MGSM-Pro: A Simple Strategy for Robust Multilingual Mathematical Reasoning Evaluation

Anonymous ACL submission

Abstract

Large language models have made substantial progress in mathematical reasoning. However, benchmark development for multilingual evaluation has lagged behind English in both difficulty and recency. Recently, GSM-Symbolic (Mirzadeh et al., 2025) showed a strong evidence of high variance when models are evaluated on different instantiations of the same question; however, the evaluation was conducted only in English. In this paper, we introduce MGSM-Pro, an extension of MGSM dataset with GSM-Symbolic approach. Our dataset provides five instantiations per MGSM question by varying names, digits and irrelevant context. Evaluations across nine languages reveal that many low-resource languages suffer large performance drops when tested on digit instantiations different from those in the original test set. We further find that some proprietary models, notably Gemini 2.5 Flash and GPT-4.1, are less robust to digit instantiation, whereas Claude 4.0 Sonnet is more robust. Among open models, GPT-OSS 120B and DeepSeek V3 show stronger robustness. Based on these findings, we recommend evaluating each problem using at least five digit-varying instantiations to obtain a more robust and realistic assessment of math reasoning.

1 Introduction

Large language models (LLMs) have drastically improved in capability in recent years, particularly on challenging knowledge-intensive and reasoning tasks, with open models closing the gap as evidenced by public benchmarks (Liu et al., 2024; Yang et al., 2025; Gemma-Team et al., 2025). However, progress in developing benchmarks for multilingual settings, particularly for mathematical reasoning has lagged behind English in both difficulty and recency, [1] making existing multilingual benchmarks easily saturated and potentially prone to memorization or being over-optimized (Shi et al., 2022b; Chen et al., 2024).

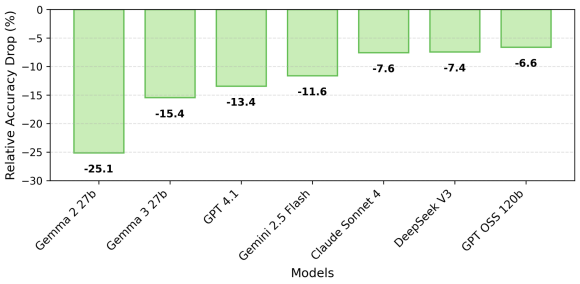

Figure 1: Relative decrease in accuracy from the original dataset, averaged across the six variants of MGSM-Pro, and aggregated over all nine languages.

One way to address this issue is to create new benchmark that are more recent such as MMath (Luo et al., 2025) and PolyMath (Wang et al., 2025) often translated from existing English benchmark but without modification of the numbers and context. However, it remains unclear whether LLMs evaluated on these benchmarks generalize to other similar problems; moreover, there is evidence that many models perform reasoning primarily in English (Tam et al., 2025; Qi et al., 2025). Even in English, there is strong evidence of high variance when presented different instantiations of the same question (known as *GSM-Symbolic*) (Mirzadeh et al., 2025). We carefully extend this finding to the multilingual setting.

In this paper, we introduce **`MGSM-Pro`**, a multilingual extension of `GSM-Symbolic` based on MGSM dataset (Shi et al., 2022a) in two steps: (1) *template construction* in English that allows easy replacement of names and digits (2) *dataset construction* that translates the template to multiple languages (with an LLM), followed by human verification—this helps to generate different instantiations of same question (e.g. 5 instances). Our results reveal a more precarious setting than GSM-Symbolic: low-resource languages experience a sharp perfor-

[1] E.g. AIME `https://artofproblemsolving.com/wiki/index.php/AIME_Problems_and_Solutions`

mance drop when accuracy is averaged over five instances instead of a single example, unlike high-resource languages. As shown in Figure 1, GPT-OSS 120B and DeepSeek V3 are more robust to this degradation, whereas smaller-sized models like Gemma 2 27B struggle to maintain accuracy relative to the original dataset.

Based on our findings on nine typologically-diverse languages, **we recommend that math reasoning evaluation should be performed on minimum of five instances** of the same problem by modifying digits.[2] We are ***releasing the new dataset*** (**MGSM-Pro**) with more instances to encourage a more robust evaluation. Similar to how we expect a good student that understands a sample problem to be able to solve various instances with modified digits. We expect both open LLMs and proprietary LLMs to be robust to these small changes.

2 Related Work

Math Reasoning Benchmarks With the increase of interest in evaluating a model's logical reasoning capabilities, multiple English math benchmarks have been introduced (Cobbe et al., 2021; Hendrycks et al., 2021; Mishra et al., 2022; Patel et al., 2021; Miao et al., 2020). Extending the investigation into the multilingual setting, (Shi et al., 2022a; Adelani et al., 2025) notices weaker model performances under low-resource language setting. However, it is unclear if success on these benchmarks translates to effectiveness on related problems or memorization of test set.

Robustness in Reasoning True logical reasoning requires robustness to minor variations and noise. Several English datasets highlight significant accuracy drops in such scenarios (Shi et al., 2023; Abedin et al., 2025; Mirzadeh et al., 2025). However, their studies remain limited to English. Our work introduces MGSM-Pro, a new dataset that expands these investigations to multilingual setting.

3 The MGSM-Pro Dataset

MGSM-Pro covers nine languages with various resource levels as defined by Joshi et al. (2020). This includes high-resource languages or HRLs (**English**, **Chinese**, **French**, and **Japanese**; Class 5) and low-resource languages or LRLs (**Swahili**, **Amharic**, **Igbo**, **Yoruba**, and **Twi**; Classes 1–2).

[2] i.e. evaluating on 1125 instances of MGSM rather than 225 problems for a more robust evaluation

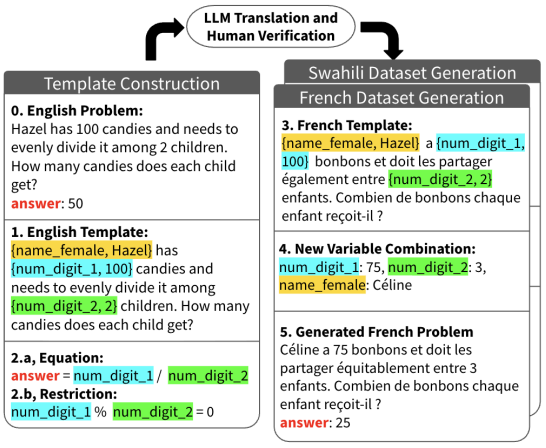

Figure 2: Workflow diagram illustrating the template creation and final data construction on sample language.

We also cover six dataset variants per language. These variations are organized into two series: Symbolic (SYM) and Irrelevant Context (IC). Each series consists of three distinct variations.

The Symbolic Series (SYM) involves systematic modifications to a problem's surface features without altering its logical structure. This series includes three variants: **SYM_N**, which replaces *names* with culturally relevant ones; **SYM_#**, which changes *numerical* data; and **SYM_N#**, which varies *both names and numbers* simultaneously.

The Irrelevant Context Series (IC) mirrors the modifications in the **SYM** series but introduces a distinct layer of difficulty as it inserts an irrelevant sentence to the problem. The resulting variants are denoted as **IC_N**, **IC_#**, and **IC_N#**.

In this section, we introduce the methodology for constructing MGSM-Pro with two steps: template construction (§3.1) and dataset construction (§3.2). Figure 2 shows an example of the data generation workflow in which names and digits are first identified and replaced with multiple instances.

3.1 Template Construction

The foundation of our dataset lies in the creation of adaptable templates. We adopt the GSM-Symbolic framework to generate symbolic templates for 225 out of 250 English MGSM questions. To simplify cross-lingual transfer, we restrict parametrization strictly to names and numbers (i.e. SYM). Each template includes a symbolic equation alongside variable constraints to ensure that generated combinations yield correct, logical answers. Once the English template is crafted, we employ Gemini 2.0 Flash to generate multilingual templates. These translations then undergo a rigorous verification

	Gemma 3 27B				GPT-OSS 120B				Deepseek V3				Claude Sonnet 4				Gemini 2.5 Flash			
Language	D_O	IC_N	SYM_#	IC_#	D_O	IC_N	SYM_#	IC_#	D_O	IC_N	SYM_#	IC_#	D_O	IC_N	SYM_#	IC_#	D_O	IC_N	SYM_#	IC_#
English	96.0	94.5	80.9	77.3	96.9	95.0	93.9	91.9	98.7	95.5	93.3	91.6	98.2	96.9	91.6	90.6	97.3	94.8	80.7	81.3
Chinese	88.4	86.4	74.4	70.4	92.0	91.7	90.3	88.1	92.4	92.0	90.1	88.9	93.3	91.8	88.9	87.6	90.2	89.4	79.8	75.8
French	90.2	83.8	74.2	70.1	92.0	89.2	87.4	86.4	91.1	89.2	88.8	86.2	92.9	91.7	86.3	86.2	92.4	87.8	77.1	74.8
Japanese	85.8	78.8	67.3	62.4	89.3	82.7	81.0	80.6	88.9	82.8	80.1	79.9	91.1	83.8	80.6	79.1	87.6	81.6	72.6	69.8
Swahili	89.8	85.4	72.4	70.9	84.9	83.7	81.5	78.4	92.0	91.2	87.8	85.8	92.4	92.6	86.0	85.9	92.4	91.1	78.8	80.4
Amharic	71.1	68.7	57.1	55.0	60.4	60.1	57.3	53.6	76.4	74.8	73.0	67.2	83.1	80.2	74.1	73.9	82.2	82.8	70.5	68.3
Igbo	65.3	56.1	51.4	41.3	77.8	73.4	69.4	65.4	75.1	69.4	66.4	60.9	79.6	78.5	68.7	67.8	82.2	81.4	69.4	67.9
Yoruba	45.3	42.1	38.3	31.2	77.3	69.7	66.4	61.8	67.6	59.2	57.6	54.1	77.8	75.5	67.7	67.6	84.9	82.7	69.2	68.4
Twi	19.6	11.9	15.1	9.4	44.9	36.9	39.6	32.4	48.9	38.0	40.4	30.3	54.7	47.1	45.9	39.5	66.2	62.8	55.7	52.0
Average	72.4	67.5	59.0	54.2	79.5	75.8	74.1	71.0	81.2	76.9	75.3	71.1	84.8	82.0	76.6	75.3	86.2	83.8	72.7	71.0

Table 1: Different models' accuracy across different dataset variations (SYM_#, IC_N, IC_#) and original (D_O).

process: they are first reviewed by native speakers, followed by automated alignment checks against the English source. Any template failing these checks is subjected to a second round of human correction. Finally, to enable a controlled increase in difficulty, we build upon GSM-IC's (Mirzadeh et al., 2025) methodology to create irrelevant context templates for every english question. We applied similar rigorous check as SYM questions.

3.2 Dataset Construction

To efficiently generate a large quantity of problem instances that share the same underlying logical structure, we leverage the symbolic equations and restrictions defined during the template phase. This methodology enables systematic sampling of new numerical values that are guaranteed to be mathematically valid and distinct from those present in the original training data.

A limitation in previous datasets, such as MGSM and AfriMGSM, was the reliance on direct translations, where names were frequently phonetic transliterations of English origin. This approach compromised the problems' local fit and cultural meaning. To ensure deep cultural relevance across all languages in MGSM-Pro, we tasked native annotators with curating a comprehensive repository of entities specific to their locale. This includes categories such as cities, personal names, and common pet names, guaranteeing that the generated problems resonate well with native speakers and accurately represent the target language's culture.

4 Experiments

4.1 Experiment Setup

Models evaluated We benchmark 12 models in a zero-shot setting across six variations within the SYM and IC series for each language. To ensure robustness, every variation is evaluated five times using different values and we report the mean performance across these iterations. We report the results of **original data** (D_O), IC_N, IC_# and SYM_# in the main paper, and others in Appendix D.4.

Prompts The prompt is structured to ensure the model adheres to the CoT format while including clear instructions to help numerical result capture. Our prompt suggests reasoning in English since previous work show LLM reason better in English (Tam et al., 2025; Qi et al., 2025).

Listing 1: Prompt

```
Explain your reasoning step by step in
    clear English to solve the problem.
Your response should end with the final
    numerical answer, without including
    units.
Question: {question}
```

4.2 Results

4.2.1 Main results

Table 1 shows the result of five LLMs: Gemma 3 27B, GPT-OSS 120B, DeepSeek V3, Claude Sonnet 4 and Gemini 2.5 Flash.

LLM performance is less sensitive to name variation Simply changing the names of person or items (i.e. SYM_N setting) does not necessarily hurt performance. However, when irrelevant contexts are added (i.e. IC_N), models experience a slight drop in performance. In general IC_N is more critical for LRLs such as Twi than HRLs. Also, we find proprietary models to be more robust to this drop, for example on average Gemini 2.5 Flash accuracy on all languages dropped by -2.4 while Gemma 3 27B and GPT-OSS 120B dropped by -4.9 and -3.7 respectively.

Numerical variation leads to huge drop in performance While names variation only leads to

small drop, changing numbers used in the questions leads to huge drop in performance especially when combined with irrelevant contexts. On average, all models experienced a performance drop of at least -8 points across all languages in the IC_# setting. However, the biggest drop comes from the Gemma 3 27B and Gemini 2.5 Flash with -18.2 and -15.2 respectively. While Gemini 2.5 Flash have the overall best performance among all the tested models on D_o configuration, it is less robust to numeric variation compared to other competitive models such as Claude Sonnet 4.

High-resource languages are more robust to variation Excluding the two models with the biggest drop (Gemma 3 27B and Gemini 2.5 Flash), we find that the HRLs (English, Chinese, and French) often have smaller drop in performance (< 8.0 point) compared to LRLs (Amharic, Igbo, Yoruba and Twi) with bigger drop. The only exception is sometimes for Japanese and Swahili. IC_# setting often leads to big drop for Japanese, although it is a HRL especially DeepSeek V3 (-9.0) and Claude Sonnet 4 (-12.0). On the otherhand, Swahili which is supposed to be LRL, often have drop similar to HRLs e.g. -6.2 on DeepSeek V3 and -6.5 on Claude Sonnet 4.

4.2.2 Model size vs. Robustness

Figure 3 shows the effect of scaling of model sizes and robustness to change in names and numbers (IC_N#). There is no clear pattern across different model architectures. For Gemma family of models, the drop in performance gets worse as the model parameters increases from 4B, 12B and 27B. However, for GPT-OSS, we have the opposite trend where bigger model size is more robust to the performance drop (see Appendix D.2). Surprisingly, we find GPT-OSS 120B more robust to degradation than GPT-4.1 which may be of bigger parameter size since it is a closed model (see Appendix D.4).

4.3 Reliability of Leaderboard ranking

Most leaderboard ranking for math reasoning are based on one instance. Our results on Table 2 shows that the model ranking is not persistent after introducing five instances of the same question when both numbers and names are changed with irrelevant contexts (i.e. IC_N#). While Gemini 2.5 Flash gave the best result on the original dataset (D_O), if we performed evaluation and averaged on five different instances (Avg-5), there is a systematic performance drop across all models. Notably,

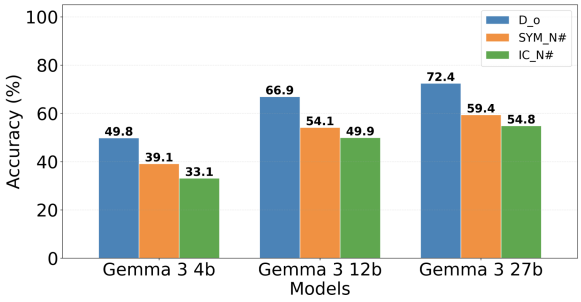

Figure 3: **Relative Accuracy Decline Across Gemma 3 Family** The figure illustrates the relative decline in accuracy for the Gemma-3 family. The drop is measured from the original dataset to two configurations: IC_N# and SYM_N#. Averaged over nine languages.

Model	D_O (%)	Avg-5 (%)	Avg-10 (%)
Gemini Flash 2.5	86.2 [1]	71.2 [4]↓	70.6 [4]
Claude Sonnet 4	84.8 [2]	74.8 [1]↑	74.5 [1]
Gemini Flash 2.0	84.3 [3]	69.3 [5]↓	68.7 [5]
DeepSeek V3	81.2 [4]	71.8 [2]↑	71.7 [2]
GPT-4.1	80.4 [5]	64.0 [6]↓	63.8 [6]
GPT-OSS 120b	79.5 [6]	71.4 [3]↑	71.1 [3]
Gemma 3 27B	72.4 [7]	54.8 [7]	54.0 [7]
GPT-OSS 20b	67.7 [8]	53.0 [8]	53.1 [8]
Gemma 3 12B	66.9 [9]	49.9 [9]↓	49.8 [9]
Gemma 2 27B	58.8 [10]	35.1 [10]	34.8 [10]
Llama 3.1 70B	52.2 [11]	28.3 [12]↓	28.3 [12]
Gemma 3 4B	49.8 [12]	33.1 [11]↑	33.2 [11]

Table 2: Ranking of model average accuracy over 9 languages on original dataset (D_O) compared on IC_N#. Arrows show change vs. the previous column.

the ranking of the best model drops to fourth place. Interestingly, we find that Claude Sonnet 4 now moved from the second rank to first. Repeating the experiments 10 times (Avg-10), gave exactly the same results as the Avg-5. This findings is interesting, since varying the questions with five instances already gave a more robust, and realistic estimation of math reasoning for the language and LLM. We therefore recommend, math reasoning evaluation should use Avg-5 setting as the default.

5 Conclusion

In this paper, we investigated the robustness of LLM evaluation for math reasoning when presented with multiple instantiation of the same question by varying names, digits and adding irrelevant contexts. All LLMs achieved significant drop in performance especially for low-resource languages. We developed MGSM-Pro, an extension of MGSM with five new instances per question to encourage more robust and realistic evaluation across nine typologically diverse languages.

6 Limitations

Our study has a few limitations. First, our dataset covers nine languages due to difficulty in finding reliable native annotators. Expanding MGSM-Pro to other languages such as Tamil would provide a more complete picture of multilnigual mathematical robustness. Moreover, our evaluation covers only 12 models because of limited compute budget. It remains to be seen how other model families, such as Qwen3 or reasoning models like GPT-5, would perform. Thirdly, we only evaluated the models with a prompt that instruct reasoning in English. It would be interesting to see how models perform when prompted to reason in other languages. Fourth, while the MGSM dataset contains 250 questions, our version utilizes 225. The remaining 25 were excluded because their numerical equation and restrictions are flagged as wrong by human verification and also because there were difficulties in generating multiple unique numerical instances. We plan on expanding the dataset to cover all 250 questions before the end of the reviewing period.

References

Zain Ul Abedin, Shahzeb Qamar, Lucie Flek, and Akbar Karimi. 2025. Arithmattack: Evaluating robustness of llms to noisy context in math problem solving. In *Proceedings of the LLMSEC Workshop at the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*.

David Ifeoluwa Adelani, Jessica Ojo, Israel Abebe Azime, Jian Yun Zhuang, Jesujoba O. Alabi, Xuanli He, Millicent Ochieng, Sara Hooker, Andiswa Bukula, En-Shiun Annie Lee, Chiamaka Chukwuneke, Happy Buzaaba, Blessing Sibanda, Godson Kalipe, Jonathan Mukiibi, Salomon Kabongo, Foutse Yuehgoh, Mmasibidi Setaka, Lolwethu Ndolela, and 8 others. 2025. Irokobench: A new benchmark for african languages in the age of large language models. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Nuo Chen, Zinan Zheng, Ning Wu, Ming Gong, Dongmei Zhang, and Jia Li. 2024. 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.

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. *Computing Research Repository*, arXiv:2110.14168. Introduces the GSM8K dataset.

Gemma-Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, and 1 others. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

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. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:24933–24949.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Wenyang Luo, Wayne Xin Zhao, Jing Sha, Shijin Wang, and Ji-Rong Wen. 2025. MMATH: A multilingual benchmark for mathematical reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 11187–11202, Suzhou, China. Association for Computational Linguistics.

Shen-Yun Miao, Chao-Chun Liang, and Keh-Yih Su. 2020. A diverse corpus for evaluating and developing english math word problem solvers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 8344–8355.

Seyed Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. 2025. GSM-symbolic: Understanding the limitations of mathematical reasoning in large language models. In *The Thirteenth International Conference on Learning Representations*.

Swaroop Mishra, Arindam Mitra, Neeraj Varshney, Bhavdeep Sachdeva, Peter Clark, Chitta Baral, and Ashwin Kalyan. 2022. Numglue: A suite of fundamental yet challenging mathematical reasoning tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, page to appear.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 4074–4085.

Jirui Qi, Shan Chen, Zidi Xiong, Raquel Fernández, Danielle Bitterman, and Arianna Bisazza. 2025. When models reason in your language: Controlling thinking language comes at the cost of accuracy. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 20279–20296.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pages 30833–30848.

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. 2022a. Language models are multilingual chain-of-thought reasoners. *Computing Research Repository*, arXiv:2210.03057. Introduces the MGSM benchmark.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, and 1 others. 2022b. Language models are multilingual chain-of-thought reasoners. In *The Eleventh International Conference on Learning Representations*.

Zhi Rui Tam, Cheng-Kuang Wu, Yu Ying Chiu, Chieh-Yen Lin, Yun-Nung Chen, and Hung-yi Lee. 2025. Language matters: How do multilingual input and reasoning paths affect large reasoning models? *arXiv preprint arXiv:2505.17407*.

Yiming Wang, Pei Zhang, Jialong Tang, Haoran Wei, Baosong Yang, Rui Wang, Chenshu Sun, Feitong Sun, Jiran Zhang, Junxuan Wu, and 1 others. 2025. Polymath: Evaluating mathematical reasoning in multilingual contexts. *arXiv preprint arXiv:2504.18428*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

A MGSM-Pro Dataset

A.1 Language Details

The resource levels and language families of the nine languages in MGSM-Pro are shown in Table 3. Each language has 225 question templates out of the 250 MGSM questions.

Language	Code	Language Family	Joshi Class
English	eng_Latn	Indo-European	Class 5
Chinese	zho_Hans	Sino-Tibetan	Class 5
French	fra_Latn	Indo-European	Class 5
Japanese	jpn_Jpan	Japonic	Class 5
Swahili	swh_Latn	Niger-Congo	Class 2
Amharic	amh_Ethi	Afro-Asiatic	Class 2
Igbo	ibo_Latn	Niger-Congo	Class 1
Yoruba	yor_Latn	Niger-Congo	Class 2
Twi	twi_Latn	Niger-Congo	Class 1

Table 3: Selected languages categorized by ISO code, linguistic family, and resource availability (Joshi Class).

A.2 Name Categories

We categorize the replaced name entities into three domains: People, Places, and Beasts. As shown in Table 4, annotators provided a set of culturally specific names for each name type to ensure local relevance. In cases where a suitable local equivalent did not exist, english alternatives would be provided.

Domain	Name Types
People	Male name, Female name, Family name
Places	City name, Mountain name
Beasts	Dragon name, Dinosaur name, Cat name

Table 4: Grouped name variables categorized by domain

A.3 IC Template Construction

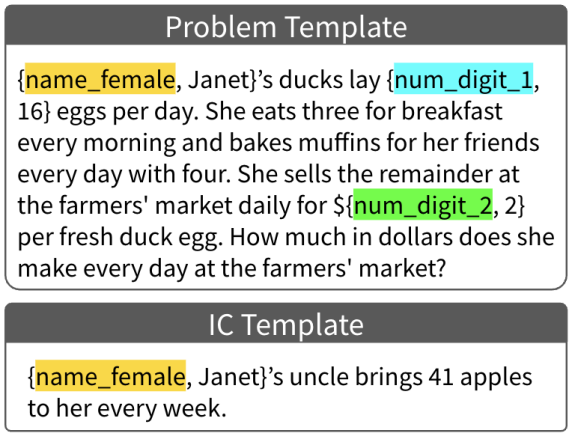

Figure 4: Example of a MGSM-Pro question template alongside its IC sentence template

Each question in the MGSM-Pro dataset is paired with a corresponding IC sentence template. The curation of IC sentence template follows the methodology of (Shi et al., 2023), where we ensure that the irrelevant sentences have: 1) some related connection with the problem and 2) uses names found in the question. An examplar is shown in Figure 4.

B Prompts for Large Language Models

B.1 Evaluation Prompts

Model Evaluation Prompt

Explain your reasoning step by step in clear English to solve the problem. Your response should end with the final numerical answer, without including units.

B.2 Template Construction Prompts

Multilingual Template Translation Prompt

Your task is to convert an English template into a native-language template, preserving all placeholder formats. Do not change the ordering of words of the output sentence, just label them with brackets.
Input:
- English template (with placeholders)
- Native sentence (no placeholders)
Output:
- Native sentence with the same placeholders, matching values and positions.
Placeholder formats:
- Names: {name_male, xxx}, {name_female, xxx}
- Digits: {num_digit, xxx}
Guidelines:
1. Tag all placeholders from English in the native sentence.
2. Names may differ across languages (e.g. James → Jacques) — match by position.
3. Always tag the first word if it's a person name.
4. Do not reword the native sentence in anyway, you should just be inserting the variable names and brackets

Input: {english template}
Native: {native question}
Output:

C Instructions for Annotators

This section provides a brief introduction to the annotation guide for the MGSM-Pro dataset. We categorize the MGSM-Pro annotation process into two main tasks: 1) correcting native templates, and 2) providing native names

C.1 Template Correction Annotation

```
The goal of our study is to
    ↪ create variations of the
    ↪ same problem by changing
    ↪ names and numbers within
    ↪ the math problem while
    ↪ keeping the logic intact.
    ↪ The templates from your
    ↪ annotation process will be
    ↪ used in the future to
    ↪ create math dataset.

For each problem template
    ↪ correction, 3 items will be
    ↪  provided for you to use.

1. English Template
This is the gold template. You
    ↪ should make sure the native
    ↪  language template is as
    ↪ similar to the english
    ↪ template as possible.

2. Original Native Question
This is the original native
    ↪ question in the dataset.
    ↪ You should use this as a
    ↪ reference alongside the
    ↪ English template to judge
    ↪ if the Native language
    ↪ template is correct.

3. Native Language Template
This is a machine-created native
    ↪ language template. It could
    ↪  very likely contain errors
    ↪ . This is the template that
    ↪  you will judge if it is
    ↪ correct or not.

Below are the five critierias the
    ↪  native language template
    ↪ must achieve in order to be
    ↪  considered as correct.
```

```
1. Native Language Templates will need to contain the original question. I.E. the wording of the native template should not change from the native question, the template should only be adding in the variable names. If this is not the case, you should ignore the Native Language Template and please provide the new annotated template inside the correction column

2. No missing variable annotation. I.E. all names or digits tagged in English template is tagged in the native language template. You should add the corresponding {type, value} annotation around the target language word or number.

3. No extra annotation. I.E. there is no extra variables annotated in the translation but was not in the English template. You should remove any { } markers around words or numbers that were not annotated in English.

4. No incorrect bracket {} span. I.E. the annotated span is not too long or too short. You should adjust the braces so they exactly enclose the intended word or number, matching the English span.
```

C.2 Native Name Annotation

```
You will be given eight types of name.

You will need to provide 10 names to each types that fit into your native language. The names should be relevant to your specific language and not English names. However, if there does not exist 10 unique names for a specific name category but at least one, it is fine to provide less. Moreover, if there are no native names for a specific name category, you can provide english substitutes .

The list of name types are as follows:

Male name, Female name, Family name, City name, Mountain name, Dragon name, Dinosaur name, Cat name
```

D Experiment Details

D.1 Evaluation Setup

Experiments were conducted on a high-performance cluster (up to 8 nodes NVIDIA H100GPUs) using VLLM (Kwon et al., 2023). For the proprietary models and DeepSeek V3, we utilized API keys to perform evaluation

D.2 Model size vs Robustness

Figure 5 shows the relationship between model size and robustness to name and number variations within the GPT-OSS family. Unlike the trends observed in the Gemma 3 family, larger GPT-OSS models show better robustness to performance drops. The contradictory findings across different model families suggests that simply increasing model scale does not automatically improve robustness; instead, other factors like training methodologies and data mixtures likely play a more significant role.

D.3 Leaderboard Rankings

As shown in Table 5, model rankings fluctuate when models are tested on five distinct instances of the same question with altered numbers and names (i.e., SYM_N#). Mirroring the trends in Table

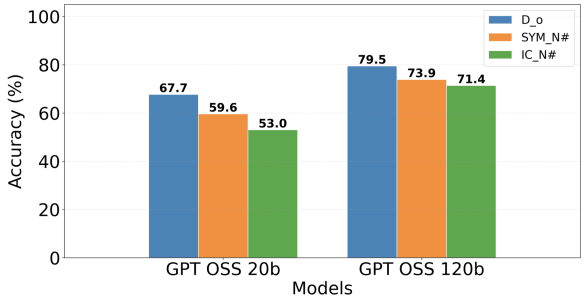

Figure 5: **Relative Accuracy Decline Across GPT-OSS Family** The figure illustrates the relative decline in accuracy for the GPT-OSS family. The drop is measured from the original dataset to two configurations: `IC_N#` and `SYM_N#`. Averaged over nine languages.

Table 2, we observe significant performance degradation and ranking instability when moving from the original to the 5-instance average. However, the rankings stabilize between the 5-instance and 10-instance averages.

Model	Orig (%)	Avg-5 (%)	Avg-10 (%)
Gemini Flash 2.5	86.2 [1]	72.9 [4]↓	72.4 [4]
Claude sonnet 4	84.8 [2]	77.1 [1]↑	76.7 [1]
Gemini Flash 2.0	84.3 [3]	72.4 [5]↓	71.7 [5]
DeepSeek V3	81.2 [4]	75.4 [2]↑	75.3 [2]
GPT-4.1	80.4 [5]	66.6 [6]↓	66.2 [6]
GPT-OSS 120b	79.5 [6]	73.9 [3]↑	74.2 [3]
Gemma 3 27B	72.4 [7]	59.4 [8]↓	58.5 [8]
GPT-OSS 20b	67.7 [8]	59.6 [7]↑	59.7 [7]
Gemma 3 12B	66.9 [9]	54.1 [9]	54.0 [9]
Gemma 2 27B	58.8 [10]	40.0 [10]	39.7 [10]
Llama 3.1 70B	52.2 [11]	34.9 [12]↓	34.4 [12]
Gemma 3 4B	49.8 [12]	39.1 [11]↑	38.6 [11]

Table 5: Ranking of model average accuracy over 9 languages on original dataset compared on `SYM_N#`. Arrows show change vs. the previous column.

D.4 Full Experiment Results

	Gemma 3 4B							Gemma 3 12B							Gemma 3 27B						
Language	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#
English	86.2	87.3	82.8	68.4	62.6	68.3	62.8	92.9	93.7	93.9	78.0	78.1	77.9	78.2	96.0	95.1	94.5	80.9	77.3	81.2	77.7
Chinese	77.8	78.5	69.3	61.0	53.8	61.7	54.0	87.1	88.7	86.3	70.5	65.8	72.4	67.1	88.4	90.4	86.4	74.4	70.4	75.1	70.8
French	79.6	78.8	69.6	61.0	54.4	63.1	55.7	88.9	88.0	83.4	71.7	67.9	71.2	66.8	90.2	89.1	83.8	74.2	70.1	73.3	69.8
Japanese	70.2	65.3	58.4	52.2	43.6	50.8	42.0	84.0	79.1	77.6	64.0	58.7	63.5	58.5	85.8	81.1	78.8	67.3	62.4	66.8	63.6
Swahili	60.9	64.8	57.8	49.5	41.0	50.9	40.8	83.1	87.0	82.0	68.9	62.6	69.5	62.4	89.8	86.9	85.4	72.4	70.9	71.6	69.1
Amharic	43.1	43.7	38.1	32.8	28.0	34.2	29.2	68.4	71.1	68.0	57.2	51.6	55.6	54.1	71.1	72.4	68.7	57.1	55.0	58.0	54.8
Igbo	16.9	15.3	13.8	11.6	8.4	11.4	8.3	56.0	55.6	46.0	39.6	35.4	42.4	35.8	65.3	66.8	56.1	51.4	41.3	51.7	44.1
Yoruba	8.4	9.7	6.3	5.5	4.2	6.4	4.1	32.9	34.3	29.5	23.9	22.0	25.9	21.3	45.3	49.1	42.1	38.3	31.2	38.8	34.3
Twi	4.9	5.1	1.4	3.8	1.1	4.7	0.8	8.4	9.6	7.1	9.3	5.7	8.5	4.8	19.6	19.7	11.9	15.1	9.4	18.2	9.6
Average	49.8	49.8	44.2	38.4	33.0	39.1	33.1	66.9	67.5	63.7	53.7	49.7	54.1	49.9	72.4	72.3	67.5	59.0	54.2	59.4	54.8

Table 6: Different models' accuracy across different dataset variations (D_O, SYM_N, IC_N, SYM_#, IC_#, SYM_N#, IC_N#) for each language

	Gemma 2 27B							Gemini Flash 2.0							Gemini Flash 2.5						
Language	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#
English	76.0	80.4	80.3	57.0	54.0	55.3	52.4	96.4	94.6	94.7	85.6	85.2	85.4	84.3	97.3	95.5	94.8	80.7	81.3	82.3	81.2
Chinese	82.7	87.5	81.3	55.8	52.6	58.2	52.4	87.1	90.4	86.9	80.3	78.8	79.4	76.8	90.2	91.4	89.4	79.8	75.8	80.2	78.7
French	67.6	70.3	59.7	49.2	38.6	51.0	41.9	92.0	88.9	89.1	77.9	76.7	78.4	76.1	92.4	89.8	87.8	77.1	74.8	75.7	74.6
Japanese	79.6	73.5	67.5	49.2	42.9	50.1	43.8	84.9	82.8	80.3	72.3	67.6	70.7	68.2	87.6	82.8	81.6	72.6	69.8	71.2	70.5
Swahili	82.7	83.5	76.7	54.6	49.0	55.8	49.3	93.3	91.6	89.8	80.6	78.0	80.4	77.5	92.4	91.6	91.1	78.8	80.4	79.7	80.1
Amharic	40.0	47.9	41.0	25.4	24.9	26.0	24.7	80.0	80.3	80.4	68.0	69.1	69.4	69.1	82.2	82.8	82.8	70.5	68.3	71.9	70.2
Igbo	44.9	45.6	41.6	26.4	22.3	28.3	23.3	79.6	81.2	76.7	66.7	63.3	67.4	62.9	82.2	82.2	81.4	69.4	67.9	68.4	67.3
Yoruba	35.6	34.4	30.7	19.9	17.5	19.6	16.6	81.3	78.6	72.9	64.6	62.6	66.7	61.9	84.9	84.7	82.7	69.2	68.4	71.0	67.2
Twi	20.0	21.7	13.5	12.9	9.6	16.0	11.7	64.0	66.7	58.6	50.9	47.5	53.7	46.9	66.2	68.2	62.8	55.7	52.0	55.7	50.7
Average	58.8	60.5	54.7	38.9	34.6	40.0	35.1	84.3	83.9	81.0	71.9	69.9	72.4	69.3	86.2	85.4	83.8	72.7	71.0	72.9	71.2

Table 7: Different models' accuracy across different dataset variations (D_O, SYM_N, IC_N, SYM_#, IC_#, SYM_N#, IC_N#) for each language

	GPT-OSS 20B							GPT-OSS 120B							GPT-4.1						
Language	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#
English	96.4	94.2	88.1	90.0	81.9	90.2	82.6	96.9	97.2	95.0	93.9	91.9	92.7	91.8	96.9	95.6	93.5	82.8	79.6	82.5	81.0
Chinese	88.9	89.6	85.0	84.3	81.6	84.8	82.0	92.0	93.2	91.7	90.3	88.1	89.0	89.7	90.2	92.9	91.6	77.8	77.5	79.6	76.8
French	88.9	88.8	84.7	84.4	81.3	83.4	81.2	92.0	90.0	89.2	87.4	86.4	86.1	86.0	90.2	88.4	87.4	76.5	74.3	76.3	75.2
Japanese	85.3	79.7	73.5	76.4	73.0	74.6	72.4	89.3	84.5	82.7	81.0	80.6	80.8	80.4	88.4	83.8	82.6	71.7	70.8	72.5	71.3
Swahili	74.7	74.7	63.8	66.2	56.2	65.6	55.5	84.9	86.6	83.7	81.5	78.4	80.4	78.9	92.4	91.0	90.0	78.9	79.1	80.1	78.7
Amharic	41.3	35.9	25.5	31.3	21.5	31.9	22.4	60.4	64.4	60.1	57.3	53.6	59.3	53.3	66.7	67.9	67.1	52.1	51.5	51.7	51.4
Igbo	57.3	58.1	46.6	49.0	39.8	48.6	38.0	77.8	79.0	73.4	69.4	65.4	69.0	65.1	80.4	78.1	72.3	62.0	56.8	60.7	57.2
Yoruba	52.0	48.1	37.3	38.9	31.5	38.0	31.1	77.3	72.7	69.7	66.4	61.8	65.1	62.1	74.2	76.2	70.8	61.3	58.0	60.1	56.1
Twi	24.4	19.8	12.9	18.8	11.7	19.7	11.9	44.9	46.9	36.9	39.6	32.4	42.7	35.2	44.4	46.1	36.0	33.2	29.5	35.8	28.0
Average	67.7	65.4	57.5	59.9	53.2	59.6	53.0	79.5	79.4	75.8	74.1	71.0	73.9	71.4	80.4	80.0	76.8	66.3	64.1	66.6	64.0

Table 8: Different models' accuracy across different dataset variations (D_O, SYM_N, IC_N, SYM_#, IC_#, SYM_N#, IC_N#) for each language

	Llama 3 70B							DeepSeek V3							Claude Sonnet 4						
Language	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#	D_O	SYM_N	IC_N	SYM_#	IC_#	SYM_N#	IC_N#
English	96.9	94.4	91.6	69.6	66.2	68.4	65.5	98.7	97.3	95.5	93.3	91.6	93.2	90.4	98.2	97.5	96.9	91.6	90.6	91.6	89.4
Chinese	69.3	74.6	65.8	52.3	41.6	51.8	46.1	92.4	93.3	92.0	90.1	88.9	90.0	88.2	93.3	94.0	91.8	88.9	87.6	89.3	86.8
French	75.1	76.2	63.3	47.8	40.4	50.1	41.0	91.1	90.6	89.2	88.8	86.2	87.6	86.1	92.9	92.2	91.7	86.3	86.2	86.7	85.8
Japanese	69.3	64.6	55.6	41.7	34.0	41.5	31.7	88.9	83.7	82.8	80.1	79.9	80.6	80.0	91.1	85.1	83.8	80.6	79.1	79.5	78.5
Swahili	64.0	66.2	51.9	42.8	33.3	41.2	33.7	92.0	91.7	91.2	87.8	85.8	87.6	86.6	92.4	93.2	92.6	86.0	85.9	86.7	86.6
Amharic	15.1	21.0	5.4	13.2	4.0	11.2	3.6	76.4	77.4	74.8	73.0	67.2	73.2	68.6	83.1	82.8	80.2	74.1	73.9	74.3	73.1
Igbo	43.6	49.7	34.3	27.6	19.0	26.4	18.8	75.1	75.6	69.4	66.4	60.9	65.3	61.2	79.6	80.0	78.5	68.7	67.8	72.3	69.0
Yoruba	20.9	23.5	12.4	14.1	9.2	12.1	9.4	67.6	66.8	59.2	57.6	54.1	59.0	53.8	77.8	79.7	75.5	67.7	67.6	67.9	65.3
Twi	16.0	16.6	7.3	8.7	4.2	11.3	5.1	48.9	44.9	38.0	40.4	30.0	42.0	30.9	54.7	54.8	47.1	45.9	39.5	46.1	38.7
Average	52.2	54.1	43.1	35.3	28.0	34.9	28.3	81.2	80.2	76.9	75.3	71.7	75.4	71.8	84.8	84.4	82.0	76.6	75.3	77.1	74.8

Table 9: Different models' accuracy across different dataset variations (D_O, SYM_N, IC_N, SYM_#, IC_#, SYM_N#, IC_N#) for each language