain-of-thought reasoning, yields even greater gains on MT-AIME2024, M-IMO, and M-MO, performing on par with GPT-4o-Mini. Notably, MR1-1.5B reaches an average score of 30.93 (+2.1% over its original baseline) with just 0.5 epochs of training, underscoring how a self-correcting model more readily benefits from multilingual data. Collectively, these results suggest that while incorporating translated data benefits a monolingual base, leveraging a model with established self-correction capabilities can amplify these gains in multilingual math reasoning.

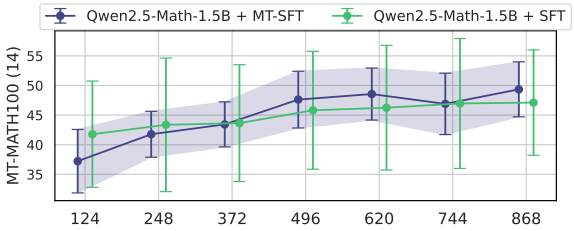

Figure 8: **Performance of Qwen2.5-Math-1.5B +SFT and + MT-SFT at each training checkpoint.** Average score and error bars for each checkpoint are displayed. The shaded region is the mean ± standard deviation for MT-SFT.

5.3 Budget-Constrained Scaling

To better compare self-correction with other scaling methods (e.g., ORM and PRM), we examine

[1] https://huggingface.co/datasets/open-r1

[2] See Appendix A.1 for the complete list of languages.

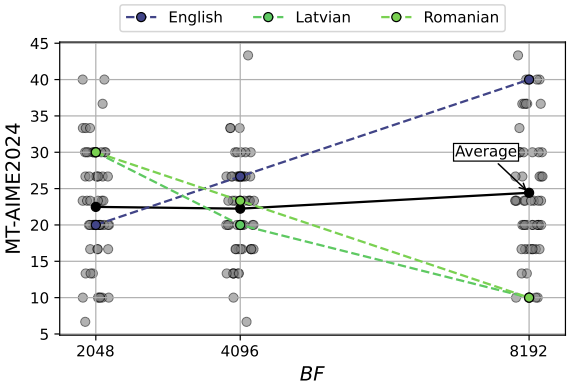

Figure 9: **Performance of MR1 on MT-AIME2024 at** $BF = \{2048, 4096, 8192\}$**.** Grey dots represent individual languages. Solid lines indicate average performance, while dashed lines highlight reference performances for selected languages.

its performance under fixed inference budgets. We apply the budget forcing approach introduced by Muennighoff et al. (2025) to constrain the generation budget of MR1-1.5B. Following the budget settings in Table 3, during inference, if the model reaches 90% of its allocated budget, we truncate the output and append "The final answer is" to prompt a concise answer. Conversely, if the model completes generation before reaching the limit, we truncate at the last line break, append "Wait...", and prompt the model to continue generating.

R1-like LLMs offer no clear edge over ORM Contrary to the recent surge of interest in "system 2" LLMs, that scale test-time compute by generating long reasoning traces, constraining these models to the same inference budgets reveals no clear advantage over test-time scaling methods such as ORM or PRM (see Figure 1). As shown in Figure 9, budget forcing yields nearly linear performance gains for English but provides limited benefits for most other languages. The distribution remains largely unchanged, achieving an overall average increase of only 1.9% as BF scales from 2048 to 8096. In some cases, such as Latvian and Romanian, performance even declines. Implying that there is scant evidence that the variance in performance diminishes or that coverage expands uniformly.

6 Related Works

Test-Time Scaling As concerns grow that the benefits of scaling pre-training compute may be saturating (Longpre et al., 2024), research has shifted toward test-time scaling, which expands the notion of chain-of-thought reasoning (Wei et al., 2022). Intuitively, the reasoning capacity of an LLM is limited by the number of tokens it can generate; hence, more challenging questions may require a longer chain of thought (Wu et al., 2025). An early example is self-consistency CoT (Wang et al., 2022), which generates multiple responses and selects the best via voting. This idea has since been developed into more cost-effective strategies for searching broader solution spaces (e.g., tree-of-thought methods (Yao et al., 2024), Monte Carlo Tree Search (Guan et al., 2025), and process supervision (Zhang et al., 2025; Luo et al., 2024)). Recently, models trained with online reinforcement learning (Shao et al., 2024) appear to exhibit an "aha moment," (Guo et al., 2025) wherein they dynamically decide to generate longer sequences to iteratively explore, solve, and self-correct.

Mathematical Reasoning in Non-English Early attempts at multilingual math reasoning involved supervised fine-tuning on translated datasets (Chen et al., 2024a; Lai and Nissim, 2024), but performance often deteriorated when models shifted away from their original language embeddings (Hong et al., 2025). To minimize such degradation, more recent work has increasingly relied on English as a pivot language. This approach can be implemented in various ways: either internally, by mapping multilingual inputs into an English-centric latent space (Yoon et al., 2024; Fan et al., 2025; Zhu et al., 2024; She et al., 2024), or externally, by translating non-English tasks into English and then back to the target language (Zhang et al., 2023; Ko et al., 2025). Although this strategy has reduced the performance gap between English and other languages, the stability of transfer under different training conditions remains underexplored. Moreover, many studies rely on the MGSM benchmark (Shi et al., 2023), which appears too easy for large-scale models or those enhanced by advanced reasoning techniques such as test-time scaling.

7 Conclusion

In this work, we examine the linguistic generalizability of three test-time scaling methodologies—Outcome Reward Modeling (ORM), Process Reward Modeling (PRM), and Budget Forcing (BF)—under budget-constrained settings. Using Qwen2.5-1.5B Math as a generator, ORM achieves a 35.84 score on our newly introduced multilingual math benchmark, **MCLM**, which spans 55

languages. With BF, **MR1-1.5B**—our multilingual LLM demonstrating extended reasoning—attains 35.23. Notably, once constrained to similar inference budgets, all three scaling methods exhibit comparable levels of improvement. Additionally, although these approaches appear promising in English (e.g., a 20-point improvement on AIME for the 1.5B model), we find that such gains do not consistently extend to other languages, where improvements average only 1.94 points—a pattern observed across all three methods. Moreover, increasing test-time compute often amplifies performance variance and reduces cross-linguistic consistency. To enable further study of these issues, we release both MCLM and MR1-1.5B.

Limitations

Although this work focuses solely on mathematical tasks, the lack of multilingual generalization we observe could be even more pronounced in areas requiring extensive cultural or domain-specific understanding; we leave this for future works. Additionally, due to budget constraints, this work primarily focuses on smaller-scale experiments (e.g., Qwen2.5-Math-1.5B-Instruct and occasionally Qwen2.5-Math-7B-Instruct). Although these parameter ranges are commonly used in both industry and academia, the observed lack of multilingual generalization for test-time scaling may not necessarily extend to significantly larger models (70B or more). Moreover, our test-time scaling experiments also remain on the smaller side; for instance, El-Kishky et al. (2025) scales to as many as 1162 candidates in their best-of-n setting. (It should still be noted that even experiments at this scale required over **2500 A100 GPU hours**) Given that the so-called "curse of multilinguality" (Conneau et al., 2020) naturally disappears as pre-training compute grows by several orders of magnitude (Aryabumi et al., 2024b), it is plausible that larger models may behave differently. Nevertheless, our findings at smaller scales—where test-time scaling shows no indication of fostering robust multilingualism—remain valuable, as they reveal potential boundaries for less resource-rich setups and highlight directions for future research.

Acknowledgments

This research was supported by Brian Impact Foundation, a non-profit organization dedicated to the advancement of science and technology for all, and Institute for Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (RS-2024-00398115, Technology research to ensure authenticity and consistency of results generated by AI) and (RS-2019-II190075, Artificial Intelligence Graduate School Program (KAIST)).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. arXiv preprint arXiv:2303.08774.

Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou, Denny Zhou, Slav Petrov, and Yonghui Wu. 2023. Palm 2 technical report. Preprint, arXiv:2305.10403.

Kushal Arora, Layla El Asri, Hareesh Bahuleyan, and Jackie Cheung. 2022. Why exposure bias matters: An imitation learning perspective of error accumulation in language generation. In Findings of the Association for Computational Linguistics: ACL

2022, pages 700–710, Dublin, Ireland. Association for Computational Linguistics.

Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, Madeline Smith, Jon Ander Campos, Yi Chern Tan, Kelly Marchisio, Max Bartolo, Sebastian Ruder, Acyr Locatelli, Julia Kreutzer, Nick Frosst, Aidan Gomez, Phil Blunsom, Marzieh Fadaee, Ahmet Üstün, and Sara Hooker. 2024a. Aya 23: Open weight releases to further multilingual progress. Preprint, arXiv:2405.15032.

Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, Madeline Smith, Jon Ander Campos, Yi Chern Tan, et al. 2024b. Aya 23: Open weight releases to further multilingual progress. arXiv preprint arXiv:2405.15032.

Axolotl AI. 2025. Axolotl: Scalable fine-tuning framework for llms. https://axolotl-ai-cloud.github.io/axolotl/. Github.

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen Marcus McAleer, Albert Q. Jiang, Jia Deng, Stella Biderman, and Sean Welleck. 2024. Llemma: An open language model for mathematics. In The Twelfth International Conference on Learning Representations.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. Advances in neural information processing systems, 28.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. 2023. Pythia: A suite for analyzing large language models across training and scaling. In International Conference on Machine Learning, pages 2397–2430. PMLR.

Hyungjoo Chae, Yeonghyeon Kim, Seungone Kim, Kai Tzu-iunn Ong, Beong-woo Kwak, Moohyeon Kim, Sunghwan Kim, Taeyoon Kwon, Jiwan Chung, Youngjae Yu, and Jinyoung Yeo. 2024. Language models as compilers: Simulating pseudocode execution improves algorithmic reasoning in language models. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 22471–22502, Miami, Florida, USA. Association for Computational Linguistics.

Nuo Chen, Zinan Zheng, Ning Wu, Ming Gong, Dongmei Zhang, and Jia Li. 2024a. Breaking language barriers in multilingual mathematical reasoning: Insights and observations. In Findings of the Association for Computational Linguistics: EMNLP 2024, pages 7001–7016, Miami, Florida, USA. Association for Computational Linguistics.

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

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

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 8440–8451, Online. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. arXiv preprint arXiv:2407.21783.

Ahmed El-Kishky, Alexander Wei, Andre Saraiva, Borys Minaev, Daniel Selsam, David Dohan, Francis Song, Hunter Lightman, Ignasi Clavera, Jakub Pachocki, et al. 2025. Competitive programming with large reasoning models. arXiv preprint arXiv:2502.06807.

Yuchun Fan, Yongyu Mu, YiLin Wang, Lei Huang, Junhao Ruan, Bei Li, Tong Xiao, Shujian Huang, Xiaocheng Feng, and Jingbo Zhu. 2025. Slam: Towards efficient multilingual reasoning via selective language alignment. In Proceedings of the 31st International Conference on Computational Linguistics, pages 9499–9515. Association for Computational Linguistics.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. Psychological bulletin, 76(5):378.

Bofei Gao, Feifan Song, Zhe Yang, Zefan Cai, Yibo Miao, Qingxiu Dong, Lei Li, Chenghao Ma, Liang Chen, Runxin Xu, Zhengyang Tang, Benyou Wang, Daoguang Zan, Shanghaoran Quan, Ge Zhang, Lei Sha, Yichang Zhang, Xuancheng Ren, Tianyu Liu, and Baobao Chang. 2025. Omni-MATH: A universal olympiad level mathematic benchmark for large language models. In The Thirteenth International Conference on Learning Representations.

Xinyu Guan, Li Lyna Zhang, Yifei Liu, Ning Shang, Youran Sun, Yi Zhu, Fan Yang, and Mao Yang. 2025. rstar-math: Small llms can master math reasoning with self-evolved deep thinking. arXiv preprint arXiv:2501.04519.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. arXiv preprint arXiv:2501.12948.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. 2024. OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 3828–3850, Bangkok, Thailand. Association for Computational Linguistics.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In International Conference on Learning Representations.

Jiwoo Hong, Noah Lee, Rodrigo Martínez-Castaño, César Rodríguez, and James Thorne. 2025. Cross-lingual transfer of reward models in multilingual alignment. In Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers), pages 82–94, Albuquerque, New Mexico. Association for Computational Linguistics.

Pin-Lun Hsu, Yun Dai, Vignesh Kothapalli, Qingquan Song, Shao Tang, Siyu Zhu, Steven Shimizu, Shivam Sahni, Haowen Ning, and Yanning Chen. 2024. Liger kernel: Efficient triton kernels for llm training. arXiv preprint arXiv:2410.10989.

Zhen Huang, Haoyang Zou, Xuefeng Li, Yixiu Liu, Yuxiang Zheng, Ethan Chern, Shijie Xia, Yiwei Qin, Weizhe Yuan, and Pengfei Liu. 2024. O1 replication journey–part 2: Surpassing o1-preview through simple distillation, big progress or bitter lesson? arXiv preprint arXiv:2411.16489.

IMO. 2024. International Mathematical Olympiad Website. Accessed: 2024-01-31.

Jellyfish042. 2024. Sudoku-rwkv: A specialized rwkv model for solving sudoku puzzles. https://github.com/Jellyfish042/Sudoku-RWKV. Accessed: 2025-02-11.

Andy L. Jones. 2021. Scaling scaling laws with board games. arXiv preprint arXiv:2104.03113.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. arXiv preprint arXiv:2001.08361.

Hyunwoo Ko, Guijin Son, and Dasol Choi. 2025. Understand, solve and translate: Bridging the multilingual mathematical reasoning gap. arXiv preprint arXiv:2501.02448.

Bespoke Labs. 2025. Bespoke-stratos-17k dataset. Accessed: February 1, 2025.

Huiyuan Lai and Malvina Nissim. 2024. mCoT: Multilingual instruction tuning for reasoning consistency in language models. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 12012–12026, Bangkok, Thailand. Association for Computational Linguistics.

Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 318–327, Singapore. Association for Computational Linguistics.

Hung Le, Yue Wang, Akhilesh Deepak Gotmare, Silvio Savarese, and Steven Chu Hong Hoi. 2022. Coderl: Mastering code generation through pretrained models and deep reinforcement learning. Advances in Neural Information Processing Systems, 35:21314–21328.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. 2022. Solving quantitative reasoning problems with language models. Advances in Neural Information Processing Systems, 35:3843–3857.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In The Twelfth International Conference on Learning Representations.

Pengfei Liu, Yiming Wang, Tianyu Gao, et al. 2025. Process reinforcement through implicit rewards. arXiv preprint arXiv:2502.01456.

Zihan Liu, Yang Chen, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2024. Acemath: Advancing frontier math reasoning with post-training and reward modeling. arXiv preprint arXiv:2412.15084.

Shayne Longpre, Robert Mahari, Ariel N. Lee, Campbell S. Lund, Hamidah Oderinwale, William Brannon, Nayan Saxena, Naana Obeng-Marnu, Tobin South, Cole J Hunter, Kevin Klyman, Christopher Klamm, Hailey Schoelkopf, Nikhil Singh, Manuel Cherep, Ahmad Mustafa Anis, An Dinh, Caroline Shamiso Chitongo, Da Yin, Damien Sileo, Deividas Mataciunas, Diganta Misra, Emad A. Alghamdi, Enrico Shippole, Jianguo Zhang, Joanna Materzynska, Kun Qian, Kushagra Tiwary, Lester James Validad Miranda, Manan Dey, Minnie Liang, Mohammed Hamdy, Niklas Muennighoff, Seonghyeon Ye, Seungone Kim, Shrestha Mohanty, Vipul Gupta, Vivek Sharma, Vu Minh Chien, Xuhui Zhou, Yizhi LI, Caiming Xiong, Luis Villa, Stella Biderman, Hanlin Li, Daphne Ippolito, Sara Hooker,

Jad Kabbara, and Alex Pentland. 2024. Consent in crisis: The rapid decline of the AI data commons. In The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track.

Liangchen Luo, Yinxiao Liu, Rosanne Liu, Samrat Phatale, Harsh Lara, Yunxuan Li, Lei Shu, Yun Zhu, Lei Meng, Jiao Sun, et al. 2024. Improve mathematical reasoning in language models by automated process supervision. arXiv preprint arXiv:2406.06592.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Erran Li, Raluca Ada Popa, and Ion Stoica. 2025. Deepscaler: Surpassing o1-preview with a 1.5b model by scaling rl. https://pretty-radio-b75.notion.site/DeepScaleR-Surpassing-O1-Preview-with-a-1-5B-Model-by-Scaling-RL-19681902c1468005bed8ca303013a4e2. Notion Blog.

MAA. 2024. American invitational mathematics examination - aime. In American Invitational Mathematics Examination - AIME 2024.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candes, and Tatsunori Hashimoto. 2025. s1: Simple test-time scaling. In Workshop on Reasoning and Planning for Large Language Models.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. 2021. Show your work: Scratchpads for intermediate computation with language models. arXiv preprint arXiv:2112.00114.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2024. 2 olmo 2 furious. arXiv preprint arXiv:2501.00656.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian O'Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang, Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro,

Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi, Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Cunninghman, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. 2024. Gpt-4o system card. Preprint, arXiv:2410.21276.

OpenAI. 2024. Openai o1 system card. arXiv preprint arXiv:2412.16720.

OpenAI. 2025. Openai o3-mini system card.

Jonas Pfeiffer, Naman Goyal, Xi Lin, Xian Li, James Cross, Sebastian Riedel, and Mikel Artetxe. 2022. Lifting the curse of multilinguality by pre-training modular transformers. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 3479–3495, Seattle, United States. Association for Computational Linguistics.

Irene Plaza, Nina Melero, Cristina del Pozo, Javier Conde, Pedro Reviriego, Marina Mayor-Rocher, and María Grandury. 2024. Spanish and llm benchmarks: is mmlu lost in translation? arXiv preprint arXiv:2406.17789.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory optimizations toward training trillion parameter models. In SC20: International Conference for High Performance Computing, Networking, Storage and Analysis, pages 1–16. IEEE.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. Preprint, arXiv:2402.03300.

Shuaijie She, Wei Zou, Shujian Huang, Wenhao Zhu, Xiang Liu, Xiang Geng, and Jiajun Chen. 2024. MAPO: Advancing multilingual reasoning through multilingual-alignment-as-preference optimization. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 10015–10027. Association for Computational Linguistics.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2023. Language models are multilingual chain-of-thought reasoners. In The Eleventh International Conference on Learning Representations.

Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, Elton Zhang, Rewon Child, Reza Yazdani Aminabadi, Julie Bernauer, Xia Song, Mohammad Shoeybi, Yuxiong He, Michael Houston, Saurabh Tiwary, and Bryan Catanzaro. 2022. Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model. arXiv preprint arXiv:2201.11990.

Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2025. Scaling test-time compute optimally can be more effective than scaling LLM parameters. In The Thirteenth International Conference on Learning Representations.

Guijin Son, Hanwool Lee, Sungdong Kim, Seungone Kim, Niklas Muennighoff, Taekyoon Choi, Cheonbok Park, Kang Min Yoo, and Stella Biderman. 2025. KMMLU: Measuring massive multitask language understanding in Korean. In Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 4076–4104, Albuquerque, New Mexico. Association for Computational Linguistics.

SRI_Lab. 2025. eth-sri/matharena. Original-date: 2025-02-12T19:06:14Z.

Qwen Team. 2024. Qwq: Reflect deeply on the boundaries of the unknown.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, YaGuang Li, Hongrae Lee, Huaixiu Steven Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, Dehao Chen, Yuanzhong Xu, Zhifeng Chen, Adam Roberts, Maarten Bosma, Vincent Zhao, Yanqi Zhou, Chung-Ching Chang, Igor Krivokon, Will Rusch, Marc Pickett, Pranesh Srinivasan, Laichee Man, Kathleen Meier-Hellstern, Meredith Ringel Morris, Tulsee Doshi, Renelito Delos Santos, Toju Duke, Johnny Soraker, Ben Zevenbergen, Vinodkumar Prabhakaran, Mark Diaz, Ben Hutchinson, Kristen Olson, Alejandra Molina, Erin Hoffman-John, Josh Lee, Lora Aroyo, Ravi Rajakumar, Alena Butryna, Matthew Lamm, Viktoriya Kuzmina, Joe Fenton, Aaron Cohen, Rachel Bernstein, Ray Kurzweil, Blaise Aguera-Arcas, Claire Cui, Marian Croak, Ed Chi, and Quoc Le. 2022. LaMDA: Language models for dialog applications. arXiv preprint arXiv:2201.08239.

Open Thoughts. 2025. Openthoughts-114k dataset. Accessed: February 1, 2025.

Ye Tian, Baolin Peng, Linfeng Song, Lifeng Jin, Dian Yu, Haitao Mi, and Dong Yu. 2024. Toward self-improvement of llms via imagination, searching, and criticizing. arXiv preprint arXiv:2404.12253.

Ziyu Wan, Xidong Feng, Muning Wen, Stephen Marcus McAleer, Ying Wen, Weinan Zhang, and Jun Wang. 2024. Alphazero-like tree-search can guide large language model decoding and training. In Proceedings of the 41st International Conference on Machine Learning, ICML'24. JMLR.org.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. arXiv preprint arXiv:2203.11171.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems, 35:24824–24837.

Yuyang Wu, Yifei Wang, Tianqi Du, Stefanie Jegelka, and Yisen Wang. 2025. When more is less: Understanding chain-of-thought length in LLMs. In Workshop on Reasoning and Planning for Large Language Models.

Violet Xiang, Charlie Snell, Kanishk Gandhi, Alon Albalak, Anikait Singh, Chase Blagden, Duy Phung, Rafael Rafailov, Nathan Lile, Dakota Mahan, Louis Castricato, Jan-Philipp Franken, Nick Haber, and Chelsea Finn. 2025. Towards system 2 reasoning in llms: Learning how to think with meta chain-of-thought. Preprint, arXiv:2501.04682.

Cheng Xu, Shuhao Guan, Derek Greene, M Kechadi, et al. 2024. Benchmark data contamination of large language models: A survey. arXiv preprint arXiv:2406.04244.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024a. Qwen2. 5 technical report. arXiv preprint arXiv:2412.15115.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. 2024b. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. arXiv preprint arXiv:2409.12122.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. Advances in Neural Information Processing Systems, 36.

Guanghao Ye, Khiem Duc Pham, Xinzhi Zhang, Sivakanth Gopi, Baolin Peng, Beibin Li, Janardhan Kulkarni, and Huseyin A Inan. 2025a. On the emergence of thinking in llms i: Searching for the right intuition. arXiv preprint arXiv:2502.06773.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. 2025b. Limo: Less is more for reasoning. arXiv preprint arXiv:2502.03387.

Dongkeun Yoon, Joel Jang, Sungdong Kim, Seungone Kim, Sheikh Shafayat, and Minjoon Seo. 2024. Langbridge: Multilingual reasoning without multilingual supervision. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 7502–7522. Association for Computational Linguistics.

Qian Zhang, Wei Li, Hao Chen, Jun Wang, and Yang Liu. 2025. The lessons of developing process reward models in mathematical reasoning. arXiv preprint arXiv:2501.07301.

Zhihan Zhang, Dong-Ho Lee, Yuwei Fang, Wenhao Yu, Mengzhao Jia, Meng Jiang, and Francesco Barbieri. 2023. Plug: Leveraging pivot language in cross-lingual instruction tuning. arXiv preprint arXiv:2311.08711.

Zirui Zhao, Wee Sun Lee, and David Hsu. 2023. Large language models as commonsense knowledge for large-scale task planning. In NeurIPS 2023 Foundation Models for Decision Making Workshop.

Wenhao Zhu, Shujian Huang, Fei Yuan, Shuaijie She, Jiajun Chen, and Alexandra Birch. 2024. Question translation training for better multilingual reasoning. CoRR, abs/2401.07817.

A Additional details on MCLM

In this section, we provide additional details on the MCLM benchmark, including the languages covered by each subset (Section A.1), the sampling process for MT-MATH100 (Section A.2), the sources for the M-IMO, and M-MO subset (Section A.3), the prompts used for GPT-4o and -mini (Section A.4), and contamination considerations (Section A.5).

A.1 Details in Language Coverage

We examine four groups of languages in this paper: (A) the 55 languages into which MATH500 and AIME2024 have been translated, (B) the 14 languages frequently sampled to reduce evaluation costs, (C) the languages covered in M-IMO, and (D) those in M-MO. The complete list for each group is provided in Table 5.

Lang. Group	Languages (ISO Codes, Sorted Alphabetically)	# Lang.
(A)	af, ar, bg, bn, ca, cs, cy, da, de, el, en, es, et, fa, fi, fr, gu, he, hi, hr, hu, id, it, ja, kn, ko, lt, lv, mk, ml, mr, ne, nl, no, pa, pl, pt, ro, ru, sk, sl, so, sq, sv, sw, ta, te, th, tl, tr, uk, ur, vi, zh-cn, zh-tw	55
(B)	af, ar, de, en, es, fr, he, id, it, ja, ko, tr, vi, zh-cn	14
(C)	af, ar, bg, cs, da, de, el, en, et, es, fi, fr, he, hr, hu, id, it, ja, ko, lt, lv, mk, nl, no, pl, pt, ro, ru, sk, sl, sq, sv, th, tr, uk, vi, zh-cn, zh-tw	38
(D)	cs, de, en, fr, ja, ko, nl, pl, ru, sk, zh-cn	11

Table 5: Full language lists for each dataset subset. MT-MATH100, MT-AIME2024, M-IMO, and M-MO cover 55, 38, and 11 ISO codes respectively.

Running evaluations on all 55 languages can be computationally intensive, particularly when testing for test-time scaling. To address this, we have created a downsampled group (B) consisting of 14 languages for our experiments. These languages were chosen to represent a broad spectrum of linguistic families and writing systems. In terms of language families, the selection includes representatives from Afro-Asiatic (Arabic, Hebrew), Austronesian (Indonesian), Japonic (Japanese), Koreanic (Korean), Turkic (Turkish), Austroasiatic (Vietnamese), and Sino-Tibetan (Chinese). Additionally, the chosen languages encompass diverse scripts: several, including Afrikaans, German, English, Spanish, French, and Italian, use the Latin alphabet; others use distinct writing systems—Arabic and Hebrew employ abjads (consonant-based scripts); Japanese combines logographic characters (Kanji) with syllabic scripts (Hiragana and Katakana); Korean is written in Hangul; Turkish uses a modified Latin alphabet; Vietnamese utilizes a Latin-based alphabet with diacritics; and Chinese is written using Chinese characters.

A.2 Sampling MATH100

Once creating MGSM, Shi et al. (2023) opted to randomly sample 250 questions from the GSM8K dataset for computational efficiency. Similarly, we sample 100 questions from MATH500 to keep evaluation costs manageable across multiple languages. Initially, we conduct random sampling. Before extending this approach to all 55 languages, we first apply it to the language group (B). For language group (B), we create both the MT-MATH100 and MT-MATH500 versions, where entire subsets are translated for the later. We then evaluate 10 models—each trained using different methods to enhance mathematical reasoning—to determine whether the sampled MATH100 subset reliably represents the full dataset.

In Table 6, we report the performance of the evaluated models. The score differences are relatively small, and even after accounting for minor variations, the ranking of the 10 models remains largely consistent—with only a few instances of rank switching. We conclude that the sampled version serves as an acceptable proxy for the full dataset and proceed accordingly.

Rank	Model	MATH-500	MATH-100	Score Diff.	Rank Diff.
1	o3-mini	85.00	85.93	0.93	-
2	Eurus-2-7B-PRIME	73.76	76.63	2.86	-
3	Qwen2.5-Math-7B-Instruct	73.70	75.98	2.27	-
4	DeepSeek-R1-Distill-Qwen-32B	72.73	75.98	3.24	-
5	DeepSeek-R1-Distill-Qwen-7B	67.25	68.69	1.44	1 ▲
6	AceMath-7B-Instruct	65.90	70.06	4.16	1 ▼
7	AceMath-1.5B-Instruct	65.60	68.19	2.58	-
8	DeepSeek-R1-Distill-Qwen-1.5B	53.74	56.78	3.05	-
9	Qwen2.5-Math-1.5B-Instruct	51.80	51.30	0.51	-
10	Qwen2.5-Math-1.5B-OREO	39.92	38.45	1.47	-

Table 6: Model rankings and score comparison between MATH-500 and MATH-100. The score difference was computed as the absolute difference between the MATH-500 and MATH-100 scores. The rank difference indicates the change in ranking on MATH-100 relative to the performance on MATH-500.

A.3 Sourcing M-IMO and M-MO

Relying solely on machine-translated benchmarks (MT-MATH100 and MT-AIME2024) carries inherent risks. To mitigate this, we supplement our dataset with questions from the International Mathematical Olympiad (IMO) and various regional math olympiads. Figure 10 provides an overview of the IMO questions included from 2004 to 2024, while Table 7 lists the sources for regional olympiads. For English, Chinese, and Korean, we utilize existing datasets rather than recollecting questions (He et al., 2024; Ko et al., 2025).

Language	Competition Links
French	https://euler.ac-versailles.fr/spip.php?rubrique207
German	DeMO
Japanese	https://www.imojp.org/domestic/jmo_overview.html#Problems
Dutch	https://prime.ugent.be/activiteiten/puma/
	https://wiskundeolympiade.nl/wedstrijdarchief/1e-ronde
Czech	https://www.matematickaolympiada.cz/mo-pro-ss/rocnik
	https://iksko.org/problems.php
Polish	https://om.sem.edu.pl/problems/
Slovakian	https://skmo.sk/dokumenty.php?rocnik=74
	https://riesky.sk/archiv/
Russian	https://mmo.mccme.ru//

Table 7: Link to mathematical competition links that has been included in M-MO subset.

A.4 Prompts

The prompts used for question translation (Figure 13), solution translation (Figure 14), and model judgment (Figure 15) are detailed accordingly.

A.5 Benchmark Contamination

As LLMs are trained on ever-growing datasets, concerns about benchmark contamination have emerged (Xu et al., 2024). Because most training data and procedures remain proprietary, it is virtually impossible to guarantee that a benchmark is entirely free of contamination. Existing detection methods remain unstable, and our MCLM dataset, particularly its M-IMO and M-MO subsets collected from the Internet, may be prone to exposure in various model-training corpora. For instance, we observe that Llama-3.1-3B-Instruct can produce correct answers without any visible reasoning, suggesting prior

familiarity with certain questions. However, lacking a robust decontamination method, we have chosen not to remove potentially contaminated samples. Despite this possibility, the overall low performance of most models on these subsets suggests that MCLM remains a challenging and valuable resource for evaluating multilingual, competition-level math. Furthermore, our primary focus is on assessing multilingual consistency. If a model correctly "guesses" a single language version of a question (possibly due to prior exposure), it does not necessarily indicate strong multilingual reasoning. Conversely, if the model can solve all language variations after seeing only a few, it may demonstrate a degree of cross-lingual robustness.

A.6 License

The machine-translated subsets of MCLM are released under the MIT License. The remaining subsets are provided under a CC BY-NC-ND license, although we may transition to a more permissive license pending further review of the original competition data policies.

B Additional details in training LLMs with system 2 thinking

In this section, we provide additional details on the dataset used to train self-correcting LLMs (Section B.1), ablation studies on MR1 training (Section B.2), and the training configurations.

B.1 Additional details on the training dataset

In training MR1 (i.e., Deepseek-R1-1.5B + SFT with translated data), we first collect thinking trajectories generated by R1 (Guo et al., 2025) from the Numina Math dataset in Be-Spoke Stratos (Labs, 2025) and OpenThoughts (Thoughts, 2025). To ensure high-quality reasoning supervision, we exclude any data distilled from smaller thinking-model variants (e.g., QwQ-32B-Preview (Team, 2024) or the R1-Distil series). We then employ GPT-4o as an LLM-based judge to filter out instances with incorrect answers.

Next, the question and solution for each remaining instance are translated into one of the 14 languages in Group (B). We opt for only 14 languages—as opposed to all 55—because our ablation studies indicate that oversampling languages can negatively impact overall performance. A rule-based parser validates that the question and answer content remains unchanged post-translation. Beyond this verification, no additional quality checks are performed. However, when we encounter elevated loss spikes during training, we backtrack through the dataset to identify and remove problematic instances. Following this process, we retain approximately 120K instances.

Proxying problem difficulty Table 8 compares EULER-INSTRUCT with existing multilingual math datasets: MGSM8KInstruct (Chen et al., 2024a) and mCoT-MATH (Lai and Nissim, 2024). MGSM8KInstruct extends GSM8K (Lightman et al., 2024) by translating it into 10 languages, yielding a parallel dataset of approximately 8,000 questions per language. In contrast, mCoT-MATH sources 560,000 seed questions from MetaMathQA and MathInstruct and translates them into 10 languages.

Dataset	# Lang.	# Inst.	Diff.
MGSM8KInstruct	10	73.6k	G.S
mCoT-MATH	10	6.3M	G.S
Euler-Instruct (Ours)	55	250K	C.L

Table 8: **Comparison of Multilingual Mathematical Reasoning Datasets.** The *Diff.* column indicates difficulty level, where *G.S* represents grade school level and *C.L* represents competition level.

To estimate the difficulty level of each dataset, we randomly sample 1,000 questions and measure the solve rate of LLMs. Since MGSM8KInstruct and mCoT-MATH were published before EULER-INSTRUCT, they may have been included in the training of existing LLMs, potentially making a direct comparison unfair. To address this, we first evaluate using OLMo2-{7, 13}B (OLMo et al., 2024); we use the base model instead of the instruct model since GSM8K is used during the instruction tuning phase. Since OLMo2-7B-base is not instruction-tuned, we provide three-shot examples and prompt it to solve the

questions in a structured format: "The answer is X." Additionally, we evaluate with Qwen2.5-{7, 32, 72}B-Instruct (Yang et al., 2024b) in a zero-shot setting, using a parser to extract answers.

Figure 11 illustrates that, for both mCoT-MATH and MGSM8KInstruct, over half of the problems are solved by Qwen2.5-7B-Instruct, and Qwen2.5-72B-Instruct solves more than 70%. In contrast, the solve rate for EULER-INSTRUCT remains low across all models, with the 7B variants solving less than 20% of the problems and even the 72B variants achieving below 40%. Notably, the Verify subset exhibits a lower solve rate than previous datasets, indicating that its difficulty remains high even when restricted to numerical answers.

B.2 Training Ablations

Before training MR1 we meet the question: How little additional data is required to incorporate a new language into a self-correcting model? To address this, we train four models under a fixed total budget of 24,000 training instances. Table 9 details the language composition and the per-language instance allocation for each model.

Languages	# Lang.	# Instances
ko	1	24k
af, fr, ko	3	8k
af, ar, fr, he, id, ko, tr	7	≈3.5k
all 14 in EULER-INSTRUCT	14	≈1.7k

Table 9: **Details on trained models.** All models are trained with a total of 24,000 instances. # Instances denote the number of instances used per language.

Figure 12 presents the performance gains over the base model (DeepSeek-R1-1.5B) as a function of the number of training instances. On MT-MATH100, performance rises sharply once the per-language budget reaches approximately 3.5k instances. By contrast, MT-AIME2024 shows more gradual improvements, suggesting that transferring self-correction capabilities to a challenging set of new-language questions requires a larger data allocation. Based on these findings, we use 14 languages within a total budget of 120K, ensuring that each language includes at least 8k instances.

B.3 Training Configurations and Logs

Axolotl (Axolotl AI, 2025) is used for the SFT training in Section 5. We train Qwen2.5-Math-1.5B with DeepSpeed-Zero1 (Rajbhandari et al., 2020) on 4 A100 80GB GPUs for 8 hours per run. Hsu et al. (2024) is used for optimization.

Category	Section 5
Sequence Length	16,384
Learning Rate	2×10^{-5}
Global Batch (Effective)	128
Learning Rate Scheduler	Cosine Decay
Warmup Ratio	0.05
Training Epochs	3

Table 10: **SFT configuration details for Section 5.**

C Additional Results

From Tables 11 to 26, we report detailed evaluation results for 55 languages with varying models and test-time scaling methods.

Language	MT-MATH100	MT-AIME2024	M-IMO	M-MO
Afrikaans	47.47	20.00	11.11	
Albanian	45.45	10.00	4.00	
Arabic	38.38	30.00	11.11	
Bengali	37.37	3.33		
Bulgarian	39.39	13.33	7.41	
Catalan	50.51	23.33		
Chinese (Simplified)	63.64	26.67	18.52	40.00
Chinese (Traditional)	61.62	20.00	18.52	
Croatian	49.49	20.00	7.41	
Czech	44.44	13.33	14.81	6.67
Danish	53.54	16.67	22.22	
Dutch	50.51	36.67	11.11	20.00
Estonian	39.39	10.00	4.00	
Finnish	41.41	16.67	8.00	
French	62.63	30.00	18.52	51.61
German	47.47	26.67	11.11	10.00
Greek	33.33	13.33	5.26	
Gujarati	39.39	10.00		
Hebrew	38.38	13.33	3.70	
Hindi	35.35	6.67		
Hungarian	51.52	10.00	8.00	
Indonesian	56.57	16.67	14.29	
Italian	51.52	20.00	20.00	
Japanese	56.57	16.67	8.00	0.00
Kannada	37.37	10.00		
Korean	44.44	13.33	3.70	36.67
Latvian	40.40	10.00	12.00	
Lithuanian	45.45	6.67	18.52	
Macedonian	43.43	10.00	11.11	
Malayalam	43.43	23.33		
Marathi	34.34	13.33		
Nepali	36.36	6.67		
Norwegian	53.54	23.33	11.11	
Persian	38.38	10.00		
Polish	54.55	26.67	14.81	26.67
Portuguese	55.56	10.00	24.00	
Punjabi	37.37	16.67		
Romanian	49.49	13.33	25.93	
Russian	59.60	20.00	16.00	20.00
Slovak	48.48	20.00	11.11	6.67
Slovenian	49.49	10.00	14.81	
Somali	42.42	23.33		
Spanish	55.56	20.00	18.52	
Swahili	34.34	16.67		
Swedish	58.59	20.00	8.00	
Tagalog	46.46	16.67		
Tamil	38.38	10.00		
Telugu	39.39	6.67		
Thai	39.39	23.33	3.70	
Turkish	43.43	13.33	7.41	
Ukrainian	38.38	13.33	11.11	
Urdu	35.35	20.00		
Vietnamese	44.44	13.33	7.41	
Welsh	39.39	16.67		
English	67.68	20.00	18.52	56.67
Average	46.01	16.36	12.23	25.00
Standard Deviation	8.61	6.89	6.02	19.10
Fleiss' Kappa	0.56	0.68	0.24	

Table 11: Evaluation results of Qwen2.5-Math-1.5B-Instruct with greedy decoding on MCLM.

Language	ORM (K=2)		ORM (K=4)		ORM (K=8)	
	MT-MATH100	MT-AIME2024	MT-MATH100	MT-AIME2024	MT-MATH100	MT-AIME2024
Afrikaans	53.54	23.33	56.57	16.67	60.61	23.33
Albanian	52.53	10.00	50.51	10.00	47.47	13.33
Arabic	43.43	20.00	46.46	13.33	51.52	16.67
Bengali	41.41	10.00	40.40	10.00	41.41	13.33
Bulgarian	45.45	26.67	46.46	20.00	51.52	16.67
Catalan	59.60	33.33	63.64	33.33	61.62	26.67
Chinese (Simplified)	69.70	36.67	76.77	30.00	78.79	26.67
Chinese (Traditional)	68.69	13.33	70.71	20.00	74.75	26.67
Croatian	51.52	16.67	59.60	23.33	58.59	30.00
Czech	49.49	13.33	56.57	10.00	59.60	16.67
Danish	53.54	23.33	56.57	20.00	59.60	26.67
Dutch	51.52	30.00	57.58	26.67	63.64	23.33
Estonian	46.46	13.33	48.48	13.33	50.51	13.33
Finnish	41.41	13.33	48.48	20.00	53.54	20.00
French	64.65	40.00	68.69	33.33	73.74	30.00
German	54.55	23.33	63.64	23.33	64.65	30.00
Greek	39.39	13.33	44.44	10.00	47.47	10.00
Gujarati	44.44	10.00	43.43	16.67	47.47	13.33
Hebrew	44.44	16.67	46.46	13.33	49.49	10.00
Hindi	40.40	10.00	45.45	13.33	47.47	16.67
Hungarian	53.54	10.00	57.58	10.00	63.64	16.67
Indonesian	58.59	20.00	56.57	20.00	59.60	16.67
Italian	57.58	26.67	60.61	26.67	69.70	16.67
Japanese	59.60	16.67	66.67	23.33	70.71	26.67
Kannada	45.45	10.00	47.47	16.67	52.53	13.33
Korean	53.54	16.67	56.57	23.33	57.58	13.33
Latvian	45.45	10.00	51.52	20.00	54.55	16.67
Lithuanian	48.48	10.00	52.53	10.00	57.58	13.33
Macedonian	50.51	13.33	51.52	13.33	50.51	10.00
Malayalam	47.47	20.00	52.53	20.00	56.57	23.33
Marathi	39.39	13.33	43.43	23.33	43.43	20.00
Nepali	38.38	6.67	46.46	3.33	46.46	6.67
Norwegian	59.60	26.67	61.62	16.67	65.66	23.33
Persian	40.40	13.33	41.41	13.33	39.39	16.67
Polish	54.55	16.67	57.58	16.67	64.65	16.67
Portuguese	58.59	13.33	60.61	13.33	62.63	26.67
Punjabi	41.41	16.67	43.43	20.00	42.42	16.67
Romanian	51.52	23.33	54.55	23.33	56.57	20.00
Russian	60.61	20.00	65.66	23.33	68.69	23.33
Slovak	52.53	10.00	54.55	20.00	55.56	33.33
Slovenian	47.47	16.67	51.52	20.00	54.55	30.00
Somali	44.44	16.67	46.46	16.67	46.46	10.00
Spanish	58.59	23.33	65.66	26.67	68.69	30.00
Swahili	37.37	13.33	41.41	20.00	45.45	13.33
Swedish	57.58	20.00	59.60	23.33	60.61	20.00
Tagalog	50.51	16.67	55.56	20.00	57.58	23.33
Tamil	41.41	16.67	44.44	16.67	47.47	16.67
Telugu	42.42	13.33	46.46	20.00	48.48	20.00
Thai	44.44	10.00	49.49	20.00	57.58	13.33
Turkish	50.51	16.67	46.46	13.33	54.55	20.00
Ukrainian	44.44	23.33	51.52	16.67	52.53	26.67
Urdu	38.38	16.67	41.41	16.67	44.44	20.00
Vietnamese	49.49	23.33	50.51	30.00	52.53	33.33
Welsh	38.38	16.67	44.44	16.67	44.44	20.00
English	71.72	16.67	73.74	26.67	76.77	36.67
Average	50.01	17.64	53.50	18.85	56.25	20.12
Standard Deviation	8.47	7.05	8.83	6.23	9.50	6.97
Fleiss' Kappa	0.57	0.66	0.60	0.64	0.61	0.63

Table 12: Evaluation results of Qwen2.5-Math-1.5B-Instruct with Best-of-N ($K = 2, 4, 8$) using Qwen2.5-Math-RM-72B as ORM on MT-MATH100 and MT-AIME2024.

Language	PRM (S=3, c=3)		PRM (S=4, c=5)		PRM (S=5, c=8)			
	MT-MATH100	MT-AIME2024	MT-MATH100	MT-AIME2024	MT-MATH100	MT-AIME2024	M-IMO	M-MO
Afrikaans	52.53	6.67	57.58	20.00	64.65	10.00	22.73	
Albanian	44.44	13.33	52.53	10.00	45.45	16.67	11.54	
Arabic	41.41	13.33	52.53	13.33	45.45	10.00	7.41	
Bengali	40.40	13.33	44.44	13.33	41.41	16.67		
Bulgarian	42.42	20.00	42.42	10.00	55.56	10.00	11.11	
Catalan	55.56	10.00	66.67	26.67	61.62	26.67		
Chinese (Simplified)	64.65	13.33	75.76	16.67	71.72	33.33	25.93	53.33
Chinese (Traditional)	63.64	26.67	73.74	16.67	72.73	26.67	29.63	
Croatian	50.51	13.33	51.52	20.00	54.55	23.33	14.81	
Czech	50.51	10.00	52.53	16.67	58.59	20.00	14.81	10.00
Danish	57.58	10.00	60.61	30.00	60.61	20.00	22.22	
Dutch	56.57	20.00	56.57	26.67	59.60	20.00	7.41	20.00
Estonian	47.47	13.33	51.52	3.33	49.49	10.00	11.54	
Finnish	41.41	10.00	43.43	6.67	49.49	10.00	15.38	
French	62.63	13.33	65.66	30.00	70.71	20.00	18.52	51.61
German	54.55	40.00	62.63	30.00	58.59	23.33	22.22	16.67
Greek	42.42	13.33	39.39	6.67	44.44	20.00	4.35	
Gujarati	42.42	6.67	39.39	13.33	41.41	13.33		
Hebrew	46.46	6.67	42.42	23.33	47.47	6.67	7.41	
Hindi	39.39	10.00	46.46	20.00	47.47	10.00		
Hungarian	57.58	26.67	61.62	10.00	57.58	3.33	19.23	
Indonesian	56.57	16.67	57.58	13.33	64.65	13.33	20.83	
Italian	61.62	13.33	61.62	20.00	67.68	23.33	23.08	
Japanese	64.65	20.00	66.67	26.67	66.67	16.67	15.38	7.14
Kannada	44.44	23.33	42.42	13.33	47.47	13.33		
Korean	46.46	10.00	45.45	13.33	50.51	13.33	14.81	26.67
Latvian	47.47	6.67	50.51	16.67	51.52	10.00	15.38	
Lithuanian	42.42	10.00	49.49	6.67	45.45	16.67	14.81	
Macedonian	41.41	13.33	47.47	16.67	48.48	23.33	11.11	
Malayalam	38.38	16.67	42.42	16.67	43.43	13.33		
Marathi	39.39	10.00	43.43	10.00	36.36	13.33		
Nepali	41.41	16.67	41.41	26.67	42.42	10.00		
Norwegian	59.60	23.33	65.66	30.00	59.60	26.67	18.52	
Persian	37.37	20.00	43.43	13.33	39.39	13.33		
Polish	49.49	23.33	58.59	23.33	62.63	20.00	25.93	36.67
Portuguese	58.59	20.00	57.58	16.67	61.62	30.00	19.23	
Punjabi	39.39	20.00	40.40	13.33	49.49	6.67		
Romanian	57.58	16.67	55.56	13.33	57.58	10.00	22.22	
Russian	53.54	23.33	65.66	23.33	64.65	20.00	15.38	23.33
Slovak	51.52	10.00	52.53	13.33	53.54	20.00	14.81	
Slovenian	44.44	23.33	47.47	16.67	45.45	26.67	11.11	
Somali	43.43	6.67	42.42	23.33	40.40	3.33		
Spanish	60.61	16.67	65.66	26.67	72.73	30.00	29.63	
Swahili	38.38	13.33	41.41	13.33	41.41	10.00		
Swedish	55.56	13.33	57.58	13.33	57.58	20.00	15.38	
Tagalog	47.47	20.00	51.52	10.00	55.56	10.00		
Tamil	41.41	10.00	45.45	16.67	45.45	16.67		
Telugu	42.42	6.67	45.45	13.33	48.48	16.67		
Thai	39.39	6.67	47.47	6.67	50.51	10.00	14.81	
Turkish	45.45	13.33	50.51	23.33	45.45	10.00	11.11	
Ukrainian	39.39	6.67	45.45	23.33	51.52	6.67	18.52	
Urdu	39.39	20.00	40.40	16.67	42.42	13.33		
Vietnamese	47.47	26.67	53.54	20.00	51.52	13.33	29.63	
Welsh	43.43	10.00	48.48	6.67	51.52	6.67		
English	73.74	26.67	79.80	23.33	72.73	23.33	29.63	60.00
Average	48.87	15.33	52.54	17.15	53.54	16.00	17.31	30.54
Standard Deviation	8.76	6.93	9.98	6.95	9.71	7.15	6.44	18.88
Fleiss' Kappa	0.57	0.78	0.58	0.61	0.60	0.62	0.43	

Table 13: Evaluation results of Qwen2.5-Math-1.5B-Instruct using Qwen2.5-Math-PRM-72B as PRM on MCLM.

Language	MT-MATH100		
	PRM (S=7, c=5)	**PRM (S=7, c=7)**	**PRM (S=7, c=11)**
Afrikaans	55.56	51.52	58.59
Arabic	44.44	42.42	44.44
Chinese (Simplified)	71.72	74.75	76.77
French	64.65	72.73	69.70
German	57.58	58.59	58.59
Hebrew	46.46	39.39	44.44
Indonesian	59.60	62.63	61.62
Italian	60.61	60.61	58.59
Japanese	67.68	67.68	63.64
Korean	48.48	45.45	50.51
Spanish	64.65	67.68	68.69
Turkish	50.51	53.54	48.48
Vietnamese	51.52	49.49	51.52
English	75.76	79.80	74.75
Average	58.51	59.02	59.31
Standard Deviation	9.62	12.57	10.60
Fleiss' Kappa	0.56	0.57	0.56

Table 14: Evaluation results of Qwen2.5-Math-1.5B-Instruct using Qwen2.5-Math-PRM-72B as PRM with steps fixed at $(S = 7)$ on MT-MATH100.

Language	MT-MATH100		
	PRM (S=3, c=8)	**PRM (S=6, c=8)**	**PRM (S=9, c=8)**
Afrikaans	54.55	55.56	60.61
Arabic	41.41	44.44	52.53
Chinese (Simplified)	71.72	71.72	70.71
French	67.68	64.65	67.68
German	56.57	57.58	66.67
Hebrew	42.42	46.46	45.45
Indonesian	60.61	59.60	62.63
Italian	56.57	60.61	61.62
Japanese	63.64	67.68	62.63
Korean	47.47	48.48	48.48
Spanish	65.66	64.65	72.73
Turkish	53.54	50.51	49.49
Vietnamese	57.58	51.52	57.58
English	75.76	75.76	77.78
Average	58.23	58.51	61.18
Standard Deviation	10.22	9.62	9.65
Fleiss' Kappa	0.56	0.58	0.58

Table 15: Evaluation results of Qwen2.5-Math-1.5B-Instruct using Qwen2.5-Math-PRM-72B as PRM with the number of candidates fixed at 8, on MT-MATH100.

Language	MT-MATH100		
	PRM (S=7, c=7)	**PRM (S=7, c=11)**	**PRM (S=7, c=18)**
Afrikaans	51.52	58.59	58.59
Arabic	42.42	44.44	52.53
Chinese (Simplified)	74.75	76.77	76.77
French	72.73	69.70	71.72
German	58.59	58.59	60.61
Hebrew	39.39	44.44	41.41
Indonesian	62.63	61.62	62.63
Italian	60.61	58.59	64.65
Japanese	67.68	63.64	61.62
Korean	45.45	50.51	50.51
Spanish	67.68	68.69	68.69
Turkish	53.54	48.48	52.53
Vietnamese	49.49	51.52	51.52
English	79.80	74.75	70.71
Average	59.02	59.31	60.32
Standard Deviation	12.57	10.60	9.84
Fleiss' Kappa	0.52	0.55	0.54

Table 16: Evaluation results of Qwen2.5-Math-1.5B-Instruct using Qwen2.5-Math-PRM-7B as PRM with the number of candidates fixed at 7, on MT-MATH100.

Language	MT-MATH100		
	PRM (S=3, c=13)	**PRM (S=6, c=13)**	**PRM (S=9, c=13)**
Afrikaans	55.56	59.60	54.55
Arabic	44.44	45.45	44.44
Chinese (Simplified)	75.76	70.71	79.80
French	64.65	71.72	73.74
German	55.56	63.64	61.62
Hebrew	46.46	43.43	47.47
Indonesian	56.57	58.59	61.62
Italian	62.63	60.61	61.62
Japanese	58.59	67.68	59.60
Korean	49.49	48.48	51.52
Spanish	60.61	73.74	64.65
Turkish	49.49	50.51	49.49
Vietnamese	52.53	48.48	45.45
English	71.72	73.74	77.78
Average	57.43	59.74	59.52
Standard Deviation	9.10	10.90	11.59
Fleiss' Kappa	0.54	0.55	0.52

Table 17: Evaluation results of Qwen2.5-Math-1.5B-Instruct using Qwen2.5-Math-PRM-7B as PRM with the number of candidates fixed at 13, on MT-MATH100.

Language	MT-MATH100	MT-AIME2024	M-IMO	M-MO
Afrikaans	47.47	36.67	5.56	
Albanian	31.31	13.33	8.00	
Arabic	36.36	23.33	7.41	
Bengali	33.33	10.00		
Bulgarian	41.41	10.00	11.11	
Catalan	47.47	16.67		
Chinese (Simplified)	57.58	23.33	18.52	23.33
Chinese (Traditional)	43.43	16.67	22.22	
Croatian	38.38	16.67	7.41	
Czech	33.33	30.00	3.70	3.33
Danish	41.41	23.33	7.41	
Dutch	45.45	16.67	7.41	16.67
Estonian	38.38	10.00	12.00	
Finnish	30.30	23.33	12.00	
French	39.39	6.67	7.41	35.48
German	45.45	23.33	18.52	6.67
Greek	30.30	16.67	0.00	
Gujarati	27.27	6.67		
Hebrew	36.36	16.67	7.41	
Hindi	36.36	10.00		
Hungarian	39.39	16.67	8.00	
Indonesian	37.37	13.33	4.76	
Italian	41.41	13.33	12.00	
Japanese	45.45	20.00	12.00	3.57
Kannada	32.32	10.00		
Korean	39.39	16.67	14.81	16.67
Latvian	30.30	6.67	4.00	
Lithuanian	31.31	6.67	14.81	
Macedonian	31.31	0.00	7.41	
Malayalam	27.27	13.33		
Marathi	33.33	13.33		
Nepali	35.35	13.33		
Norwegian	37.37	16.67	11.11	
Persian	29.29	20.00		
Polish	38.38	6.67	11.11	13.33
Portuguese	47.47	20.00	8.00	
Punjabi	29.29	16.67		
Romanian	41.41	10.00	18.52	
Russian	46.46	16.67	12.00	20.00
Slovak	35.35	16.67	11.11	10.00
Slovenian	35.35	23.33	11.11	
Somali	26.26	16.67		
Spanish	46.46	16.67	11.11	
Swahili	36.36	6.67		
Swedish	39.39	13.33	8.00	
Tagalog	35.35	13.33		
Tamil	33.33	10.00		
Telugu	34.34	13.33		
Thai	30.30	10.00	7.41	
Turkish	42.42	6.67	11.11	
Ukrainian	35.35	3.33	11.11	
Urdu	28.28	13.33		
Vietnamese	31.31	10.00	7.41	
Welsh	30.30	23.33		
English	65.66	20.00	25.93	53.33
Average	37.47	14.85	10.50	18.40
Standard Deviation	7.56	6.69	5.16	14.92
Fleiss' Kappa	0.41	0.13	0.19	

Table 18: Evaluation results of Qwen2.5-Math-1.5B-Instruct + SFT on MCLM.

Language	MT-MATH100	MT-AIME2024	M-IMO	M-MO
Afrikaans	39.39	10.00	13.64	
Albanian	39.39	16.67	7.69	
Arabic	41.41	16.67	14.81	
Bengali	39.39	30.00		
Bulgarian	42.42	10.00	11.11	
Catalan	51.52	26.67		
Chinese (Simplified)	50.51	23.33	7.41	13.33
Chinese (Traditional)	52.53	20.00	11.11	
Croatian	38.38	13.33	11.11	
Czech	51.52	23.33	11.11	10.00
Danish	40.40	6.67	3.70	
Dutch	48.48	20.00	11.11	20.00
Estonian	37.37	23.33	15.38	
Finnish	40.40	20.00	7.69	
French	46.46	10.00	7.41	32.26
German	49.49	10.00	7.41	3.33
Greek	28.28	20.00	17.39	
Gujarati	42.42	13.33		
Hebrew	39.39	13.33	3.70	
Hindi	45.45	13.33		
Hungarian	43.43	40.00	11.54	
Indonesian	51.52	16.67	16.67	
Italian	48.48	13.33	11.54	
Japanese	50.51	6.67	11.54	3.57
Kannada	32.32	10.00		
Korean	55.56	10.00	11.11	26.67
Latvian	42.42	10.00	15.38	
Lithuanian	36.36	13.33	7.41	
Macedonian	39.39	13.33	18.52	
Malayalam	34.34	26.67		
Marathi	37.37	23.33		
Nepali	42.42	16.67		
Norwegian	42.42	10.00	3.70	
Persian	47.47	10.00		
Polish	38.38	10.00	14.81	20.00
Portuguese	50.51	26.67	11.54	
Punjabi	29.29	16.67		
Romanian	45.45	6.67	11.11	
Russian	57.58	13.33	7.69	36.67
Slovak	47.47	20.00	7.41	
Slovenian	39.39	23.33	18.52	
Somali	22.22	26.67		
Spanish	44.44	16.67	0.00	
Swahili	34.34	6.67		
Swedish	42.42	10.00	3.85	
Tagalog	35.35	6.67		
Tamil	36.36	23.33		
Telugu	36.36	13.33		
Thai	34.34	26.67	14.81	
Turkish	39.39	23.33	7.41	
Ukrainian	49.49	10.00	7.41	
Urdu	32.32	20.00		
Vietnamese	47.47	10.00	18.52	
Welsh	28.28	20.00		
English	51.52	26.67	7.41	40.00
Average	42.02	16.67	10.52	20.58
Standard Deviation	7.46	7.31	4.63	13.17
Fleiss' Kappa	0.40	0.13	0.25	

Table 19: Evaluation results of Qwen2.5-Math-1.5B-Instruct + MT-SFT on MCLM.

Language	MT-MATH100	MT-AIME2024	M-IMO	M-MO
Afrikaans	58.59	20.00	11.11	
Albanian	46.46	30.00	16.00	
Arabic	51.52	20.00	18.52	
Bengali	56.57	10.00		
Bulgarian	57.58	16.67	11.11	
Catalan	64.65	30.00		
Chinese (Simplified)	69.70	16.67	25.93	33.33
Chinese (Traditional)	67.68	20.00	18.52	
Croatian	59.60	36.67	18.52	
Czech	57.58	33.33	18.52	16.67
Danish	56.57	16.67	14.81	
Dutch	64.65	30.00	22.22	23.33
Estonian	39.39	6.67	12.00	
Finnish	52.53	16.67	20.00	
French	63.64	26.67	29.63	48.39
German	63.64	16.67	25.93	26.67
Greek	38.38	13.33	10.53	
Gujarati	47.47	3.33		
Hebrew	61.62	23.33	7.41	
Hindi	61.62	23.33		
Hungarian	55.56	26.67	24.00	
Indonesian	69.70	13.33	23.81	
Italian	69.70	36.67	28.00	
Japanese	62.63	16.67	12.00	3.57
Kannada	42.42	16.67		
Korean	61.62	20.00	11.11	30.00
Latvian	49.49	6.67	20.00	
Lithuanian	40.40	23.33	14.81	
Macedonian	59.60	23.33	25.93	
Malayalam	41.41	3.33		
Marathi	39.39	23.33		
Nepali	50.51	10.00		
Norwegian	67.68	13.33	18.52	
Persian	61.62	13.33		
Polish	62.63	16.67	22.22	23.33
Portuguese	75.76	23.33	16.00	
Punjabi	42.42	13.33		
Romanian	58.59	26.67	22.22	
Russian	68.69	33.33	20.00	26.67
Slovak	58.59	13.33	11.11	20.00
Slovenian	56.57	30.00	14.81	
Somali	30.30	20.00		
Spanish	69.70	30.00	25.93	
Swahili	42.42	20.00		
Swedish	54.55	13.33	20.00	
Tagalog	47.47	23.33		
Tamil	40.40	16.67		
Telugu	36.36	23.33		
Thai	59.60	13.33	29.63	
Turkish	61.62	36.67	22.22	
Ukrainian	67.68	16.67	18.52	
Urdu	50.51	20.00		
Vietnamese	61.62	13.33	33.33	
Welsh	34.34	16.67		
English	67.68	20.00	14.81	66.67
Average	55.61	19.94	19.20	28.97
Standard Deviation	10.93	8.10	6.24	16.64
Fleiss' Kappa	0.47	0.30	0.19	

Table 20: Evaluation results of DeepSeek-R1-1.5B + MT-SFT on MCLM.

Language	BF (N=2048)	BF (N=4096)	BF (N=8192)			
	MT-AIME2024	MT-AIME2024	MT-MATH100	MT-AIME2024	M-IMO	M-MO
Afrikaans	23.33	23.33	59.60	30.00	9.09	
Albanian	23.33	26.67	48.48	26.67	7.69	
Arabic	16.67	23.33	60.61	26.67	14.81	
Bengali	33.33	30.00	54.55	23.33		
Bulgarian	33.33	33.33	61.62	26.67	22.22	
Catalan	20.00	43.33	64.65	43.33		
Chinese (Simplified)	20.00	16.67	69.70	16.67	22.22	40.00
Chinese (Traditional)	26.67	26.67	70.71	36.67	18.52	
Croatian	30.00	30.00	60.61	30.00	37.04	
Czech	40.00	20.00	62.63	20.00	29.63	33.33
Danish	30.00	33.33	61.62	30.00	22.22	
Dutch	10.00	23.33	70.71	36.67	25.93	20.00
Estonian	23.33	16.67	40.40	20.00	15.38	
Finnish	20.00	33.33	51.52	20.00	30.77	
French	16.67	23.33	72.73	16.67	25.93	51.61
German	26.67	20.00	75.76	26.67	25.93	30.00
Greek	6.67	13.33	42.42	16.67	21.74	
Gujarati	16.67	16.67	51.52	16.67		
Hebrew	33.33	23.33	60.61	16.67	14.81	
Hindi	26.67	10.00	61.62	20.00		
Hungarian	30.00	26.67	58.59	23.33	26.92	
Indonesian	10.00	30.00	73.74	30.00	25	
Italian	20.00	26.67	74.75	36.67	23.08	
Japanese	20.00	16.67	63.64	36.67	23.08	7.14
Kannada	10.00	13.33	49.49	10.00		
Korean	16.67	23.33	64.65	20.00	11.11	40.00
Latvian	30.00	20.00	52.53	10.00	23.08	
Lithuanian	10.00	6.67	46.46	26.67	18.52	
Macedonian	20.00	20.00	63.64	23.33	25.93	
Malayalam	10.00	13.33	51.52	13.33		
Marathi	20.00	26.67	51.52	23.33		
Nepali	30.00	13.33	54.55	20.00		
Norwegian	26.67	26.67	65.66	20.00	18.52	
Persian	26.67	23.33	62.63	36.67		
Polish	23.33	20.00	66.67	16.67	14.81	23.33
Portuguese	20.00	26.67	79.80	20.00	15.38	
Punjabi	23.33	26.67	51.52	20.00		
Romanian	30.00	23.33	60.61	10.00	22.22	
Russian	36.67	30.00	72.73	30.00	23.08	30.00
Slovak	40.00	23.33	66.67	30.00	25.93	
Slovenian	20.00	20.00	60.61	33.33	25.93	
Somali	20.00	16.67	35.35	16.67		
Spanish	30.00	30.00	71.72	40.00	18.52	
Swahili	13.33	13.33	41.41	30.00		
Swedish	13.33	16.67	62.63	23.33	19.23	
Tagalog	10.00	20.00	52.53	23.33		
Tamil	26.67	20.00	44.44	23.33		
Telugu	13.33	16.67	44.44	20.00		
Thai	26.67	13.33	64.65	23.33	11.11	
Turkish	20.00	16.67	61.62	16.67	33.33	
Ukrainian	30.00	26.67	73.74	23.33	22.22	
Urdu	23.33	20.00	46.46	20.00		
Vietnamese	20.00	26.67	62.63	40.00	25.93	
Welsh	20.00	16.67	42.42	13.33		
English	20.00	26.67	71.72	40.00	22.22	76.67
Average	22.48	22.24	59.45	24.42	21.55	35.21
Standard Deviation	7.94	6.85	10.52	8.32	6.44	19.01
Fleiss' Kappa	0.33	0.37	0.44	0.32	0.19	

Table 21: Evaluation results of Qwen2.5-Math-1.5B-Instruct with Budget Forcing ($BF = 2048, 4096, 8192$).

Language	MT-MATH100	MT-AIME2024	M-IMO	M-MO
Afrikaans	72.73	13.33	27.78	
Albanian	60.61	16.67	20	
Arabic	76.77	13.33	14.81	
Bengali	72.73	16.67		
Bulgarian	72.73	16.67		
Catalan	73.74	20.00		
Chinese (Simplified)	77.78	20.00	7.41	56.67
Chinese (Traditional)	73.74	23.33	11.11	
Croatian	73.74	30.00	22.22	
Czech	75.76	20.00	11.11	16.67
Danish	72.73	23.33	18.52	
Dutch	77.78	16.67	18.52	23.33
Estonian	57.58	13.33	20	
Finnish	70.71	20.00	16	
French	77.78	20.00	25.93	48.39
German	76.77	23.33	25.93	26.67
Greek	64.65	13.33	10.53	
Gujarati	55.56	16.67		
Hebrew	71.72	20.00	7.41	
Hindi	70.71	30.00		
Hungarian	71.72	26.67	20	
Indonesian	69.70	20.00	19.05	
Italian	78.79	23.33	12	
Japanese	76.77	23.33	16	3.57
Kannada	57.58	20.00		40
Korean	77.78	20.00	14.81	
Latvian	59.60	13.33	20	
Lithuanian	61.62	16.67	25.93	
Macedonian	77.78	16.67	22.22	
Malayalam	56.57	10.00		
Marathi	63.64	16.67		
Nepali	67.68	20.00		
Norwegian	73.74	23.33	22.22	
Persian	74.75	30.00		
Polish	71.72	16.67	22.22	26.67
Portuguese	78.79	26.67	20	
Punjabi	58.59	16.67		
Romanian	76.77	23.33	14.81	
Russian	77.78	20.00	20	43.33
Slovak	74.75	23.33	18.52	23.33
Slovenian	71.72	23.33	14.81	
Somali	38.38	6.67		
Spanish	75.76	30.00	14.81	
Swahili	46.46	13.33		
Swedish	76.77	16.67	24	
Tagalog	60.61	16.67		
Tamil	54.55	10.00		
Telugu	60.61	16.67		
Thai	73.74	20.00	14.81	
Turkish	70.71	20.00	7.41	
Ukrainian	76.77	23.33	14.81	
Urdu	63.64	50.00		
Vietnamese	76.77	26.67	14.81	
Welsh	50.51	20.00		
English	83.84	20.00	22.22	46.67
Average	69.33	20.12	17.64	32.30
Standard Deviation	9.42	6.57	5.38	15.92
Fleiss Kappa	0.61	0.51	0.38	15.81

Table 22: Evaluation results of Qwen2.5-Math-7B-Instruct with greedy decoding on MCLM.

Language	ORM (K=2)		ORM (K=4)		ORM (K=8)	
	MT-MATH100	MT-AIME2024	MT-MATH100	MT-AIME2024	MT-MATH100	MT-AIME2024
Afrikaans	74.75	16.67	73.74	26.67	76.77	33.33
Albanian	68.69	20.00	65.66	26.67	68.69	26.67
Arabic	76.77	13.33	82.83	23.33	83.84	20.00
Bengali	69.70	16.67	75.76	16.67	74.75	16.67
Bulgarian	73.74	16.67	77.78	20.00	79.80	16.67
Catalan	75.76	26.67	77.78	20.00	76.77	30.00
Chinese_(Simplified)	77.78	20.00	81.82	26.67	82.83	26.67
Chinese_(Traditional)	77.78	23.33	81.82	23.33	81.82	23.33
Croatian	75.76	30.00	78.79	33.33	78.79	33.33
Czech	75.76	20.00	81.82	23.33	81.82	23.33
Danish	73.74	26.67	72.73	43.33	74.75	43.33
Dutch	76.77	20.00	78.79	26.67	81.82	40.00
Estonian	62.63	16.67	64.65	23.33	65.66	30.00
Finnish	73.74	23.33	77.78	33.33	75.76	33.33
French	81.82	23.33	81.82	20.00	81.82	26.67
German	78.79	33.33	81.82	40.00	83.84	40.00
Greek	65.66	20.00	67.68	23.33	70.71	16.67
Gujarati	58.59	13.33	59.60	20.00	64.65	16.67
Hebrew	73.74	13.33	75.76	20.00	76.77	30.00
Hindi	70.71	26.67	75.76	26.67	75.76	36.67
Hungarian	73.74	26.67	76.77	20.00	76.77	23.33
Indonesian	75.76	30.00	76.77	33.33	77.78	43.33
Italian	79.80	26.67	79.80	26.67	82.83	33.33
Japanese	78.79	23.33	79.80	30.00	80.81	23.33
Kannada	55.56	13.33	57.58	13.33	59.60	20.00
Korean	79.80	16.67	76.77	23.33	77.78	26.67
Latvian	61.62	16.67	65.66	10.00	66.67	10.00
Lithuanian	63.64	20.00	68.69	30.00	69.70	20.00
Macedonian	76.77	16.67	80.81	20.00	79.80	23.33
Malayalam	59.60	10.00	62.63	16.67	68.69	23.33
Marathi	65.66	26.67	68.69	20.00	69.70	16.67
Nepali	64.65	13.33	69.70	16.67	68.69	16.67
Norwegian	72.73	26.67	74.75	30.00	76.77	33.33
Persian	76.77	23.33	75.76	23.33	76.77	16.67
Polish	77.78	10.00	78.79	10.00	78.79	16.67
Portuguese	81.82	26.67	80.81	36.67	83.84	40.00
Punjabi	58.59	20.00	59.60	16.67	62.63	26.67
Romanian	79.80	23.33	81.82	26.67	79.80	30.00
Russian	78.79	26.67	82.83	20.00	86.87	26.67
Slovak	77.78	30.00	79.80	33.33	81.82	30.00
Slovenian	73.74	13.33	78.79	20.00	78.79	23.33
Somali	38.38	6.67	42.42	13.33	44.44	20.00
Spanish	75.76	26.67	78.79	26.67	81.82	30.00
Swahili	48.48	13.33	49.49	20.00	51.52	23.33
Swedish	77.78	30.00	76.77	30.00	77.78	30.00
Tagalog	58.59	13.33	65.66	10.00	66.67	16.67
Tamil	59.60	16.67	65.66	10.00	62.63	10.00
Telugu	61.62	20.00	63.64	23.33	62.63	16.67
Thai	76.77	16.67	79.80	23.33	77.78	30.00
Turkish	76.77	26.67	79.80	26.67	79.80	26.67
Ukrainian	77.78	23.33	78.79	23.33	79.80	26.67
Urdu	66.67	33.33	67.68	30.00	72.73	30.00
Vietnamese	73.74	33.33	76.77	33.33	80.81	36.67
Welsh	51.52	20.00	53.54	16.67	56.57	6.67
English	84.85	26.67	84.85	30.00	86.87	26.67
Average	70.98	21.21	73.35	23.82	74.62	25.76
Standard Deviation	9.46	6.52	9.20	7.41	8.86	8.37
Fleiss' Kappa	0.62	0.55	0.65	0.57	0.67	0.57

Table 23: Evaluation results of Qwen2.5-Math-7B-Instruct with Best-of-N ($K = 2, 4, 8$) using Qwen2.5-Math-RM-72B as ORM on MT-MATH100 and MT-AIME2024.

Language	PRM (S=3, c=3)		PRM (S=4, c=5)		PRM (S=5, c=8)	
	MT-MATH100	MT-AIME2024	MT-MATH100	MT-AIME2024	MT-MATH100	MT-AIME2024
Afrikaans	70.71	20.00	70.71	16.67	70.71	20.00
Albanian	60.61	16.67	62.63	33.33	61.62	26.67
Arabic	65.66	26.67	78.79	26.67	82.83	30.00
Bengali	67.68	16.67	70.71	10.00	68.69	23.33
Bulgarian	69.70	20.00	74.75	10.00	75.76	30.00
Catalan	72.73	16.67	70.71	20.00	71.72	16.67
Chinese (Simplified)	72.73	16.67	73.74	33.33	78.79	30.00
Chinese (Traditional)	71.72	16.67	76.77	20.00	77.78	23.33
Croatian	69.70	20.00	72.73	16.67	70.71	33.33
Czech	69.70	16.67	77.78	10.00	73.74	30.00
Danish	63.64	23.33	69.70	33.33	66.67	30.00
Dutch	71.72	6.67	72.73	26.67	75.76	26.67
Estonian	46.46	20.00	51.52	13.33	59.60	20.00
Finnish	64.65	16.67	66.67	13.33	72.73	33.33
French	73.74	20.00	72.73	16.67	76.77	26.67
German	73.74	10.00	68.69	10.00	76.77	26.67
Greek	63.64	16.67	64.65	13.33	67.68	13.33
Gujarati	56.57	13.33	56.57	26.67	55.56	13.33
Hebrew	66.67	10.00	68.69	20.00	75.76	26.67
Hindi	58.59	16.67	63.64	20.00	72.73	13.33
Hungarian	68.69	16.67	69.70	30.00	72.73	20.00
Indonesian	69.70	26.67	68.69	20.00	72.73	10.00
Italian	71.72	16.67	77.78	30.00	73.74	23.33
Japanese	71.72	23.33	75.76	16.67	76.77	13.33
Kannada	46.46	16.67	53.54	10.00	54.55	16.67
Korean	69.70	16.67	72.73	13.33	74.75	16.67
Latvian	59.60	10.00	63.64	13.33	63.64	16.67
Lithuanian	55.56	20.00	62.63	13.33	65.66	16.67
Macedonian	69.70	16.67	75.76	16.67	75.76	23.33
Malayalam	49.49	20.00	57.58	23.33	52.53	20.00
Marathi	56.57	20.00	55.56	23.33	57.58	23.33
Nepali	51.52	16.67	61.62	20.00	52.53	23.33
Norwegian	69.70	20.00	67.68	20.00	69.70	26.67
Persian	71.72	26.67	71.72	16.67	72.73	23.33
Polish	61.62	13.33	67.68	13.33	76.77	10.00
Portuguese	72.73	10.00	71.72	26.67	79.80	26.67
Punjabi	46.46	13.33	45.45	10.00	52.53	20.00
Romanian	66.67	13.33	70.71	33.33	77.78	30.00
Russian	75.76	16.67	76.77	16.67	76.77	33.33
Slovak	70.71	23.33	75.76	26.67	70.71	13.33
Slovenian	70.71	23.33	72.73	30.00	74.75	20.00
Somali	40.40	3.33	42.42	10.00	42.42	6.67
Spanish	71.72	13.33	77.78	20.00	80.81	23.33
Swahili	48.48	6.67	42.42	10.00	44.44	16.67
Swedish	70.71	16.67	76.77	36.67	71.72	26.67
Tagalog	55.56	23.33	59.60	16.67	58.59	13.33
Tamil	50.51	10.00	55.56	3.33	57.58	20.00
Telugu	53.54	13.33	58.59	20.00	54.55	20.00
Thai	67.68	10.00	71.72	16.67	71.72	26.67
Turkish	63.64	20.00	71.72	20.00	64.65	16.67
Ukrainian	75.76	20.00	77.78	26.67	79.80	20.00
Urdu	57.58	26.67	62.63	20.00	66.67	23.33
Vietnamese	72.73	23.33	73.74	13.33	73.74	33.33
Welsh	50.51	20.00	43.43	13.33	45.45	20.00
English	73.74	23.33	75.76	20.00	75.76	23.33
Average	64.17	17.27	67.09	19.27	68.45	22.00
Standard Deviation	9.25	5.33	9.65	7.61	10.02	6.56
Fleiss' Kappa	0.56	0.56	0.54	0.57	0.56	0.59

Table 24: Evaluation results of Qwen2.5-Math-7B-Instruct using Qwen2.5-Math-PRM-72B as PRM on MT-MATH100 and MT-AIME2024.

Language	MT-MATH100	MT-AIME2024	M-IMO	M-MO
Afrikaans	73.74	23.33	9.09	
Albanian	66.67	20.00	15.38	
Arabic	71.72	16.67	3.70	
Bengali	64.65	3.33		
Bulgarian	72.73	20.00	18.52	
Catalan	70.71	26.67		
Chinese (Simplified)	70.71	23.33	14.81	26.67
Chinese (Traditional)	69.70	23.33	11.11	
Croatian	72.73	16.67	18.52	
Czech	71.72	33.33	11.11	36.67
Danish	71.72	23.33	22.22	
Dutch	69.70	20.00	3.70	3.33
Estonian	76.77	16.67	15.38	
Finnish	72.73	6.67	15.38	
French	70.71	23.33	14.81	48.39
German	73.74	20.00	18.52	26.67
Greek	71.72	10.00	13.04	
Gujarati	67.68	13.33		
Hebrew	71.72	10.00	7.41	
Hindi	70.71	6.67		
Hungarian	73.74	26.67	11.54	
Indonesian	68.69	13.33	16.67	
Italian	72.73	23.33	11.54	
Japanese	70.71	30.00	7.69	7.14
Kannada	61.62	23.33		
Korean	72.73	26.67	22.22	36.67
Latvian	69.70	20.00	7.69	
Lithuanian	68.69	16.67	7.41	
Macedonian	71.72	20.00	22.22	
Malayalam	62.63	23.33		
Marathi	63.64	20.00		
Nepali	67.68	10.00		
Norwegian	75.76	30.00	11.11	
Persian	66.67	26.67		
Polish	72.73	13.33	22.22	26.67
Portuguese	70.71	26.67	7.69	
Punjabi	69.70	16.67		
Romanian	73.74	26.67	11.11	
Russian	73.74	23.33	15.38	50.00
Slovak	72.73	20.00	18.52	
Slovenian	72.73	16.67	7.41	
Somali	57.58	20.00		
Spanish	71.72	26.67	14.81	
Swahili	65.66	23.33		
Swedish	72.73	23.33	23.08	
Tagalog	71.72	20.00		
Tamil	67.68	20.00		
Telugu	66.67	16.67		
Thai	70.71	26.67	7.41	
Turkish	71.72	10.00	11.11	
Ukrainian	73.74	23.33	14.81	
Urdu	68.69	23.33		
Vietnamese	71.72	6.67	14.81	
Welsh	65.66	26.67		
English	75.76	33.33	7.41	50.00
Average	70.30	20.18	13.33	30.81
Standard Deviation	3.68	6.83	5.36	15.80
Fleiss' Kappa	0.71	0.33	0.25	

Table 25: Evaluation results of GPT-4O-MINI with greedy decoding on MCLM.

Language	MT-MATH100	MT-AIME2024	M-IMO	M-MO
Afrikaans	85.86	46.67	33.33	
Albanian	86.87	53.33	28.00	
Arabic	86.87	43.33	22.22	
Bengali	86.87	43.33		
Bulgarian	87.88	46.67	40.74	
Catalan	87.88	53.33		
Chinese (Simplified)	85.86	50	25.93	66.67
Chinese (Traditional)	84.85	40	29.63	
Croatian	84.85	46.67	33.33	
Czech	84.85	36.67	29.63	53.33
Danish	85.86	40	40.74	
Dutch	86.87	50	33.33	40.00
Estonian	83.84	50	28.00	
Finnish	84.85	40	28.00	
French	86.87	43.33	29.63	67.74
German	86.87	43.33	33.33	43.33
Greek	87.88	56.67	21.05	
Gujarati	83.84	46.67		
Hebrew	81.82	40	7.41	
Hindi	83.84	43.33		
Hungarian	86.87	53.33	28.00	
Indonesian	84.85	43.33	33.33	
Italian	82.83	50	36.00	
Japanese	86.87	50	16.00	17.86
Kannada	86.87	43.33		
Korean	77.78	46.67	25.93	60.00
Latvian	87.88	46.67	32.00	
Lithuanian	85.86	46.67	33.33	
Macedonian	83.84	43.33	33.33	
Malayalam	85.86	46.67		
Marathi	83.84	36.67		
Nepali	79.8	46.67		
Norwegian	82.83	53.33	22.22	
Persian	87.88	53.33		
Polish	81.82	43.33	37.04	40.00
Portuguese	82.83	36.67	36.00	
Punjabi	87.88	43.33		
Romanian	81.82	40	40.74	
Russian	85.86	56.67	20.00	50.00
Slovak	87.88	46.67	33.33	46.67
Slovenian	85.86	46.67	29.63	
Somali	87.88	50		
Spanish	72.73	50	29.63	
Swahili	86.87	43.33		
Swedish	79.8	43.33	28.00	
Tagalog	85.86	46.67		
Tamil	84.85	43.33		
Telugu	82.83	33.33		
Thai	84.85	40	22.22	
Turkish	84.85	40	33.33	
Ukrainian	84.85	50	29.63	
Urdu	84.85	36.67		
Vietnamese	85.86	46.67	37.04	
Welsh	85.86	46.67		
English	83.84	36.67	29.63	80.00
Average	84.89	45.33	29.75	51.42
Standard Deviation	2.80	5.35	6.86	16.94
Fleiss' Kappa	0.88	0.73	0.44	

Table 26: Evaluation results of O3-MINI with greedy decoding on MCLM.

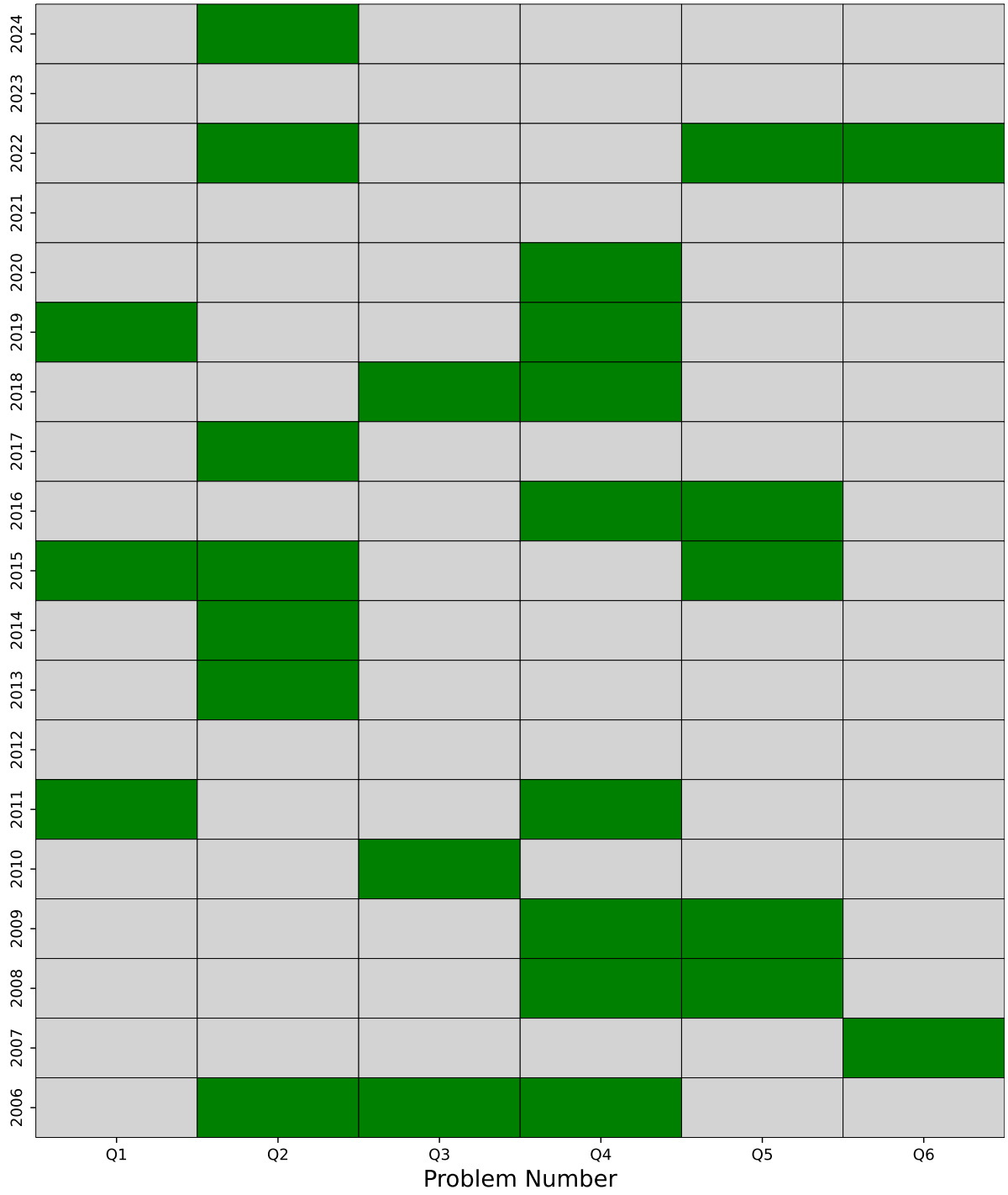

Figure 10: Heatmap representation of IMO problems from 2006 to 2024. Each row corresponds to a competition year, and each column represents a problem (Q1–Q6). Green cells indicate questions that have been included in the M-IMO subset, while gray cells represent problems that were not selected.

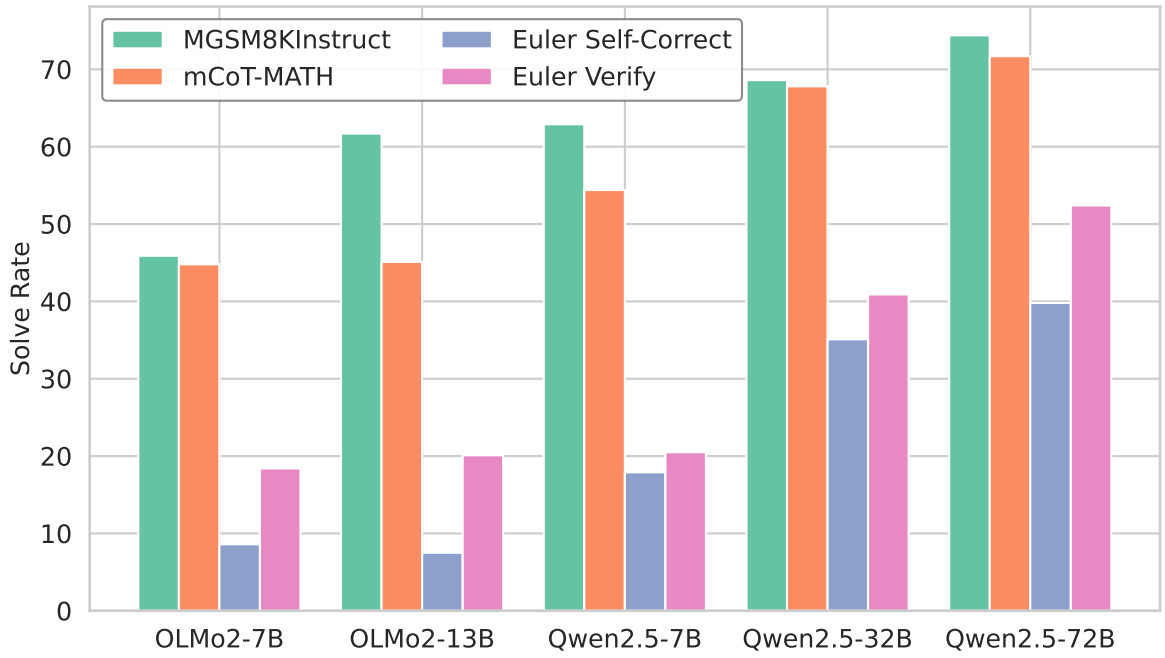

Figure 11: **Solve rates (%) of different multilingual math datasets evaluated.** For the OLMo2 series, we use the base models, while for the Qwen2.5 series, the instruct-tuned variants are used. EULER-INSTRUCT presents a significantly lower solve rate, indicating its greater difficulty.

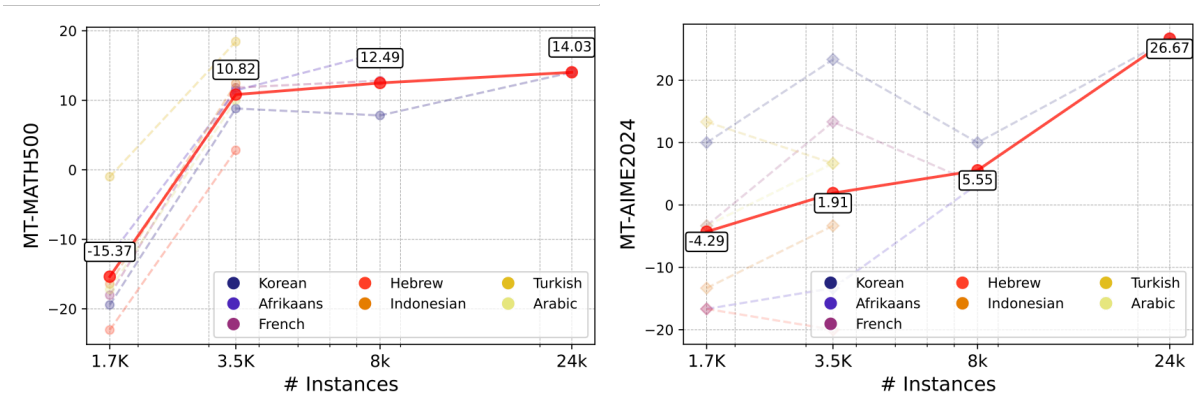

Figure 12: **Model Results from Table 9.** Left shows accuracy on MT-MATH500 (entire translated subset for language group (B)), and right shows average performance of MT-AIME2024.

```
You will be given an English question in the following format.

[Question]
<question>
{..question...}
</question>

Your job is to return a translated version of the question.
* Translate to <language>.
* The translation must be fluent, easy to read by native speakers.
* Do not solve the prompt translate it.
* You must preserve all details including math notations (latex) and code.
* The math notations and code must not be translated, keep it as is.
* Return you translation in the following format.

[Translation]
<translation>
{..translated question...}
</translation>

--------------------------------------------------
The following is the math problem for you task:

[Question]
<question>
<source_question>
</question>
```

Figure 13: Question Translation Template

```
You will be given an English solution in the following format.

[Solution]
<solution>
{..solution in English...}
</solution>

Your job is to rewrite the English solution to <language>.
* The solution must preserve the original structure and details.
* You must preserve all details including math notations (latex) and code.
* The math notations and code must not be translated, keep it as is.
* The solution must be natural, easy and polite for a native speaker to read.

[Translation]
<translation>
{..translated solution...}
</translation>

--------------------------------------------------
The following is the math problem and solution for your task:

[Solution]
<solution>
<source_solution>
</solution>
```

Figure 14: Solution Translation Template

```
You will be given a math problem, the correct answer, and a solution generated by a language model.
 Your task is to determine whether the solution generated by the model is correct.

[Question]
<question>
{..math question...}
</question>

[Correct Answer]
<answer>
{..correct answer...}
</answer>

[Model Solution]
<solution>
{..model-generated solution...}
</solution>

Instructions:
* Compare the model's solution with the correct answer.
* If the model's solution is correct, output [[TRUE]].
* If the model's solution is incorrect, output [[FALSE]].
* You do not have to judge the solution process; there are numerous possible 'Gold' solutions, and
 the model solution does not have to be identical with the one provided. As long as the model
reaches the correct answer, it is correct.
* Do not provide any explanations -- only return your judgment ONLY.

--------------------------------------------------
The following is the math problem and solution for your task:

[Question]
<question>
<math_question>
</question>

[Correct Answer]
<answer>
<correct_answer>
</answer>

[Model Solution]
<solution>
<model_solution>
</solution>
```

Figure 15: Judge Template