LLMs cannot spot math errors, even when allowed to peek into the solution

KV Aditya Srivatsa **Kaushal Kumar Maurya** **Ekaterina Kochmar**
Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE
kvaditya.edu@gmail.com, {kaushal.maurya, ekaterina.kochmar}@mbzuai.ac.ae

Abstract

Large language models (LLMs) demonstrate remarkable performance on math word problems, yet they have been shown to struggle with *meta-reasoning* tasks such as identifying errors in student solutions. In this work, we investigate the challenge of locating the first error step in stepwise solutions using two error reasoning datasets: `VtG` and `PRM800K`. Our experiments show that state-of-the-art LLMs struggle to locate the first error step in student solutions even when given access to the reference solution. To that end, we propose an approach that generates an intermediate *corrected student solution*, aligning more closely with the original student's solution, which helps improve performance.[1]

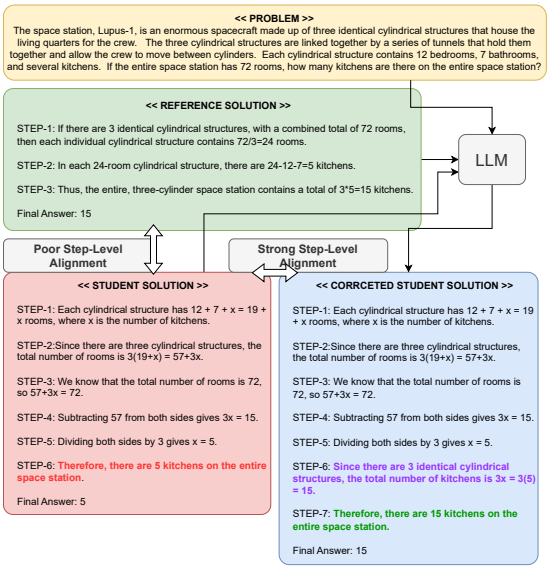

Figure 1: In this example, the *corrected* version of the original *student solution* (with `Llama3-70B`) makes the location of the student's first mistake more apparent as compared to the reference (i.e., gold) solution.

1 Introduction

Large language models (LLMs) demonstrate impressive performance on existing reasoning benchmarks, particularly on math word problems (Liu et al., 2024; Dubey et al., 2024; Achiam et al., 2023). For example, the state-of-the-art `Llama3.1-405B` (Dubey et al., 2024) model achieves 96.8% accuracy on the challenging `GSM8K` reasoning benchmark (Cobbe et al., 2021). However, recent work has revealed that models excelling at end-task accuracy often fail when probed about their underlying reasoning processes – what we refer to here as *meta-reasoning*. For instance, both Zeng et al. (2024) and Tyen et al. (2024) have reframed LLMs from passive problem solvers into active evaluators, revealing that even top-performing LLMs struggle with tasks like locating the first error step in a student's solution.

The ability to pinpoint and categorize errors is not only a critical diagnostic tool for understanding models' limitations but is also essential for developing assistive educational feedback tools in intelligent tutoring systems (Jia et al., 2024; Niño-Rojas et al., 2024). Accurate error detection and categorization promote better personalized and effective feedback generation (Anderson et al., 1990; Hattie and Timperley, 2007). This was demonstrated using LLMs by Daheim et al. (2024), where decomposing the feedback generation process to verify the student's solution before providing hints produces more reliable results. Existing research typically tests LLMs on *error localization* using only the original question and the student's erroneous solution (Zeng et al., 2024; Tyen et al., 2024). Pedagogical research indicates that when teachers have a canonical solution, they can "offload" the problem-solving process and focus on comparing student work against the expert path (Sweller et al., 1998; Carpenter et al., 1989). Motivated by this, we investigate **what information helps an LLM in locating the first error step in a math problem solution**. Our preliminary experiments suggest that

[1]All data and code are available at https://github.com/kvadityasrivatsa/llms-cannot-spot-math-errors

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 10914–10928

even when gold (reference) solution is provided, LLMs still struggle.

Therefore, we explore an alternative approach that generates an intermediate corrected version of the student's solution. This version preserves the student's method while applying only necessary structural adjustments, yielding a reference that both mirrors the student's reasoning and maintains consistency. We call it *corrected student solution*. Figure 1 shows a math problem from GSM8K with its reference solution, an incorrect stepwise student solution from the VtG (Daheim et al., 2024) dataset, and a corrected version produced by Llama3-70B LLM (Grattafiori et al., 2024). Aligning the student's solution with the ground truth is crucial to pinpoint the first error but is challenging due to: **(1) Poor Step Alignment**: the student's 6 steps versus the reference's 3 steps do not correspond directly, and there are no matching intermediate variables until the 4th step; and **(2) Different Approaches**: the student introduces an unknown variable x, while the reference follows a more direct method. The corrected student solution shows that although the student computes the number of kitchens per cylindrical structure correctly, they overlook calculating the total number of kitchens in the space station. By updating the solution to mirror the reference while retaining the student's approach, the corrected solution achieves better step alignment, simplifying error identification and error localization. Our analysis (see §C.1) shows that these generated corrected student solutions semantically align with student solutions better than the gold solutions.

In this paper, we formulate and investigate two key research questions: **RQ1**: Can LLMs accurately locate errors in incorrect math problem solutions when provided with access to the reference solution? and **RQ2**: Can the incorporation of intermediate reasoning steps – such as corrected student solution – enhance the overall performance of LLMs in the task of error localization?

Our experiments on two public datasets – VtG (Daheim et al., 2024) and PRM800K (Lightman et al., 2023) – confirm that state-of-the-art models like Llama3.1-405B and GPT-4o face significant difficulties in accurately localizing the first error even when furnished with the dataset-provided gold solution. In contrast, supplying a corrected student solution markedly improves error localization performance, especially for more capable models, which suggests that overall problem solving ability has little bearing on error detection accuracy.

2 Methodology

Data We perform our experiments on two error-reasoning datasets to investigate LLMs' capabilities across varying levels of problem difficulty and error typologies. The first dataset, referred to in this work as VtG, was released by Daheim et al. (2024) and comprises 1,002 incorrect stepwise student attempts on grade school-level math word problems in English. These attempts are sourced from MathDial (Macina et al., 2023) and originally from GSM8K (Cobbe et al., 2021), and include annotations for the first erroneous step, a description of the mistake, and its classification into one of seven error types (see §A for more details). The second dataset, PRM800K (Lightman et al., 2023), consists of 80,000 incorrect stepwise student solutions – with 2,077 designated for testing – each marked at the first error step, and features math questions drawn from the MATH (Hendrycks et al., 2021) dataset, with more advanced problems than those in GSM8K. Together, these datasets present diversity to explore error reasoning and meta-reasoning across elementary and advanced math levels.

LLMs We select a diverse array of 6 open and closed-source, as well as generic and fine-tuned LLMs for our experiments (listed in Table 6). Notably, Qwen2.5-72B-Math (Yang et al., 2024) has been fine-tuned for solving math problems, and LearnLM-1.5-Pro (Team et al., 2024) has been built for advanced pedagogical reasoning, guiding mistake discovery and providing constructive feedback. In addition to these properties, these models were chosen based on their problem-solving performance on the underlying math problems in our two datasets (see definition and scores in §B.3). In particular, while most models in our selection excel at the grade-school arithmetic problems from GSM8K (Cobbe et al., 2021), they exhibit varied performance on the more advanced questions from the MATH (Hendrycks et al., 2021) dataset.

Modeling Approach Let Q be the problem, G the reference solution, $S = \{s_i\}_{i=1}^{n}$ the student trace, and E the first erroneous step. An LLM_θ with prompt template P predicts E under three settings: (i) **problem + student** — $E = \text{LLM}_\theta\big(P(Q, S)\big)$; (ii) **problem + student + gold** — $E = \text{LLM}_\theta\big(P(Q, S, G)\big)$; (iii) **problem + student + correction** — first align the gold to the student trace, $S' = \{s'_j\}_{j=1}^{m} = \text{LLM}_\theta\big(P(G, S)\big)$, producing a structurally and stylistically matched

correction of S (see §C.1); then detect the error with Q, S, S': $E = \text{LLM}_\theta\big(P(Q, S, S')\big)$.

Gold Solution vs. Corrected Solution Among recent work, the approach of Li et al. (2024) is most similar to ours. They ask the model to generate a corrected version of the student's solution from scratch and then localize errors against that self-generated reference, so they find that their success depends on the model's own problem-solving ability. Instead, we cast the model as a "teacher": it is given the gold solution and tasked with generating a corrected version of the student's solution. Providing the gold answer disentangles error detection abilities from problem solving abilities. Since gold solutions in benchmarks like GSM8K and MATH often differ in style, step order, and content, we include a brief intermediate step to rewrite the gold solution to closely match that of the student, making only minimal edits needed for correctness.

To evaluate the quality of these corrections (S'), we manually annotated 90 randomly selected outputs across models for (1) correctness (overall and step-level) and (2) stylistic similarity to the student's work. A subset of 30 was double-annotated (Cohen's κ = 0.82 and 0.85 for correctness and stylistic similarity) (see §C.2 for more details). Most models produced accurate corrections in over 93.3% of cases and maintained stylistic similarity in 87.4%. The exception was `Qwen2.5-72B-Math`, which scored significantly lower on both metrics (69.6% correctness, 63.3% stylistic similarity), consistent with its weaker error localization performance (see §4.1).

3 Experimental Setup

To generate first error step predictions, we adapt the few-shot prompt from Daheim et al. (2024) and define four prompt types, as described in Section 2: **(1) *w/o-S*** (without gold solution) presents only the math problem and the student's incorrect stepwise solution, asking the LLM to identify the first error; **(2) *w-GS*** (with gold solution) additionally provides the dataset's stepwise gold solution; **(3) *w-Cor*** (with corrected student solution) first prompts the LLM to generate a corrected version of the student's solution—retaining their approach but fixing errors using the problem and gold solution—and then uses this in the main prompt; and **(4) random** selects a random error step within the student's solution span, averaged over 100 runs with different seeds. We also evaluate each LLM's problem-solving ability on both datasets to compare against their error localization performance. Exact prompt templates, LLM settings, and other details are provided in §B.1.

Model	VtG			PRM800K		
Random	18.32			9.52		
	w/o-S	*w-GS*	*w-Cor*	*w/o-S*	*w-GS*	*w-Cor*
Llama3-70B	42.51	49.50	**61.28**	19.64	24.12	**33.03**
Llama3.1-70B	49.10	57.98	**64.17**	24.46	34.23	**38.39**
Llama3.1-405B	49.90	62.38	**64.77**	24.12	39.54	**47.86**
GPT-4o	54.49	63.57	**64.57**	39.29	43.72	**49.40**
Qwen2.5-72B-Math	**45.01**	30.44	19.10	21.86	**28.50**	21.47
LearnLM-1.5-Pro	54.89	**64.07**	63.67	42.51	49.69	**51.13**

Table 1: First error step localization accuracy (in %) on VtG and PRM800K datasets. For each task, within each dataset, the **bold** value represents the highest accuracy per LLM, whereas the underlined value represents the overall highest accuracy.

4 Results and Analyses

4.1 Exact Error Step Prediction

Table 1 shows error step prediction accuracies for all model and prompt type combinations across both datasets. In general, scores are higher for VtG than PRM800K—likely due to a greater number of steps per solution on average in PRM800K (13.3) than in GSM8K (5.9). Score variation is larger in PRM800K, with smaller models like Llama3-70B and Llama3.1-70B performing comparably to larger ones on VtG. Without any reference solution (*w/o-S*), accuracy remains low, as reported in previous studies (Zeng et al., 2024; Tyen et al., 2024). *Although providing the gold solution (w-GS) increases accuracy, most models still struggle to pinpoint the exact error step* (see **RQ1**). The *corrected solution (w-Cor) improves performance over w-GS* and yields the highest accuracy across most models for both datasets (see **RQ2**). Interestingly, LearnLM-1.5-Pro shows almost no gain from intermediate corrections: its *w-GS* accuracy slightly surpasses *w-Cor* on VtG. This likely reflects the model's prior fine-tuning for mistake detection and feedback generation, which already leverages the gold solution signal, leaving little headroom for additional corrections. In sharp contrast, Qwen2.5-72B-Math—tuned for problem solving rather than critique—records the lowest accuracies overall and even drops in both *w-GS* and *w-Cor* compared to *w/o-S*, while scoring highly in problem-solving on both datasets (see Table 5).

A follow-up qualitative analysis of the predictions of Qwen2.5-72B-Math reveals that while generating the corrected solution (*w-Cor*), the model often fails to rectify the first error step in student

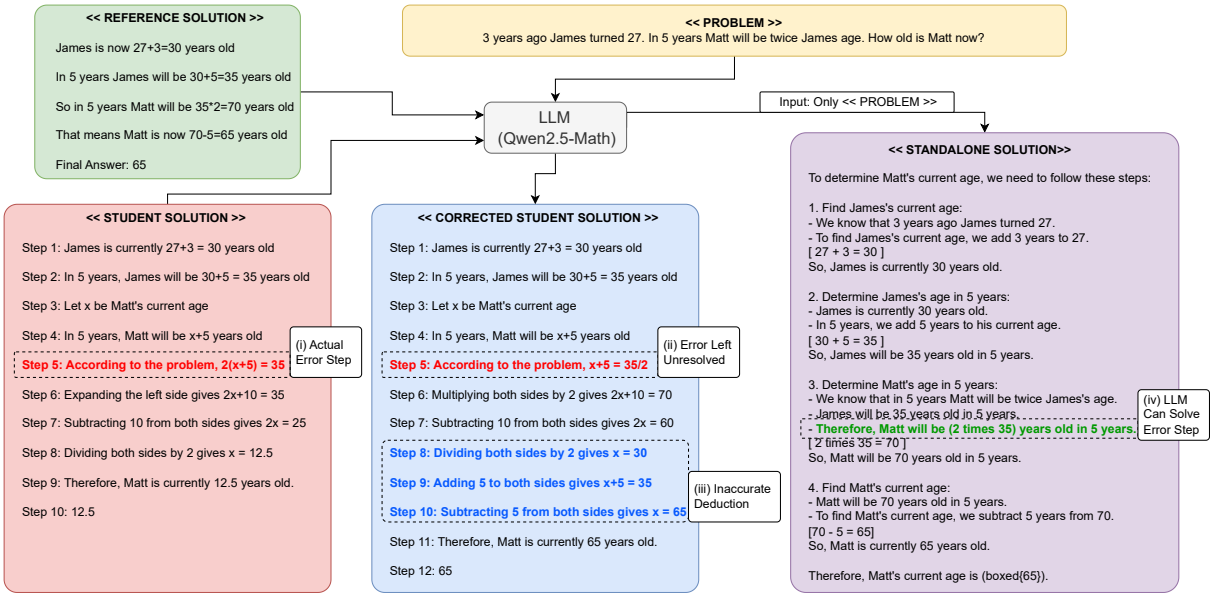

Figure 2: `Qwen2.5-72B-Math` is often unable to rectify the first error step (i) in the student's solution when generating the corrected solution (ii). Instead, additional erroneous deductions (iii) are made later in the solution to make sure that the final answer matches that of the gold (reference) solution. Note that the model can correctly solve the corresponding step in a standalone problem-solving setup (iv).

solutions and instead produces inaccurate deductions later in the solution, possibly hallucinating to ensure that the final answer matches that of the gold solution. We present one such example in Figure 2. We see that the actual error step (i) remains erroneous in the corrected solution (ii). The model later generates multiple contradictory values of x (iii). This inaccurate correction, in turn, leads to an incorrect error-step prediction. Note that `Qwen2.5-72B-Math` can correctly solve the underlying math problem by itself, including the exact step (iv), which corresponds to the student's first error step.

4.2 Feature Importance Analysis

Following prior work on interpretability for black-box models (Thakur et al., 2025; Dang et al., 2024), we train a Random Forest classifier to predict whether a student error will be correctly localized, using key features related to the problem, solution, and error. We favor a Random Forest model over a Linear Regressor based on their F1-scores as goodness-of-fit proxies (0.996 and 0.572 respectively). The feature set includes linguistic attributes of the math problem (e.g., FKGL, constituency tree depth), the complexity of the gold solution (e.g., counts and types of operations), and descriptors of the student error (e.g., error type and position). We also include a semantic alignment estimate, `Semantic Recall` (§C.1), measuring how well

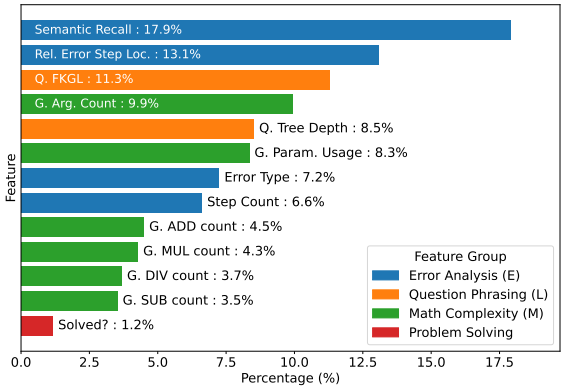

Figure 3: Relative Importance of Features Toward Correct Error Localization

the reference solution (G for *w-GS*, S' for *w-Cor*) aligns with the student's work up to the first error. See §C.4 for feature set definitions and detailed analysis description.

Figure 3 shows mean feature importances. The features related to question phrasing (orange) and math complexity (green) are some of the most informative. However, the two most informative features pertain to the error made (blue). `Semantic Recall` is the most important (17.9%), highlighting the role of alignment in successful error localization. The relative position of the error (`Rel. Error Step Loc.`, 13.1%) and error type (`Error Type`, 7.2%) also rank highly.

Interestingly, whether the LLM solved the problem correctly (`Solved?`) has low importance

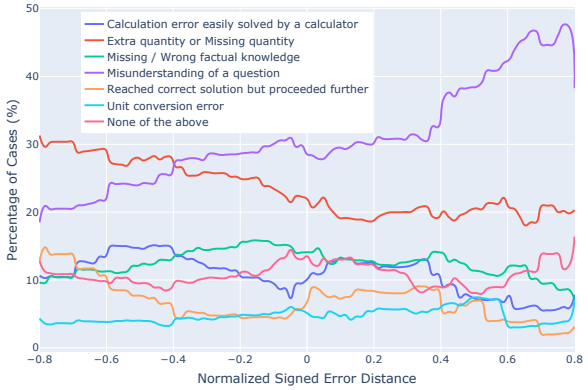

Figure 4: Distribution of ground-truth error types in VtG across models' normalized error step distance

(1.1%). A chi-squared test for independence confirms that error localization and problem-solving accuracy are weakly correlated ($p > 0.01$; $\phi < 0.2$) across models and prompts (see §C.3 for more details), suggesting that LLMs are not guaranteed to localize errors correctly even when they can solve the underlying problem.

4.3 Error Location vs. Type

Finally, we examine cases where models' predicted error steps increasingly deviate from the actual steps. We define the normalized error-step distance as the difference between the predicted and actual error steps divided by the total number of steps in the student's solution. In Figure 4, we plot the distribution of ground-truth error types from VtG against the combined normalized error-step distance across models and prompt types. A distribution shifted to the right indicates that models tend to overshoot the actual error step, while a leftward shift indicates undershooting. We observe that question-independent errors, such as calculation or unit-conversion mistakes, are uniformly distributed regardless of prediction deviation, whereas errors resulting from question misunderstanding are predicted much later than they occur. In contrast, errors involving missing or extra variables tend to be predicted a little before they occur. *This suggests that error-step prediction strategies should also account for the type of error.*

5 Conclusions

In this paper, we explored whether incorporating gold solutions enhances LLMs' ability to pinpoint errors in student math solutions from VtG and PRM800K datasets. Gold solutions improve performance compared to using only the problem and student response, though scores remain low. Replacing the gold solution with an intermediate corrected student solution further boosts performance—especially for smaller models—even though error localization still lags behind overall problem-solving accuracy. Our analysis shows that high problem-solving ability does not guarantee effective error detection, highlighting the need for targeted meta-reasoning improvements. Our feature-led analysis also shows that the alignment between the incorrect solution and the reference supports better error localization. These insights will guide our future work in enhancing LLM performance on error localization and related meta-reasoning tasks.

Limitations

This work is subject to several limitations that frame the scope of our findings. First, our experiments are confined to the math domain. While using math word problems provides a controlled setting to explore error localization, it remains unclear whether the observed challenges and benefits would generalize to other domains requiring different reasoning strategies. Second, the study depends on corrections generated by LLMs that are guided by a ground-truth solution. Although these corrected solutions were confirmed to yield the correct final answer, they may still harbor inconsistencies in their intermediate steps. Expert-annotated corrections, which could potentially offer a more reliable reference, were not employed due to the considerable resources required. Third, our evaluation uses a targeted prompting setup designed for comparability across models. Advanced prompting strategies—such as tree-of-thought prompting—have not been explored in this study, leaving open the possibility that alternative approaches might impact error localization performance. Finally, the study is limited to English-language math problems. Given that error localization performance is already challenged in English, it is plausible that the difficulties would be exacerbated in languages with less extensive data representation.

Ethical Statement

As the scope of our study is solely to evaluate LLM performance and does not involve private data or manual data creation, we do not foresee any major ethical implications of our work. However, LLMs inherently present risks. These models may generate outputs that, despite being plausible, are factually inaccurate or nonsensical. Such hallu-

cinations can lead to misguided decision-making and the propagation of biases, particularly in high-stakes contexts where accuracy is paramount. In the absence of appropriate safeguards, the broad deployment of LLMs could exacerbate these issues. Thus, it is imperative to develop mechanisms that mitigate the risks of hallucinations to ensure the responsible and effective application of these models.

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*.

John R. Anderson, C. F. Boyle, Albert T. Corbett, and R. Pelletier. 1990. Cognitive Tutors: Lessons Learned. In *Advances in Instructional Psychology*, volume 4, pages 179–202. Lawrence Erlbaum Associates, Inc.

Thomas P. Carpenter, Elizabeth Fennema, Maria L. Franke, Linda Levi, and Sandra B. Empson. 1989. Cognition and teaching: The role of content knowledge. In Jane Kilpatrick, William G. Martin, and David Schifter, editors, *Teachers' Knowledge and the Mathematics Classroom*, pages 49–84. National Council of Teachers of Mathematics.

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.

Jacob Cohen. 1988. *Statistical Power Analysis for the Behavioral Sciences*, 2 edition. Lawrence Erlbaum Associates, Hillsdale, NJ.

Nico Daheim, Jakub Macina, Manu Kapur, Iryna Gurevych, and Mrinmaya Sachan. 2024. Stepwise Verification and Remediation of Student Reasoning Errors with Large Language Model Tutors. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8386–8411, Miami, Florida, USA. Association for Computational Linguistics.

Cuong Dang, Dung D. Le, and Thai Le. 2024. A Curious Case of Searching for the Correlation between Training Data and Adversarial Robustness of Transformer Textual Models. *Preprint*, arXiv:2402.11469.

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*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal

Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The Llama 3 Herd of Models. *Preprint*, arXiv:2407.21783.

John Hattie and Helen Timperley. 2007. The Power of Feedback. *Review of Educational Research*, 77(1):81–112.

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. *Preprint*, arXiv:2103.03874.

Qinjin Jia, Jialin Cui, Ruijie Xi, Chengyuan Liu, Parvez Rashid, Ruochi Li, and Edward Gehringer. 2024. On Assessing the Faithfulness of LLM-generated Feedback on Student Assignments. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 491–499.

J. Peter Kincaid, Robert P. Fishburne, Richard L. Rogers,

and Brad S. Chissom. 1975. Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel. Technical Report Research Branch Report 8-75, Naval Air Station Memphis, Research Branch, Millington, TN.

Hang Li, Tianlong Xu, Kaiqi Yang, Yucheng Chu, Yanling Chen, Yichi Song, Qingsong Wen, and Hui Liu. 2024. Ask-Before-Detection: Identifying and Mitigating Conformity Bias in LLM-Powered Error Detector for Math Word Problem Solutions. *Preprint*, arXiv:2412.16838.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's Verify Step by Step. *Preprint*, arXiv:2305.20050.

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

Jakub Macina, Nico Daheim, Sankalan Chowdhury, Tanmay Sinha, Manu Kapur, Iryna Gurevych, and Mrinmaya Sachan. 2023. MathDial: A Dialogue Tutoring Dataset with Rich Pedagogical Properties Grounded in Math Reasoning Problems. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5602–5621, Singapore. Association for Computational Linguistics.

Francisco Niño-Rojas, Diana Lancheros-Cuesta, Martha Tatiana Pamela Jiménez-Valderrama, Gelys Mestre, and Sergio Gómez. 2024. Systematic Review: Trends in Intelligent Tutoring Systems in Mathematics Teaching and Learning. *International Journal of Education in Mathematics, Science and Technology*, 12(1):203–229.

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, Mi-

nal 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.

Louis M. Rea and Richard A. Parker. 1992. *Designing and Conducting Survey Research: A Comprehensive Guide*. Jossey–Bass, San Francisco, CA.

Sidney Siegel. 1956. *Nonparametric Statistics for the Behavioral Sciences*. McGraw–Hill, Singapore.

Kv Aditya Srivatsa and Ekaterina Kochmar. 2024. What Makes Math Word Problems Challenging for LLMs? In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1138–1148, Mexico City, Mexico. Association for Computational Linguistics.

John Sweller, Jeroen JG van Merriënboer, and Fred Paas. 1998. Cognitive architecture and instructional design. *Educational Psychology Review*, 10(3):251–296.

LearnLM Team, Abhinit Modi, Aditya Srikanth Veerubhotla, Aliya Rysbek, Andrea Huber, Brett Wiltshire, Brian Veprek, Daniel Gillick, Daniel Kasenberg, Derek Ahmed, Irina Jurenka, James Cohan, Jennifer She, Julia Wilkowski, Kaiz Alarakyia, Kevin R. McKee, Lisa Wang, Markus Kunesch, Mike Schaekermann, Miruna Pîslar, Nikhil Joshi, Parsa Mahmoudieh, Paul Jhun, Sara Wiltberger, Shakir Mohamed, Shashank Agarwal, Shubham Milind Phal, Sun Jae Lee, Theofilos Strinopoulos, Wei-Jen Ko, Amy Wang, Ankit Anand, Avishkar Bhoopchand, Dan Wild, Divya Pandya, Filip Bar, Garth Graham, Holger Winnemoeller, Mahvish Nagda, Prateek Kolhar, Renee Schneider, Shaojian Zhu, Stephanie Chan, Steve Yadlowsky, Viknesh Sounderajah, and Yannis Assael. 2024. LearnLM: Improving Gemini for Learning. *Preprint*, arXiv:2412.16429.

Nandan Thakur, Suleman Kazi, Ge Luo, Jimmy Lin, and Amin Ahmad. 2025. MIRAGE-Bench: Automatic Multilingual Benchmark Arena for Retrieval-Augmented Generation Systems. 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 274–298, Albuquerque, New Mexico. Association for Computational Linguistics.

Gladys Tyen, Hassan Mansoor, Victor Carbune, Peter Chen, and Tony Mak. 2024. LLMs cannot find reasoning errors, but can correct them given the error location. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13894–13908, Bangkok, Thailand. Association for Computational Linguistics.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. 2024. Qwen2.5-Math Technical Report: Toward Mathematical Expert Model via Self-Improvement. *Preprint*, arXiv:2409.12122.

Zhongshen Zeng, Pengguang Chen, Shu Liu, Haiyun Jiang, and Jiaya Jia. 2024. MR-GSM8K: A Meta-Reasoning Benchmark for Large Language Model Evaluation. *Preprint*, arXiv:2312.17080.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. *Preprint*, arXiv:1904.09675.

A Dataset Details

This section provides further details about the datasets used in our experiments. Tables 2 and 3 provide the distribution statistics of various aspects for `VtG` and `PRM800K` respectively, including the number of steps in the student and gold solutions and the location of error steps. Table 4 lists the error types annotated in `VtG` and their corresponding share of cases in the dataset. There is no associated human labeling for error type in `PRM800K`. Both `VtG` (CC BY 4.0 License) and `PRM800K` (MIT License) are publicly accessible datasets.

Dimension	Min	Max	Median	μ	σ
Gold Solution Word Length	6	125	49	50.51	20.13
Student Solution Word Length	48	109	73	73.62	10.97
Gold Solution Step Length	3	5	4	4.27	0.73
Student Solution Step Length	3	15	5	5.92	1.84
First Error Step Index	1	9	3	2.77	1.43

Table 2: Key distributional statistics for VtG

Dimension	Min	Max	Median	μ	σ
Gold Solution Word Length	1	441	72	87.75	67.82
Student Solution Word Length	6	1470	209	235.86	123.57
Gold Solution Step Length	1	62	3	5.72	7.02
Student Solution Step Length	2	52	12	13.31	6.52
First Error Step Index	1	34	5	5.81	4.29

Table 3: Key distributional statistics for PRM800K

Error Type	Percent of Cases (%)
Calculation error easily solved by a calculator	12.77
Extra quantity or Missing quantity	23.95
Missing / Wrong factual knowledge	13.97
Misunderstanding of a question	28.64
Reached correct solution but proceeded further	6.99
Unit conversion error	4.99
None of the above	8.68

Table 4: Annotated error types for student solutions in VtG with their respective percentage of each case

B Querying Details

B.1 Prompts

This section presents the exact prompts used in our experiments. We begin with the problem-solving prompts used to collect the solutions and final answers to the underlying math questions from VtG and PRM800K in Figure 5. The initial prompt is used to generate verbose solutions to the questions, followed by a follow-up prompt, where we append the model output with a concluding phrase (i.e., `Therefore, the final answer is:`) to get the model to specify the final numerical or expression-based answer clearly. We find that this method works best to extract the final answer without additional pattern matching. Next, we show the prompts to predict the exact error-step in the three settings, i.e., without any reference solution (*w/o-S*), with the gold solution (*w-GS*), and with the corrected student solution (*w-Cor*) in Figures 6, 7, and 8 respectively. Finally, we show the prompt used to generate the corrected form of the student solutions in Figure 9.

B.2 Querying Setup

This section describes the querying setup used for our experiments. Table 7 shows the exact model versions used. All models were queried with the `temperature` set to 0, `top_p` to 0.95, and `max_tokens` to 2048. All `Llama` models were queried using the Google Cloud (Vertex) API and `GPT-4o` queries were made using the OpenAI API.

B.3 Problem-Solving Performance

We define LLMs' problem-solving performance as their average accuracy on the math word problems from the test sets of VtG and PRM800K. Each model is prompted with the math word problem using the prompt templates shown in Figure 5 and described in §B.1.

Model	VtG	PRM800K
Llama3-70B	81.04	48.15
Llama3.1-70B	88.82	62.88
Llama3.1-405B	92.22	69.76
GPT-4o	77.45	76.22
Qwen2.5-72B-Math	83.13	87.34
LearnLM-1.5-Pro	83.93	85.36

Table 5: Mean problem-solving accuracy (%) on the underlying math problems from VtG and PRM800K

C Additional Analyses & Details

C.1 Alignment with Student Solution

We discuss the importance of aligning the ground truth and student solutions for effective comparison and error localization in Section 1. Specifically, we generated intermediate corrected versions of the student solution to serve as ground truth instead of the dataset-provided gold solutions. Ideally, a ground-truth solution should match the student solution up to the first error step, after which divergence is expected. Thus, we measure the semantic overlap between the ground truth and student solutions (truncated before the first error) using

LLM	Open Source?	Parameter Count	Fine-Tuned?
LLaMA3-70B (Dubey et al., 2024)	✓	70B	✗
LLaMA3.1-70B (Dubey et al., 2024)	✓	70B	✗
LLaMA3.1-405B (Dubey et al., 2024)	✓	405B	✗
GPT-4o (OpenAI et al., 2024)	✗	–	✗
Qwen2.5-72B-Math (Yang et al., 2024)	✓	72B	✓
LearnLM-1.5-Pro (Team et al., 2024)	✗	–	✓

Table 6: The diverse set of LLMs included in this study

Model	Version
Llama3-70B	meta-llama/Meta-Llama-3-70B-Instruct
Llama3.1-70B	meta-llama/Llama-3.1-70B
Llama3.1-405B	meta-llama/Llama-3.1-405B
GPT-4o	gpt-4o-2024-08-06
Qwen2.5-72B-Math	Qwen/Qwen2.5-Math-72B-Instruct
LearnLM-1.5-Pro	learnlm-1.5-pro-experimental

Table 7: Model versions for the LLMs used in our experiments

Problem-Solving Prompts

(A) Initial Prompt

You are experienced at solving math word problems. Solve the following problem to the best of your ability in a stepwise manner.
Clearly specify your final answer at the end of your solution in a new line.

Problem: <<PROBLEM>>

(B) Follow-Up Prompt

You are experienced at solving math word problems. Solve the following problem to the best of your ability in a stepwise manner.
Clearly specify your final answer at the end of your solution in a new line.

Problem: <<PROBLEM>>

<<RESPONSE_TO_INITIAL_PROMPT>>

Therefore, the final answer is:

Figure 5: Problem-solving prompts

BERTScore (Zhang et al., 2020) recall (see Table 8). Maximizing recall ensures that the ground truth closely follows the student's approach. We also use this quantity (Semantic Recall) as a feature for further analysis in §4.2. *The results show that corrected solutions from all models yield much higher recall than the corresponding gold solutions for both datasets, indicating superior semantic and stylistic alignment with the student solutions.*

C.2 Manual Verification of Generated Corrections

We test whether an intermediate generation of a corrected student solution using the gold solution and the student solution serves as a better reference for error localization than the gold solution itself.

The annotation for each correction involves two questions:

- **Correctness:** Is the LLM-generated correction factually and mathematically sound at each step and does it arrive at the correct answer? (Yes/No)
- **Stylistic Similarity:** Is the LLM-generated correction, stylistically and in approach, similar to the student's solution up to the first error step? (Yes/No)

Without Solution (w/o-S) Prompt

You are an experienced math teacher. Your goal is to identify the step of the first mistake in the Student's Solution to a Problem.

Problem: Natalia sold clips to 48 of her friends in April, and then she sold half as many clips in May. How many clips did Natalia sell altogether in April and May?

Student Solution:
Step 1 - Natalia sold 48 clips in April.
Step 2 - She sold 48*2 = 96 clips in May.
Step 3 - She sold 48+96 = 144 clips in April and May together.

Q: Write only the step number with the first error.
A: 2

Problem: Natalia sold clips to 48 of her friends in April, and then she sold half as many clips in May. How many clips did Natalia sell altogether in April and May?

Student Solution:
Step 1 - She sold 48/2 = 16 clips in May.
Step 2 - Natalia sold 48+16 = 64 clips in April and May together.

Q: Write only the step number with the first error.
A: 1

Problem: <<PROBLEM>>

Student Solution:
<<STUDENT_STEPS>>

Q: Write only the step number with the first error.
A:

Figure 6: Prompt without solution (*w/o-S*)

Solution Type	Model	VtG	PRM800K
Gold	–	89.52	85.12
Corrected	Llama3-70B	94.77	94.46
	Llama3.1-405B	96.04	95.34
	Llama3.1-70B	96.18	95.88
	GPT-4o	95.86	93.79
	Qwen2.5-72B-Math	95.03	87.66
	LearnLM-1.5-Pro	94.98	92.98

Table 8: BERTScore recall between ground-truth solutions (gold or corrected) and student solutions. Recall values for corrected solutions from different LLMs have been recorded separately.

Table 9 shows the average percentage values for the two questions for each LLM. The annotation set spans 90 samples (15 samples per LLM), with 30 randomly selected samples (of the 90) to build the agreement subset. We conducted a two-person annotation where both annotators hold at least a

With Gold Solution (w-GS) Prompt

You are an experienced math teacher. Your goal is to identify the step of the first mistake in the Student's Solution to a Problem.

Problem: Natalia sold clips to 48 of her friends in April, and then she sold half as many clips in May. How many clips did Natalia sell altogether in April and May?

Expected Answer:
Step 1 - Natalia sold 48/2 = 24 clips in May.
Step 2 - Natalia sold 48+24 = 72 clips altogether in April and May.

Student Solution:
Step 1 - Natalia sold 48 clips in April.
Step 2 - She sold 48*2 = 96 clips in May.
Step 3 - She sold 48+96 = 144 clips in April and May together.

Q: Write only the step number with the first error.
A: 2

Problem: Natalia sold clips to 48 of her friends in April, and then she sold half as many clips in May. How many clips did Natalia sell altogether in April and May?

Expected Answer:
Step 1 - Natalia sold 48/2 = 24 clips in May.
Step 2 - Natalia sold 48+24 = 72 clips altogether in April and May.

Student Solution:
Step 1 - She sold 48/2 = 16 clips in May.
Step 2 - Natalia sold 48+16 = 64 clips in April and May together.

Q: Write only the step number with the first error.
A: 1

Problem: Ignatius owns 4 bicycles. A friend of his owns different types of cycles, which have three times are many tires as Ignatius's bikes have. He has one unicycle, a tricycle, and the rest are bikes. How many bicycles does the friend own?

Expected Answer:
<<GOLD_STEPS>>

Student Solution:
<<STUDENT_STEPS>>

Q: Write only the step number with the first error.
A:

Figure 7: Prompt with gold solution (*w-GS*)

With Corrected Student Solution (w-Cor) Prompt

You are an experienced math teacher. Your goal is to identify the step of the first mistake in the Student's Solution to a Problem.

Problem: Natalia sold clips to 48 of her friends in April, and then she sold half as many clips in May. How many clips did Natalia sell altogether in April and May?

Expected Answer:
Step 1 - Natalia sold 48/2 = 24 clips in May.
Step 2 - Natalia sold 48+24 = 72 clips altogether in April and May.

Student Solution:
Step 1 - Natalia sold 48 clips in April.
Step 2 - She sold 48*2 = 96 clips in May.
Step 3 - She sold 48+96 = 144 clips in April and May together.

Q: Write only the step number with the first error.
A: 2

Problem: Natalia sold clips to 48 of her friends in April, and then she sold half as many clips in May. How many clips did Natalia sell altogether in April and May?

Expected Answer:
Step 1 - Natalia sold 48/2 = 24 clips in May.
Step 2 - Natalia sold 48+24 = 72 clips altogether in April and May.

Student Solution:
Step 1 - She sold 48/2 = 16 clips in May.
Step 2 - Natalia sold 48+16 = 64 clips in April and May together.

Q: Write only the step number with the first error.
A: 1

Problem: Ignatius owns 4 bicycles. A friend of his owns different types of cycles, which have three times are many tires as Ignatius's bikes have. He has one unicycle, a tricycle, and the rest are bikes. How many bicycles does the friend own?

Expected Answer:
<<CORRECTED_STUDENT_STEPS>>

Student Solution:
<<STUDENT_STEPS>>

Q: Write only the step number with the first error.
A:

Figure 8: Prompt with corrected student solution (*w-Cor*)

masters degree in a STEM field. The first annotator annotates all 90 samples and the second annotator annotates just the agreement set. With this, we estimate that the inter-annotator agreement is κ = 0.82 for correctness and 0.85 for stylistic similarity.

C.3 Chi-Square Test

In §4.2, we present the correlation between LLMs' problem-solving performance and error localization performance. As both variables are binary categorical variables, the appropriate method to determine their relation is to construct a 2×2 contingency table and perform a Chi-Square test to check for a statistically significant association between them (Siegel, 1956). Table 10 presents the results for the test (χ^2 statistic, p-value, and ϕ-coefficient) across models, datasets, and prompt types. We interpret the correlation in each case by the corresponding p-values and ϕ-coefficients. A

Corrected Solution Generation Prompt

Here's a problem: <<PROBLEM>>

Here's the problem's correct reference solution:
<<GOLD_SOLUTION>>

Here's a stepwise candidate solution to the same problem:

<<STUDENT_STEPS>>

Based on the problem and the reference solution, correct and rewrite the candidate solution.
Change only the portions that are incorrect and need edits.

Figure 9: Corrected solution generation prompt

Model	Correctness	Stylistic Similarity
Llama3-70B	93.75	91.02
Llama3.1-70B	96.67	93.33
Llama3.1-405B	94.00	88.40
GPT-4o	93.33	89.67
Qwen2.5-72B-Math	69.59	63.33
LearnLM-1.5-Pro	95.71	87.43

Table 9: Average percentage values of correctness and stylistic similarity based on manual annotation on LLM-generated corrected student solutions

Model	Prompt Type	VtG			PRM800K		
		χ^2	p-value	ϕ	χ^2	p-value	ϕ
Llama3-70B	*w/o-S*	4.30	0.038	0.068	3.55	0.059	0.043
	w-GS	8.00	0.005	0.092	1.34	0.247	0.027
	w-Cor	0.00	1.000	0.002	4.30	0.038	0.047
Llama3.1-70B	*w/o-S*	0.49	0.482	-0.025	10.09	0.001	0.071
	w-GS	0.24	0.620	0.019	16.49	0.000	0.090
	w-Cor	0.08	0.774	0.012	29.70	0.000	0.121
Llama3.1-405B	*w/o-S*	0.08	0.768	-0.013	3.59	0.058	0.043
	w-GS	3.34	0.067	-0.062	12.19	0.000	0.078
	w-Cor	0.05	0.824	-0.011	27.71	0.000	0.117
GPT-4o	*w/o-S*	2.14	0.143	0.049	7.28	0.007	0.060
	w-GS	0.94	0.332	0.033	4.51	0.034	0.048
	w-Cor	1.77	0.183	0.045	6.92	0.009	0.059
Qwen2.5-72B-Math	*w/o-S*	17.356	0.000	0.134	3.077	0.079	0.040
	w-GS	0.030	0.863	0.008	9.840	0.002	0.070
	w-Cor	4.880	0.027	-0.073	5.042	0.025	0.051
LearnLM-1.5-Pro	*w/o-S*	13.021	0.000	0.117	3.426	0.064	0.042
	w-GS	5.149	0.023	0.075	2.056	0.152	0.033
	w-Cor	3.208	0.073	0.059	2.198	0.138	0.034

Table 10: Chi-Square test and ϕ-coefficient statistics between models' problem-solving performance and error localization performance on VtG and PRM800K test sets. A high p-value (>0.01) or a low ϕ-coefficient (<0.2) (i.e., weak effect size) are both indicative of a poor correlation (Cohen, 1988; Rea and Parker, 1992).

high p-value (>0.01) is indicative of poor statistical significance and a low ϕ-coefficient (<0.2) is indicative of a low effect size (Cohen (1988); Rea and Parker (1992) regard coefficient values <0.2 as weak).

We observe that no setting in Table 10 yields a strong correlation. *This means that, within an LLM's set of responses, solving the problem correctly does not strongly predict model's ability to pinpoint an error.* E.g., even though Qwen2.5-72B-Math outperforms most other models in problem solving across datasets (see Table 5), its error localization performance is the poorest. This further motivates the need for LLMs tuned for better error diagnostic capabilities among other meta-reasoning abilities.

C.4 Feature Importance Analysis Details

We fit a Random Forest classifier to predict whether an LLM will be able to correctly predict the first error step in a given student solution using key features to determine their relative importance in determining the LLM's performance. In this section, we describe the feature set that we used and the process of fitting the model and extracting the feature importance scores.

Feature Set We use a feature set capturing the phrasing of the math problem (L), the mathematical complexity of the underlying gold solution (M), and details about the error made by the student (E). We borrow the L and M type features and their exact extraction implementation from Srivatsa and Kochmar (2024). The feature set is as follows:

- Q. Word Length (L): The number of space-separated words in the question text.
- Q. Arg. Count (L): Number of distinct numerical quantities in the question text. E.g., "**20** boxes" or "**1.5** hours later".
- Q. FKGL (L): The FKGL readability grade (Kincaid et al., 1975) of the question text.
- Q. Tree Depth (L): The average depth of the constituency tree for the sentences in the question text.
- Q. NP Count (L): The number of unique noun phrases in the question text.
- G. Arg. Count (M): The number of distinct numerical quantities in the gold solution. These may include the arguments imported from the question text and intermediate arguments calculated in the solution steps.
- G. ADD/ SUB/ MUL/ DIV Count (M): The number of instances of each of the arithmetic operators used in the gold solution.
- G. Op. Unique Count (M): The number of unique arithmetic operators used in the gold solution.
- G. Op. Diversity (M): Ratio of G. Op. Unique Count and G. ADD/ SUB/ MUL/ DIV Count.
- G. Param. Usage (M): Ratio of G. Arg. Count and Q. Arg. Count. This serves as a measure of the proportion of input arguments that are actually relevant to solving the problem. A lower ratio means a greater number of distractors.

- `G. World Knowledge (M)`: The number of arguments in the gold solution that are neither input arguments from the question text nor intermediate variables. Such arguments are mainly real world quantities required to solve the problem but not explicitly provided by the question.

- `Step Count (E)`: The total number of steps in the incorrect student solution.

- `Rel. Error Step Loc. (E)`: The relative position of the first error step in the incorrect student solution. This is defined as the ratio of the first error's step index and the total number of steps in the student solution.

- `Error Type (E)`: One of the 7 error types as shown in Table 4.

- `Semantic Recall (E)`: An estimate of semantic alignment between the steps of the student solution and the reference solution (gold solution for *w-GS* and corrected student solution for *w-Cor*) up to the first error step in the student solution. This is defined as the BERTScore (Zhang et al., 2020) recall between the two solutions for the solution texts before the first erroneous step in the student solution. See more details in C.1.

Model Fitting We use the Random Forest (RF) implementation from `Scikit-Learn`. Before training the RF model, we prune the feature data to only retain features with an absolute Spearman correlation value <0.4. This removes redundant features, which would otherwise make interpreting feature importance scores difficult. The RF model is trained with 200 estimators and each model and prompt setting is trained 10 times with varying initialization seeds.

Feature Importance Calculation Trained RF models return the normalized (sum = 1.0) Gini importance values for each input feature. The overall importance value (Λ_i %) for a feature i across RF models is aggregated as weighted mean of each model's feature importance for feature i (λ_{ij}), by the corresponding goodness of fit, i.e., accuracy (a_j) (see Eq. 1).

$$\Lambda_i = 100 \times \frac{\sum_j a_j \cdot \lambda_{ij}}{\sum_j a_j} \quad (1)$$

Model	Prompt Type	VtG		PRM800K	
		±1	±2	±1	±2
Random	–	31.89	55.5	17.53	32.42
Llama3-70B	*w-GS*	56.47	82.64	28.17	46.78
	w-GS	56.47	82.64	28.17	46.78
	w-Cor	**59.20**	**84.48**	**31.25**	**49.36**
Llama3.1-70B	*w/o-S*	57.28	**82.82**	24.52	41.91
	w-GS	53.70	78.84	**27.5**	**45.38**
	w-Cor	**58.27**	81.30	25.97	41.70
Llama3.1-405B	*w/o-S*	47.61	77.68	27.16	45.49
	w-GS	54.86	82.00	26.72	**48.03**
	w-Cor	**56.01**	**83.58**	**30.79**	47.85
GPT-4o	*w/o-S*	56.21	83.16	25.36	43.31
	w-GS	53.12	78.30	**28.92**	**49.18**
	w-Cor	**60.86**	**84.57**	23.48	37.52
Qwen2.5-72B-Math	w/o-S	**48.36**	73.24	37.78	47.54
	w-GS	46.35	**74.92**	**48.97**	**58.51**
	w-Cor	32.68	65.08	43.69	56.37
LearnLM-1.5-Pro	w/o-S	54.36	80.77	61.70	72.22
	w-GS	**58.14**	**83.72**	66.11	75.15
	w-Cor	57.26	83.24	**66.94**	**75.66**

Table 11: Percentage of incorrect first error-step predictions where the prediction lies within ±1 and ±2 steps of the actual first error step. *Bold* values denote the greatest percentage value among the three prompt settings for a given model and dataset.

C.5 How far off are LLMs?

We aim to assess how close the predicted error step is to the true error step when the prediction is incorrect. To do so, we compute the percentage of incorrect predictions that fall within ±1 and ±2 steps of the actual first error step, as detailed in Table 11 and further distributions in §C.6. Our analysis reveals that for `VtG` (median step count: 5), between 45% and 60% of the incorrect predictions are within ±1 step, whereas for `PRM800K` (median step count: 12), approximately 25% fall within ±1 step and nearly 50% within ±2 steps. Additionally, among the three prompt settings, *w-Cor* most consistently achieves the highest number of predictions within both windows and performance across models is similar, with `Llama3-70B` matching or surpassing both `GPT-4o` and `Llama3.1-405B`. *These results suggest that while models often miss the exact first error step, their predictions remain close, motivating the development of fine-grained policies to precisely pinpoint error steps in future work.*

C.6 Error-Step Distance Distribution

In §C.5, we report the proportion of incorrectly predicted error steps that lie within ±1 and ±2 steps of the actual error step. In Figures 10 and 11, we present a more detailed distribution of incorrectly predicted error steps by their relative distance from the actual error step for both datasets.

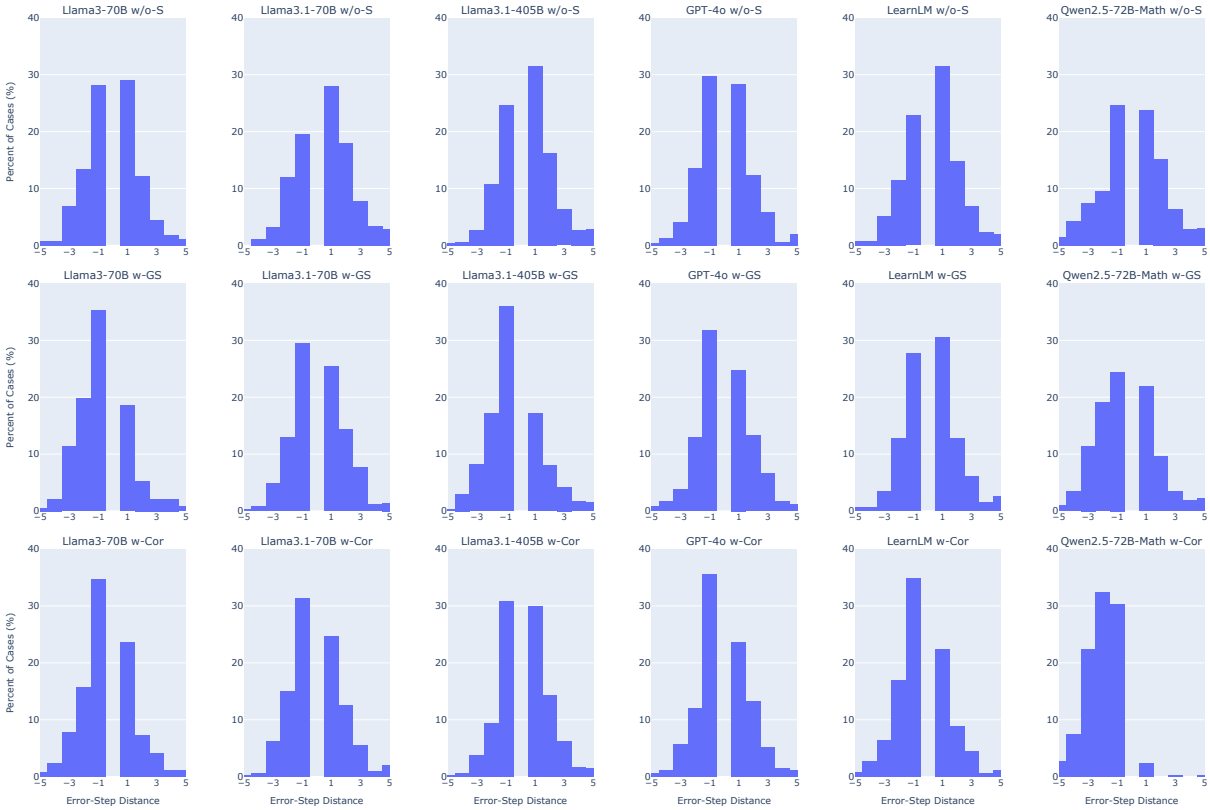

Figure 10: Error-step distance distributions for VtG

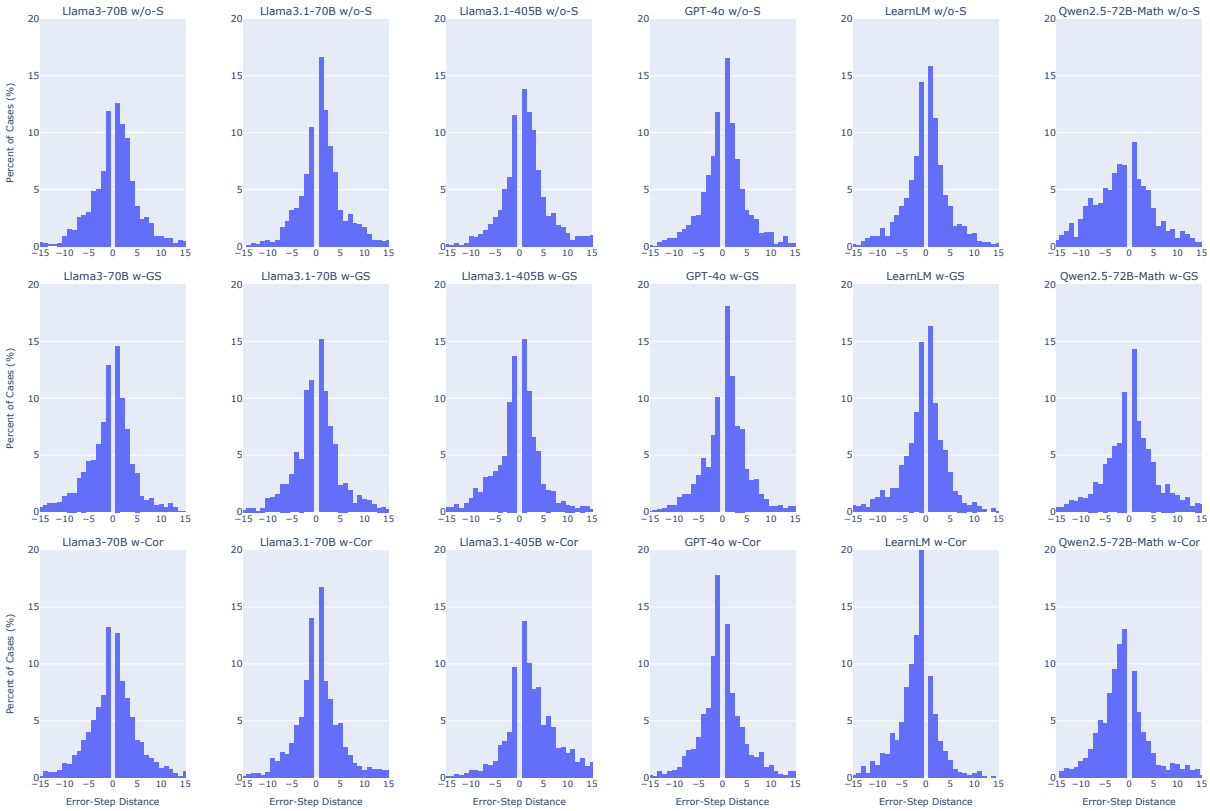

Figure 11: Error-step distance distributions for PRM800K