

mAceReason-Math: A Dataset of High-Quality Multilingual Math Problems Ready for RLVR

Konstantin Dobler^{*†}

Hasso Plattner Institute &

ELLIS Unit Potsdam, Germany

konstantin.dobler@hpi.de

Simon Lehnerer^{*}

Federico Scozzafava

Jonathan Janke

Mohamed Ali

Apple

{simon.lehnerer, fscozzafava,

jonathan_janke, mohamed}

@apple.com

Abstract

Reinforcement Learning with Verifiable Rewards (RLVR) has been successfully applied to significantly boost the capabilities of pre-trained large language models, especially in the math and logic problem domains. However, current research and available training datasets remain English-centric. While multilingual training data and benchmarks have been created in the past, they were not created with RLVR and current model capability in mind, and their level of difficulty is often too low to provide appropriate training signals for current models. To address this gap, we provide **mAceReason-Math**, a dataset of high-quality translations of challenging math problems sourced from a corpus specifically curated for RLVR (AceReason-Math). We further take specific care to clean and improve our translations, resulting in a coverage of 14 languages with more than 10,000 samples per language. We release the dataset to facilitate multilingual RLVR research and benchmarking in the research community.¹

1 Introduction

Reinforcement Learning with Verifiable Rewards (RLVR, Lambert et al., 2025) has unlocked the latest jump in capabilities of Large Language Models (LLMs), with a particular focus of recent research on the math and logical reasoning domains. In particular, Group Relative Policy Optimization (GRPO, Shao et al., 2024) coupled with RLVR has been widely applied. For the success of these Reinforcement Learning-based training methods, hitting a “Goldilocks zone” of difficulty that matches the current policy’s capabilities is critical (An et al., 2025). For current state-of-the-art models, which substantially exceed the capabilities of previous generations, this has led to the curation and generation

of challenging problem corpora such as AceReason-Math (Chen et al., 2025), DAPO-Math-17k (Yu et al., 2025) or POLARIS-53k (An et al., 2025).

However, these valuable resources remain limited to English, posing a significant barrier to research on GRPO and RLVR in multilingual settings. In this work, we aim to address this gap by providing over 140k high-quality translations of difficult mathematical problems from AceReason-Math (Chen et al., 2025), covering 14 languages with 10k+ samples per language, 7,572 of which are parallel across all languages. As human translation at this magnitude carries prohibitive cost, we employ a hybrid approach using both LLM-based translation and human native-speaker verification in 10 of our translation languages. This approach allows us to scale our pipeline beyond small-scale datasets. We release the collected data to benefit the research community and facilitate multilingual research in the important new RLVR paradigm.

2 Related Work

RLVR (Lambert et al., 2025) on mathematical reasoning problems has been widely used in the recent literature to study GRPO (Shao et al., 2024). As an appropriate difficulty level of the problems to match model capabilities is crucial, researchers have quickly shifted from easier datasets, such as GSM8K (Cobbe et al., 2021), to datasets with more challenging problems and more available samples (Hendrycks et al., 2021; Chen et al., 2025; Yu et al., 2025; An et al., 2025). This research direction has yielded numerous improvements over the originally proposed GRPO formulation (e.g., Liu et al., 2025; Yu et al., 2025; An et al., 2025; Luo et al., 2025a). However, these studies have been almost exclusively *English-centric*. Multilingual datasets on the difficulty level of GSM8K exist (Shi et al., 2022; Chen et al., 2023) but efforts to provide more challenging multilingual problem sets have only recently been

^{*}Equal contribution.

[†]Work done during an internship at Apple.

¹The data is available at <https://github.com/apple-aiml-research/ml-macereason-math>.

English (Original)	Translated (Language)
How many integers are there between $(11.1)^3$ and $(11.2)^3$?	German: Wie viele ganze Zahlen gibt es zwischen $(11,1)^3$ und $(11,2)^3$?
Given the function $f(x) = x^{2-m}$ is defined on the interval $[-3-m, m^2-m]$ and is an odd function, then $f(m) = ?$	Italian: Data la funzione $f(x) = x^{2-m}$ definita nell'intervallo $[-3-m, m^2-m]$ che è una funzione dispari, allora $f(m) = ?$
A circle with radius R is inscribed in an isosceles trapezoid. The upper base of the trapezoid is half of its height. Find the area of the trapezoid.	Japanese: 半径 R の円が二等辺台形に内接している。この台形の上底は高さの半分である。台形の面積を求めよ。
Given that an office at a school needs to arrange a duty roster from the 1st to the 6th day with six designated individuals participating, find the total number of different arrangements possible, given that person A and person B cannot be adjacent, and person C and person D also cannot be adjacent.	Chinese: 学校某办公室需要安排 6 个指定人员在第 1 天到第 6 天进行值班, 已知甲和乙不能相邻, 丙和丁也不能相邻, 求共有多少种不同的安排方案。
Given a function $f(x) = \log_a(\sqrt{x^2+1}+x) + \frac{1}{a^{x-1}} + \frac{3}{2}$, where $a > 0$ and $a \neq 1$. If $f(\log_3 b) = 5$ for $b > 0$ and $b \neq 1$, find the value of $f(\log_{\frac{1}{3}} b)$.	Russian: Дана функция $f(x) = \log_a(\sqrt{x^2+1}+x) + \frac{1}{a^{x-1}} + \frac{3}{2}$, где $a > 0$ и $a \neq 1$. Если $f(\log_3 b) = 5$ для $b > 0$ и $b \neq 1$, найди значение $f(\log_{\frac{1}{3}} b)$.
If 105 customers purchased a total of 300 cans of soda, with every customer purchasing at least one can of soda, determine the maximum possible median number of cans of soda bought per customer that day.	Brazilian Portuguese: Se 105 clientes compraram um total de 300 latas de refrigerante, sendo que cada cliente comprou pelo menos uma lata de refrigerante, determine a mediana máxima possível do número de latas de refrigerante compradas por cliente naquele dia.

Table 1: Examples of original samples and their translations to selected target languages from our final dataset.

made (Wang et al., 2025; Luo et al., 2025b). Still, these more challenging datasets remain limited in size with fewer than 600 samples per language. This leaves them as valuable resources for multilingual model evaluation but does not enable multilingual GRPO training studies. This gap motivates our work in this paper to create a larger multilingual dataset of challenging mathematical problems.

3 Dataset Creation

In this paper, we release a dataset with over 140k high-quality translations of challenging math problems sourced from existing English datasets used for GRPO training (Chen et al., 2025). In the following, we describe the dataset creation process. We show samples from the final dataset in Table 1.

3.1 Cleaning the base data

We manually inspect the original samples from AceReason-Math and observe several patterns of corrupted or unclean data. We distinguish these into salvageable/surface-level issues (we fix and translate these) and critical issues (which exclude samples from further processing). We describe our processing in the following and provide illustrative samples in Table 5.

Critical issues (~4%). We employ a hybrid strategy for filtering out samples, combining programmatic and LLM-based filtering. Programmatically,

we use custom-written regular expressions to filter out any samples which contain links to images or image placeholders like `[ IMAGE ]`. In general, we remove all samples which contain an URL to avoid cases where a question can only be solved with context provided in an external resource. We also search for samples containing further keywords associated with referring to external content – such as “Figure”, “diagram” – and filter out any that are problematic in a manual review. We further remove any samples that contain `\boxed{[...]}`, as this was often part of an instruction for the answer format, which might be different from the setup a user of the dataset might want to employ.²

For other – less structured – failure patterns, we collect examples and inject them into our data cleaning prompt (see Appendix A.1). These include: (1) stating the solution in the actual question, rendering the sample useless for RLVR training and benchmarking.³ (2) Samples that are only solvable with access to referenced supplementary material (e.g., “Given f as defined previously...”), which is not present. (3) Questions phrased as “How can you ...” when the actual solution given is just a single number (e.g., “[...] How can we choose f such that y is minimal?” when the solution given is just the

²For example, “<answer>[...]</answer>”.

³These solutions could be stripped to keep using the samples but we exclude these for simplicity, as we are not using the entire base dataset for translation in any case.

minimal amount). (4) Some samples are in a language other than English (Croatian or Romanian).

We exclude all samples which are classified as belonging to any of the above categories of critical issues, which amounts to roughly 4% of our considered original data.

Salvageable issues (~11%). Salvageable issues are more common than critical issues: (1) by far the largest share is made up by “task annotations” such as `Task 5.4: or [21st Math Olympiad]`. These might not be very harmful during RLVR training but we choose to remove these on principle to yield a cleaner dataset. For uniquely identifiable patterns such as `Task X.X (X points)`, we write regular expressions to remove these. However, we do not catch all occurrences, so we also add these to our LLM-based cleaning pipeline. (2) Extraction, $\text{\LaTeX}$ and formatting issues that are easily correctable based on the context (*e.g.*, `\$x^2 + 5 = ?\$` when it is clear from context that `\$x^2 + 5 = ?\$` is correct). (3) Some samples appear to also contain LLM instructions that were used during dataset construction, *e.g.*, “Translate the above text into English, please retain the original text’s line breaks and format, and output the translation result directly”. In total, roughly 11% of the original samples we treat have salvageable issues.

We collect examples for commonly occurring patterns and add these to a “data-cleaning” prompt. Before we translate a sample, we first prompt Claude Sonnet 4 (Anthropic, 2025b) to adapt the English version if such salvageable issues are detected.

Additionally, some samples refer to diagrams provided in the actual question as `[asy]` code. Since solving these tasks is possible but heavily relies on the specialized skill of parsing these `[asy]` instructions, we still translate these samples but provide them in a separate split.

3.2 Translation

After we have cleaned and filtered the source data, we now enter the translation phase. We provide our prompts in Appendix A.

Initial human translation quality review. We first create initial translations of a random sample of 100 English problems using Claude Sonnet 4 and conduct a human review with native-speaker annotators in German, French, Italian, Spanish, Chinese, Korean, Japanese, Thai, Russian, and Brazilian Por-

tuguese⁴. In this round, 97% of translations were rated as “acceptable” (understandable but phrasing could be improved) or “excellent”. Of the translations that were rated as problematic, a large portion is due to mismatches or changes of $\text{\LaTeX}$ content compared to the original English version. We analyze the suggested improvements and issues detected by the native-speaker annotators and improve our prompts for the full translation run.

Full translation run. For the full translation run, we employ an iterative LLM-based refinement process (also using Claude Sonnet 4). We first create an initial translation as in our initial run (but using the improved prompts). In particular, we prompt to respect number formatting conventions (*e.g.*, US English 12,345.67 vs. German 12.345,67). Then, the translations are first graded using pre-defined grading criteria similar to our human annotator guidelines and subsequently improved using a refinement prompt, in case issues are detected. We observed that this approach leads to an improved translation quality and avoids rare but obvious errors in the initial translations. We run this improvement loop up to five times and discard translations which are not accepted in the LLM-based grading round by then. The majority of samples do not need to enter this improvement loop and are accepted immediately. We report improvement loop statistics in Appendix B.

3.3 Compiling the final dataset

Parallel train split. For research into multilingual model training and evaluation, parallel data across all languages ensures that results are comparable. Therefore, we construct a parallel split which contains only samples for which we have translations that have been accepted in our translation stage for all of our target languages. Our final parallel train split contains 7,572 samples per language.

Full train split. We additionally compile full train splits which contain all available translations for each language, although these splits are no longer parallel across all languages. The full train splits per language contain 10,137–12,172 samples, varying per language.

⁴Native-speaker validation excludes Swahili, Telugu and Bengali. We still provide translations into these languages to increase language coverage.

Model	en	zh	de	es	fr	it	pt	ru	ja	ko	th	bn	te	sw	Average
gpt-oss-20b	79.3	79.5	78.0	75.5	79.0	76.0	76.7	80.0	78.9	76.1	78.5	74.4	78.1	65.3	76.8 ± 3.6
SmolLM3-3B	77.8	53.4	69.6	72.7	72.0	69.7	71.6	69.0	58.1	54.7	59.1	27.8	16.4	20.0	56.6 ± 19.8
gemma-3-12b-it	58.0	50.3	47.6	53.4	53.5	53.2	52.2	51.8	47.2	47.3	52.1	41.7	42.0	41.6	49.4 ± 4.9
gemma-3-4b-it	28.3	34.5	29.8	45.1	41.6	42.2	41.0	38.6	23.2	32.2	34.1	23.6	24.0	22.1	32.9 ± 7.6
gemma-3-1b-it	19.7	9.3	9.6	11.6	10.3	11.1	10.8	9.9	8.0	6.1	6.7	7.2	4.7	3.1	9.1 ± 3.8
DeepSeek-R1-Distill-Qwen-7B	83.5	80.7	65.7	69.2	69.9	69.5	67.6	68.8	63.0	65.1	59.7	67.4	47.4	24.4	64.4 ± 13.8
DeepSeek-R1-Distill-Qwen-1.5B	64.9	57.2	32.3	41.4	46.1	40.3	43.8	44.9	49.7	30.5	29.4	33.2	20.5	16.7	39.3 ± 12.9

Table 2: We report the average accuracy over eight sampled responses per prompt (avg@8) of different open-weight models across languages on the test split of our dataset. The Average column reports mean $\pm$ std across languages.

Train ↓	Eval: training languages				Eval: other languages											Average
	en	de	ko	sw	es	fr	it	pt	zh	ja	ru	bn	te	th		
en	30.9 (+6.8)	23.7 (+15.8)	22.8 (+14.2)	9.4 (+4.3)	25.1 (+15.8)	25.2 (+16.4)	26.7 (+16.7)	23.6 (+12.9)	18.0 (+8.5)	18.4 (+10.1)	25.6 (+17.8)	17.8 (+12.2)	12.2 (+6.4)	19.5 (+13.5)	21.4 (+12.2)	
de	29.3 (+5.3)	24.8 (+16.9)	22.8 (+14.2)	9.7 (+4.7)	28.8 (+19.5)	27.6 (+18.9)	26.4 (+16.4)	27.4 (+16.6)	26.2 (+16.8)	25.2 (+17.0)	26.9 (+19.1)	19.6 (+13.9)	14.3 (+8.5)	24.1 (+18.0)	23.8 (+14.7)	
ko	30.1 (+6.1)	27.2 (+19.3)	24.2 (+15.6)	9.9 (+4.9)	29.5 (+20.2)	26.8 (+18.0)	27.6 (+17.6)	27.2 (+16.4)	28.4 (+19.0)	27.3 (+19.1)	27.1 (+19.3)	20.3 (+14.7)	15.6 (+9.8)	24.3 (+18.3)	24.7 (+15.6)	
sw	28.0 (+4.0)	23.7 (+15.8)	20.3 (+11.7)	12.0 (+6.9)	25.7 (+16.4)	26.4 (+17.6)	23.9 (+13.9)	26.3 (+15.5)	18.3 (+8.9)	23.7 (+15.5)	26.1 (+18.3)	18.6 (+13.0)	14.7 (+8.9)	22.0 (+15.9)	22.1 (+13.0)	

Table 3: Performance of Qwen2.5-1.5B-Instruct on our test split after GRPO training on the parallel train split of our dataset in different languages. We report the average accuracy over eight sampled responses per prompt (avg@8) and the improvement over the untrained checkpoint’s avg@8 score in parentheses.

Human-validated test set. We use a random sample of 183 items⁵ which have translations in all our target languages as our test set (these samples are removed from all other splits). We again conduct a human review of these samples with native-speaker annotators for the same subset of languages. This time, we render $\text{\LaTeX}$ content on our annotation platform using MathJax⁶ to directly display mathematical content instead of $\text{\LaTeX}$ code, which our annotators were not familiar with.

As in our initial rounds, samples marked as “problematic” and suggestions provided by human native-speaker annotators in case of “acceptable” ratings can be noisy. Issues often stem from lack of mathematical understanding or domain knowledge.⁷ Therefore, we do not directly apply the suggestions but instead filter them selectively with Claude Sonnet 4. We additionally manually check all translations marked as “problematic”.

We are left with roughly 0.1% of individual translations rated as “problematic” (two translations in total, to French and Brazilian Portuguese, respectively). Additionally, we apply post-editing suggestions by our annotators which were only marked as improving naturalness or stylistic choices rather than critical in 36% of samples.

⁵We started with 200 samples but filtered out samples in an additional post-hoc cleaning round based on the steps described in Section 3.1.

⁶<https://www.mathjax.org>

⁷For example, the following suggested “improvement” in German: “sequence” should be translated as “Sequenz” instead of “Folge”. Here, “Sequenz” would be the literal German translation but “Folge” is the appropriate German mathematical term.

4 Evaluation

We evaluate open-weight models from several families – Gemma 3 (Gemma Team et al., 2025), gpt-oss (OpenAI et al., 2025), SmolLM3 (Bakouch et al., 2025), and the distilled DeepSeek-R1 models (DeepSeek-AI et al., 2025) – on the human-validated parallel test set in Table 2. Overall, we find consistent trends of larger models exhibiting better performance (*i.e.*, within model families). Additionally, we observe substantial differences in performance across languages, motivating the creation of our dataset. We provide further evaluations and details in Appendix E.

We further run GRPO training of Qwen2.5-1.5B-Instruct (Qwen et al., 2025) in four selected languages (English, German, Korean, Swahili) on our parallel train split⁸ and report the results in Table 3. The training details are reported in Appendix D. Interestingly, while training in English yields the best performance when testing in English, the best performance in other test languages is often achieved by training in German or Korean. All training languages yield substantial crosslingual transfer across tested languages, even when training in the lesser-resourced language Swahili.

Taken together, our results show substantial language-dependent performance differences and asymmetric crosslingual transfer in mathematical reasoning in our small-scale experiments. We hope that the release of mAceReason-Math will enable

⁸Training used an earlier version of this split that included some samples which we removed in the release, in particular reworded copies of 23 test problems. Results on the other test items are consistent with those on the full test set.

further studies of these phenomena.

At the time of writing, frontier systems solve olympiad problems at gold-medal standard (Google DeepMind, 2025) and have contributed to resolving long-standing open problems such as the Erdős unit distance conjecture (Alon et al., 2026). mAceReason-Math is not designed to challenge such systems. We instead see its main value in multilingual RLVR and GRPO research with the smaller open models commonly used in academic work (Liu et al., 2025; Luo et al., 2025a; Zeng et al., 2025; Shao et al., 2026).

5 Conclusion

We release **mAceReason-Math**, a multilingual version of a filtered subset of AceReason-Math, covering 14 languages in total and comprising 10k+ translated problems per language in 13 languages, cleaned versions of the original English problems, a fully parallel train split of 7,572 samples per language, and a test set of 183 samples with native-speaker validation of the translations in 10 languages. Our pipeline combines targeted cleaning with iterative LLM translation and native-speaker checks to produce faithful, high-quality translated problems. While mAceReason-Math is not intended to challenge frontier models, it provides a multilingual resource for RLVR research in academic work.

Limitations

mAceReason-Math relies on an LLM-based translation and refinement pipeline, so residual translation or formatting errors may persist despite extensive quality control. mAceReason-Math is further based on English data from AceReason-Math (Chen et al., 2025). We apply cleaning and filtering as described in Section 3.1 but some wrong or underspecified problems might remain. We validate the pipeline with native-speaker annotators, but this excludes Swahili, Telugu, and Bengali due to a lack of available annotators. Translation quality may therefore be more variable for these languages. Moreover, only the test set received full native-speaker review of the final translations (again excluding Swahili, Telugu, and Bengali). Our evaluation also depends on automated answer verification tools⁹, which often handle non-US number formatting (e.g., commas as decimal separators) but may be less reliable

for complex expressions, potentially leading to occasional false negatives.

Acknowledgments

We thank Adam Golinski and Michael Kirchhof for their insightful discussion and valuable feedback. We also thank our team of professional translators and native speaker validators who ensured high-quality translations in their respective languages.

References

- Noga Alon, Thomas F. Bloom, W. T. Gowers, Daniel Litt, Will Sawin, Arul Shankar, Jacob Tsimmerman, Victor Wang, and Melanie Matchett Wood. 2026. *Remarks on the disproof of the unit distance conjecture. Preprint*, arXiv:2605.20695.
- Chenxin An, Zhihui Xie, Xiaonan Li, Lei Li, Jun Zhang, Shansan Gong, Ming Zhong, Jingjing Xu, Xipeng Qiu, Mingxuan Wang, and Lingpeng Kong. 2025. *Polaris: A post-training recipe for scaling reinforcement learning on advanced reasoning models*.
- Anthropic. 2024. *Model card addendum: Claude 3.5 Haiku and upgraded Claude 3.5 Sonnet*.
- Anthropic. 2025a. *System card: Claude Haiku 4.5*.
- Anthropic. 2025b. *System card: Claude Opus 4 & Claude Sonnet 4*.
- Anthropic. 2025c. *System card: Claude Sonnet 4.5*.
- Elie Bakouch, Loubna Ben Allal, Anton Lozhkov, Noumane Tazi, Lewis Tunstall, Carlos Miguel Patiño, Edward Beeching, Aymeric Roucher, Aksel Joonas Reedi, Quentin Gallouédec, Kashif Rasul, Nathan Habib, Clémentine Fourrier, Hynek Kydlicek, Guilherme Penedo, Hugo Larcher, Mathieu Morlon, Vaibhav Srivastav, Joshua Lochner, and 4 others. 2025. *SmolLM3: smol, multilingual, long-context reasoner*. <https://huggingface.co/blog/smollm3>.
- Nuo Chen, Zinan Zheng, Ning Wu, Ming Gong, Dongmei Zhang, and Jia Li. 2023. *Breaking language barriers in multilingual mathematical reasoning: Insights and observations. Preprint*, arXiv:2310.20246.
- Yang Chen, Zhuolin Yang, Zihan Liu, Chankyu Lee, Peng Xu, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2025. *Acereason-nemotron: Advancing math and code reasoning through reinforcement learning. Preprint*, arXiv:2505.16400.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. *Training verifiers to solve math word problems. Preprint*, arXiv:2110.14168.

⁹Such as <https://github.com/huggingface/Math-Verify>.

- Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Naveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 3416 others. 2025. [Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities](#). *Preprint*, arXiv:2507.06261.
- DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. [DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning](#). *Preprint*, arXiv:2501.12948.
- Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. [Gemma 3 technical report](#). *Preprint*, arXiv:2503.19786.
- Google DeepMind. 2025. [Advanced version of Gemini with Deep Think officially achieves gold-medal standard at the International Mathematical Olympiad](#). Blog post, July 21, 2025. Accessed September 2026.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *NeurIPS*.
- Diederik P. Kingma and Jimmy Ba. 2017. [Adam: A method for stochastic optimization](#). *Preprint*, arXiv:1412.6980.
- Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, and 4 others. 2025. [Tulu 3: Pushing frontiers in open language model post-training](#). *Preprint*, arXiv:2411.15124.
- Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025. Understanding r1-zero-like training: A critical perspective. In *Conference on Language Modeling (COLM)*.
- Ilya Loshchilov and Frank Hutter. 2019. [Decoupled weight decay regularization](#). *Preprint*, arXiv:1711.05101.
- Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y. Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Li Erran Li, Raluca Ada Popa, and Ion Stoica. 2025a. [Deepscaler: Surpassing o1-preview with a 1.5b model by scaling rl](#). Notion Blog.
- Wenyang Luo, Wayne Xin Zhao, Jing Sha, Shijin Wang, and Ji-Rong Wen. 2025b. [Mmath: A multilingual benchmark for mathematical reasoning](#). *Preprint*, arXiv:2505.19126.
- OpenAI, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, Che Chang, and 107 others. 2025. [gpt-oss-120b & gpt-oss-20b model card](#). *Preprint*, arXiv:2508.10925.
- Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, and 24 others. 2025. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Rulin Shao, Shuyue Stella Li, Rui Xin, Scott Geng, Yiping Wang, Sewoong Oh, Simon Shaolei Du, Nathan Lambert, Sewon Min, Ranjay Krishna, Yulia Tsvetkov, Hannaneh Hajishirzi, Pang Wei Koh, and Luke Zettlemoyer. 2026. [Spurious rewards: Rethinking training signals in RLVR](#). In *Forty-third International Conference on Machine Learning*.
- 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.
- 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. 2022. [Language models are multilingual chain-of-thought reasoners](#). *Preprint*, arXiv:2210.03057.
- Yiming Wang, Pei Zhang, Jialong Tang, Haoran Wei, Baosong Yang, Rui Wang, Chenshu Sun, Feitong Sun, Jiran Zhang, Junxuan Wu, Qiqian Cang, Yichang Zhang, Fei Huang, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. [Polymath: Evaluating mathematical reasoning in multilingual contexts](#). *Preprint*, arXiv:2504.18428.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. [Qwen3 technical report](#). *Preprint*, arXiv:2505.09388.
- Qiyang Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gao-hong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi

Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, and 16 others. 2025. [Dapo: An open-source llm reinforcement learning system at scale](#). *Preprint*, arXiv:2503.14476.

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. 2025. [Simplerl-zoo: Investigating and taming zero reinforcement learning for open base models in the wild](#). In *Conference on Language Modeling (COLM)*.

A Translation Pipeline Prompts

We report the prompts used in our LLM-based translation pipeline in the following. We report the prompt used for filtering and cleaning the English base data in [Appendix A.1](#). In [Appendix A.2](#), we report the prompt used for initial translations into the target languages. [Appendix A.3](#) contains the prompt used for LLM-based assessment of translations and the generation of improvement suggestions. Finally, [Appendix A.4](#) contains the prompt used for improving translations based on these detected issues and improvement suggestions.

A.1 Filtering

Mathematical Content Quality Assessment & Cleanup

You are analyzing mathematical problem-solution pairs for
 → quality issues and artifacts. This is critical quality
 → control before translation.

Content to Analyze:
 <problem>{problem}</problem>
 <solution>{answer}</solution>

Your Task:
 1. ****Assess if severely corrupted**** (unsuitable for
 → translation)
 2. ****Clean up artifacts**** (if content is salvageable)
 3. ****Assign appropriate tags**** based on issues found

SEVERE CORRUPTION CRITERIA (Filter Out):

Content Integrity Issues:
 - ****corruption_solution_given****: Answer provided in the
 → question itself. Examples:
 {self._format_examples_section("solution_in_question", 5)}
 - ****corruption_open_ended****: Questions asking "how can
 → you..." instead of specific answers. Examples:
 {self._format_examples_section("open_ended", 5)}
 - ****corruption_non_english****: Problems not in English.
 → Examples:
 {self._format_examples_section("non_english", 1)}
 - ****corruption_missing_context****: References undefined
 → context or nonsensical Q&A pairs. Examples:
 {self._format_examples_section("nonsensical_qa", 1)}
 - ****corruption_unintelligible****: Completely incoherent
 → problem statements
 - ****corruption_domain_specific****: Requires ultra-specific
 → non-mathematical knowledge. Examples:
 - Tax problems requiring specific German tax brackets for
 → 2023
 - Problems requiring knowledge of specific company stock
 → prices on exact dates
 - ****corruption_multiple_unrelated****: Multiple unrelated
 → problems or repeated questions
 - ****corruption_other****: Severe corruption that doesn't fit
 → any of the above categories. ****Use sparingly**** and only
 → when content is clearly severely corrupted but doesn't
 → match existing patterns. Must provide detailed
 → corruption_reasons explaining the specific issue.

ARTIFACT CLEANUP (Fixable Issues):

****CRITICAL SAFETY RULE****: Only remove/fix artifacts that do
 → NOT change the expected solution or important
 → mathematical context. Never alter mathematical meaning.

Question Artifacts to Clean:

- ****cleanup_question_cleaned****: Remove these artifacts from
 → problems:
 - Author attributions: ****Created by [Name]****, "By:
 → [Name]", etc.
 - Creation timestamps or metadata
 - Platform-specific tags or markers
 - Problem source artifacts like "International Math
 → Olympiad: [Problem]"
 - Graded question formatting like "[Problem] (6 points)"
 → or "## Task 6.2" or "Example 7"
 - Leading task artifacts such as "I-22" in "I-22."
 → Calculate 2+3." or "3-2." in "3.2. Is 234 a prime
 → number?"

- ****cleanup_question_fixed****: Fix malformed "question"
 → statements by rephrasing into proper questions. ****Use
 → sparingly and only when it significantly improves the
 → problem quality.**** Examples:
 - Incomplete statements: "10. The positive integer pairs
 → $(a, b) = \frac{a}{b^2}$, where $a \neq b$, that make
 → $\frac{a}{b^2} = a + b$ a prime number." → "Find
 → the positive integer pairs (a, b) where $a \neq b$,
 → that make $\frac{a}{b^2} = a + b$ a prime
 → number."
 - Statement format: "The value of $\sqrt{16}$." → "What
 → is the value of $\sqrt{16}$?"
 - Rephrasing unclear statements: "Calculate the area of
 → triangle ABC with vertices A(0,0), B(3,0), C(0,4)
 → using the coordinate formula." → "What is the area of
 → triangle ABC with vertices A(0,0), B(3,0), C(0,4)?"

****IMPORTANT****: Only apply cleanup_question_fixed when the
 → original statement is genuinely unclear or confusing. Do
 → not rephrase statements that are already mathematically
 → clear, even if they lack traditional question
 → formatting. When in doubt, leave the problem unchanged.

Solution Artifacts to Clean:

- ****cleanup_solution_cleaned****: Remove these from solutions:
 - Unnecessary LaTeX spacing commands: "\," (small space),
 → "\;" (medium space), "\!" (negative space)
 - Stray formatting characters that don't contribute to
 → mathematical meaning
 - Metadata or attribution text in solutions
 - ****cleanup_solution_fixed****: Fix obvious extraction errors
 → in solutions (only when correction is completely obvious
 → from context). Examples:
 - Incomplete expressions: " $x^2 =$ " missing variable → " $x^2 =$ "
 - Missing variables: " $(^2 + b^2 + c^2)$ " → " $(a^2 + b^2 + c^2)$ "
 - Obviously broken notation

****WARNING****: Never "fix" solutions in complex ways requiring
 → actual mathematical reasoning or computation.

Response Format:

****If severely corrupted****
 ```json  
 {{  
 "severity\_assessment": {{  
 "is\_severely\_corrupted": true,  
 "corruption\_reasons": ["Specific reasons"]  
 }},  
 "tags": ["corruption\_solution\_given",  
 → "corruption\_open\_ended"],  
 "artifacts\_found": false  
 }}  
 ```

****If clean (no issues)****
 ```json  
 {{  
 "severity\_assessment": {{  
 "is\_severely\_corrupted": false,  
 "corruption\_reasons": []  
 }},  
 "tags": [],  
 "artifacts\_found": false  
 }}  
 ```

****If artifacts cleaned****
 ```json  
 {{  
 "artifacts\_found": true,  
 "changes\_made": ["Removed problem numbering 'I-22'",  
 → "Fixed incomplete statement to proper question"],  
 "cleaned\_problem": "Problem after cleanup",  
 "cleaned\_solution": "Solution after cleanup",  
 "severity\_assessment": {{  
 "is\_severely\_corrupted": false,  
 "corruption\_reasons": []  
 }},  
 }}  
 ```

```

    "tags": ["cleanup_question_cleaned",
    ↪      "cleanup_question_fixed"]
  }}
  ...

```

****Conservative approach**:** Only rephrase question / solution
 ↪ when >90% confident.

****Think step by step before giving your final output.**
 ↪ **Examine every aspect of the content for mathematical**
 ↪ **coherence, logical consistency, and fundamental**
 ↪ **integrity before making your assessment.****

Your final output should contain ONLY a valid JSON object
 ↪ within <answer> and </answer> tags

A.2 Translation

Mathematical Problem Translation

You are translating mathematical and logical reasoning
 ↪ problems from English to {language_name}.

Problem to Translate:
 <problem>{task.original_problem}</problem>
 <solution>{task.original_answer}</solution>

Critical Translation Guidelines:

Mathematical Accuracy (CRITICAL):

1. ****Preserve all mathematical concepts**:** All variables, numbers, equations, and logical relationships must remain identical
2. ****Maintain problem structure**:** The mathematical logic and problem logic must be unchanged
3. ****Verify mathematical consistency**:** Ensure translated problem leads to the same solution method

LaTeX Handling (CRITICAL):

1. ****NEVER translate LaTeX content**:** Preserve all content within '\$...\$' or '\\(...\\)' exactly as written
2. ****Preserve exact spacing**:** Match LaTeX spacing and notation style precisely (e.g., do not change '\$\\triangle A B C\$' to '\$\\triangle ABC\$')
3. ****Within LaTeX, only translate \\text{...} content**:** Within LaTeX, only translate natural language inside '\\text{...}'
4. ****Keep LaTeX environments**:** Maintain '\$...\$', '\\(...\\)', '\\begin{equation}...\\end{equation}' notation exactly and do not switch in the translation
5. ****DO NOT TRANSLATE mathematical or other language-agnostic content**:**
 - Pure numbers, mathematical expressions, formulas, symbols or notation: "42", "-7", "0", "x²", "sin(n/4)", "y = mx + b", ...
 - Keep LaTeX or mathematical markup such as:
 - ↪ "\\frac{...}{...}", "\\angle{...}", "\\overline{...}", "\\triangle{...}", ...

Natural Language Quality (CRITICAL):

1. ****Create natural flow**:** Use natural {language_name} sentence structure, not word-for-word translation. A natural translation that restructures sentences flow or word order is better than an awkward literal word-for-word translation.
2. ****Use informal register**:** Use informal forms (e.g. German: "Du" instead of "Sie", French: "tu" instead of "vous", etc.)
3. ****Apply number formatting**:** Use {language_name} number formatting conventions (e.g. for German use 12.345,60 instead of the English: 12,345.60)
4. ****Standard mathematical terminology**:** Use terms familiar to native {language_name} speakers. For jargon or technical terms, use the terms that would be commonly used in {language_name} higher education contexts.
5. ****Cultural Context**:** If it does not influence the logic of the problem, localize English-centric references to a {language_name} cultural context.

Answer Format Requirements:

1. ****Always use Arabic numerals**:** Answers must use 0-9 digits regardless of target language
2. ****Apply number formatting**:** Use {language_name} number formatting conventions (e.g. for German use 12.345,60 instead of the English: 12,345.60)
3. ****Translate natural language**:** Convert "True"/"False"/"Increasing" etc. to {language_name}
4. ****Keep mathematical notation unchanged**:** Preserve LaTeX, formulas, expressions exactly
5. ****Match original LaTeX style**:** Don't add or remove LaTeX formatting unless number conventions differ

Language-Specific Guidelines for {language_name}:
 {self._get_language_specific_guidelines(task.language)}

Response Format:

Respond with a JSON object containing:
 - 'translated_problem': string (the problem translated to {language_name})

- 'translated_solution': string (translated if natural language, unchanged if mathematical)

First, think step by step. Then output your final JSON
 ↪ translation within <answer> and </answer> tags.

A.3 Grading

Mathematical Problem Translation Quality Assessment

You are assessing the quality of a mathematical problem
 ↪ translation from English to {language_name}.

Original (English):
 <problem>{task.original_problem}</problem>
 <solution>{task.original_answer}</solution>

Translation ({language_name}):
 <problem>{translated_problem}</problem>
 <solution>{translated_solution}</solution>

Assessment Criteria (Based on Annotation Guidelines):

Critical Issues (Must be perfect):

1. ****Mathematical Accuracy**:** All mathematical concepts, relationships, and logical structures preserved exactly
2. ****LaTeX Consistency**:**
 - All '\$...\$' and '\\(...\\)' content identical to original
 - Spacing within LaTeX matches exactly (e.g., '\\triangle A B C\$' spacing preserved)
 - Only '\\text{...}' content translated within LaTeX
3. ****Logical Consistency**:** Translation maintains same mathematical relationships as original. No new details should be "invented"
4. ****Answer Format**:**
 - Natural language properly translated to {language_name}
 - Mathematical notation unchanged from original
 - Arabic numerals used correctly
 - Answer matches question if cultural or other details in the question were changed to be more appropriate for {language_name}

Critical Quality Considerations:

1. ****Mathematical Terminology**:** Uses appropriate {language_name} mathematical terms commonly used in {language_name} higher education
2. ****Natural Fluency**:** Sounds natural to native {language_name} speakers (no "translationese"). A natural translation that restructures sentences flow or word order is better than an awkward literal word-for-word translation
3. ****Register Appropriateness**:** Uses informal register correctly
4. ****Number Formatting**:** Applies {language_name} conventions appropriately

Important Checks:

- ****Terminology Accuracy**:** Flag non-standard or incorrect mathematical terms
- ****LaTeX Preservation**:** Verify exact match with original LaTeX formatting
- ****Natural Flow**:** Assess if translation sounds natural in {language_name}, avoid overly literal and stilted translations
- ****Cultural Appropriateness**:** Check mathematical conventions for {language_name}

Required Response Format:

Respond with a JSON object containing:
 - 'translation_quality_is_excellent': boolean (true only if NO issues found)
 - 'concrete_issues': array of strings (specific issues with format: "Issue: [description]. Suggestion: [fix]")
 - 'mathematical_accuracy_preserved': boolean (true if all math concepts preserved)
 - 'answer_handling_correct': boolean (true if answer was handled correctly)

For terminology issues, use format: "Mathematical terminology: '[English term]' should be translated as '[correct {language_name} term]' instead of '[current translation]'"

First, think step by step. Then, as your final answer,
 ↪ output ONLY a valid JSON object within <answer> and
 ↪ </answer> tags.

A.4 Translation improvement

Mathematical Problem Translation Improvement

You are improving a mathematical problem translation from
 ↪ English to {language_name} based on specific identified
 ↪ issues.

Original (English):
 <problem>{task.original_problem}</problem>
 <solution>{task.original_answer}</solution>

Language	Train (Parallel)	Test (Parallel)	Train (All)
English	7,572	183	12,172
German	7,572	183	11,066
French	7,572	183	10,915
Spanish	7,572	183	11,278
Chinese	7,572	183	10,336
Russian	7,572	183	11,093
Japanese	7,572	183	10,249
Swahili	7,572	183	10,982
Telugu	7,572	183	10,825
Bengali	7,572	183	10,939
Thai	7,572	183	10,967
Brazilian Portuguese	7,572	183	10,565
Italian	7,572	183	10,577
Korean	7,572	183	10,137

Table 4: Number of samples in mAceReason-Math per split and language.

```

## Current Translation ({language_name}):
<problem>{current_translation.get("translated_problem",
  ↳ "")}</problem>
<solution>{current_translation.get("translated_solution",
  ↳ "")}</solution>

## Improvement Guidelines (Based on Annotation Guidelines):

### Priority 1 - Critical Fixes:
1. Mathematical Accuracy: Ensure all mathematical
  ↳ concepts preserved exactly
2. LaTeX Consistency: Fix any spacing or notation
  ↳ inconsistencies with original
3. Logical Preservation: Maintain identical mathematical
  ↳ relationships

### Priority 2 - Quality Enhancements:
1. Natural Fluency: Create natural {language_name} while
  ↳ preserving mathematical meaning
2. Mathematical Terminology: Use standard
  ↳ {language_name} mathematical terms used in higher
  ↳ education contexts
3. Register Correction: Apply informal forms correctly
4. Number Formatting: Use {language_name} conventions
  ↳ appropriately

### LaTeX Correction Rules:
- Match original spacing exactly: If English has
  ↳ '$\triangle A B C$', the translation should also use
  ↳ '$\triangle A B C$' instead of e.g. '$\triangle ABC$'
- Preserve notation style: '$\{...\}$' vs '$...$' must match
  ↳ original
- Only translate '$\text{...}$' content within LaTeX
- Never alter mathematical expressions or formulas

## Language-Specific Guidelines for {language_name}:
{self._get_language_specific_guidelines(task.language)}

## Issues to Fix:
{issues_text}

## Required Response Format:
Respond with a JSON object containing:
- 'improvement_notes': string (explanation of changes made)
- 'translated_problem': string (improved problem
  ↳ translation)
- 'translated_solution': string (improved solution -
  ↳ translated if natural language, unchanged if
  ↳ mathematical)

First, think step by step. Then output ONLY the improved
  ↳ JSON translation within <answer> and </answer> tags.

```

B Additional Statistics

We report the number of samples per split and language in mAceReason-Math in Table 4. We further report the number of accepted samples per improvement stage in our translation pipeline in Figure 1.

C Data Cleaning Examples

We provide an overview of our filtering criteria for excluding samples present in the original

AceReason-Math dataset (Chen et al., 2025) in Table 5.

D Training

D.1 Training Details

We run GRPO training with a DAPO-style loss (Yu et al., 2025) without scaling the rewards by the group standard deviation as recommended by Liu et al. (2025). We train 500 steps with a learning rate warmup for the first 3% of steps and no learning rate decay. Each step has a total batch size of 256. As we use 8 rollouts per prompt, this results in 32 unique prompts seen per step. We use a maximum learning rate of $1e-6$, maximum gradient norm of 1.0, and the AdamW optimizer (Kingma and Ba, 2017; Loshchilov and Hutter, 2019) with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1e-8$. We use a temperature of 1.0 for sampling rollouts and a maximum rollout length of 4096 tokens. We do not mask completions exceeding the maximum rollout length. We apply a Kullback-Leibler (KL) penalty with a $\beta = 0.0001$. We run training on the parallel split of mAceReason-Math and evaluation on the test split.

E Evaluation

E.1 Evaluation Details

For evaluating hosted models, we sample a single response per prompt (pass@1). We conducted the evaluations in October and November 2025.

To extract answers, we prompt the models to provide the final answer in `<answer>...</answer>` tags. This instruction is appended at the end of the prompt and given in the prompt’s language (translated from English). We first attempt to extract an answer from the `<answer>...</answer>` tags but additionally attempt to extract an answer from `\boxed{...}` if no `<answer>...</answer>` tags are present. We then evaluate the extracted answer against the original ground truth using Math-Verify. Additionally, since in some cases the answer has been adapted in the translated version, e.g., to match the German number formatting conventions, we also check for string equality of the extracted answer with the translated version’s answer. In practice, current tools already handle this correctly except in complicated cases.

E.2 Further Evaluation Results

We provide further evaluation results for the Qwen3 model family (Yang et al., 2025) as well as closed

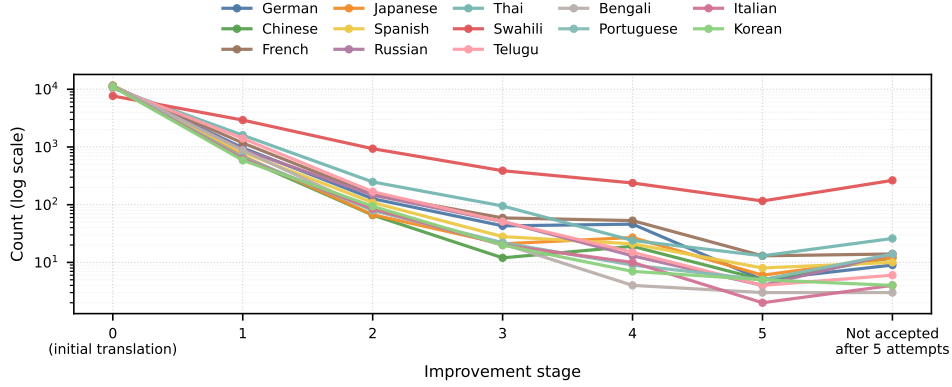


Figure 1: Number of accepted samples per improvement stage in the translation pipeline. Most samples are accepted after the initial translation or one additional round of LLM-based refinement.

Primary Category	Share	Subcategory	Illustrative Examples
Missing Context	~2%	Missing Visual	“The diagram shows three rectangles and three straight lines. What is the value of $p + q + r$ [...]” (No diagram provided)
		Missing Definitions	“Define a regular n -pointed star as described in the original problem [...]” (Critical definitions not provided)
Content and Extraction Issues	~1.5%	Answer Leakage	“[...] Answer: decreased by 6%” (Solution provided in problem)
		Processing Artifacts	“[...] text above has been translated as requested [...]” (Leftover data-processing artifact replaces or corrupts problem statement)
		Truncated Content	“[...] We know that for any pair of distinct numbers from this set there is” with answer: “50” (Cuts off before finishing question, answer is just “50”)
Ill-posed Question	~0.5%	Free-text answer	“ How can you cut a 5×5 square [...] into 50 equal squares?” (No methodology provided, answer is just a single integer)
		Mathematical Errors	“ $x + \frac{1}{x_i}$ has a maximum ” (Impossible claim)
Other Issues	<0.01%	Various	“Odredi sve prirodne brojeve n za koje postoje [...]” (e.g., non-English problems, ultra-specific domain knowledge, ...)

Table 5: Overview of “critical issues” identified in the original data. Examples abbreviated with [...] for space constraints. **Bold text** highlights key corruption indicators within the quoted examples.

Model	en	zh	de	es	fr	it	pt	ru	ja	ko	th	bn	te	sw	Average
Open-weight models (avg@8 evaluation)															
Qwen3-14B-Base	30.5	28.9	24.2	26.5	24.5	20.8	24.8	27.6	20.6	19.5	18.0	12.3	11.0	5.4	21.0 $\pm$ 7.0
Qwen3-8B-Base	25.3	22.8	17.5	21.4	20.3	18.3	22.7	24.1	14.6	14.1	12.0	8.0	12.2	2.3	16.8 $\pm$ 6.4
Qwen3-4B-Base	14.8	15.1	11.5	13.3	13.0	12.8	14.1	12.8	9.7	7.9	7.3	7.3	7.8	2.7	10.7 $\pm$ 3.5
Qwen3-1.7B-Base	7.4	4.7	2.8	3.6	3.6	2.9	3.8	3.6	2.2	1.9	2.4	0.6	0.8	0.4	2.9 $\pm$ 1.8
Qwen3-14B	89.1	90.5	91.8	91.7	91.3	92.2	91.6	87.6	89.6	89.5	90.7	91.0	87.9	69.5	88.9 $\pm$ 5.5
Qwen3-8B	85.9	85.3	86.8	88.3	87.7	88.0	86.2	85.7	86.6	85.5	85.5	86.4	83.0	57.6	84.2 $\pm$ 7.5
Qwen3-4B	85.6	85.9	89.1	87.5	87.7	87.4	88.2	83.0	84.0	84.2	85.0	84.5	78.3	43.8	82.4 $\pm$ 11.0
Qwen3-1.7B	73.2	72.6	73.6	74.8	76.2	75.9	74.3	71.4	72.2	70.1	69.5	65.1	57.6	25.7	68.0 $\pm$ 12.7
Closed models (pass@1 evaluation) [†]															
Gemini 2.5 Flash	78.9	85.8	81.6	79.5	82.1	81.1	81.1	82.1	81.6	84.2	80.5	80.5	83.7	76.3	81.4 $\pm$ 2.2
Gemini 2.5 Flash-Lite	77.9	68.9	67.9	71.6	70.0	72.1	71.6	73.7	64.2	64.7	68.4	52.1	62.1	62.6	67.7 $\pm$ 6.1
Claude Sonnet 4.5	76.8	78.9	75.8	78.9	76.8	76.3	73.2	80.0	74.2	77.9	76.3	71.1	71.1	72.6	75.7 $\pm$ 2.8
Claude Haiku 4.5	78.4	71.6	72.1	68.9	68.4	71.1	69.5	71.1	70.0	68.9	71.6	69.5	70.0	65.3	70.5 $\pm$ 2.8
Claude Sonnet 3.5	52.6	50.0	48.4	48.9	50.0	51.6	51.1	54.7	47.4	51.1	48.9	38.4	40.5	36.3	47.9 $\pm$ 5.3
Claude Haiku 3.5	40.0	27.4	33.2	31.1	33.2	38.9	33.2	39.5	30.5	28.9	30.5	23.2	17.4	20.5	30.5 $\pm$ 6.6

Table 6: Accuracy (%) of different models across languages on our test split. The best result for each language within the open- and closed-model categories is marked in **bold**. The Average column reports mean $\pm$ std across languages. We report the pass@1 score (classic accuracy) for closed models and the average accuracy over eight sampled responses per prompt (avg@8) for open-weight models. [†]Closed-model scores include seven test samples later removed from the release.

models from the Gemini 2.5 (Comanici et al., 2025) and Claude (Anthropic, 2024, 2025a,c) families in Table 6. The Qwen3 reasoning models (“non-Base”) in particular achieve strong performance across many languages, even outperforming closed models such as Claude Sonnet 4.5.