JOURNAL OF
EMERGING INVESTIGATORS

Article

Training and comparison of fine-tuned small language models as math tutoring assistants

Yash Maheshwari[1], Rinky Gupta[2]
[1] Mountain View High School, Mountain View, California
[2] Tech Spark, Los Altos, California

SUMMARY

Recent breakthroughs in artificial intelligence (AI) have revolutionized not only computer science but also automation, enabling machines to perform tasks only humans could once do. AI tutoring assistants can support math education by providing step-by-step guidance, but their effectiveness depends substantially on model size and training duration. However, large language models require a very significant number of resources, making them impractical for schools. This study evaluates Llama 3.2 configurations with 1 billion (1B), 3 billion (3B), and 8 billion (8B) parameters, trained for 1, 3, or 5 epochs, as tutoring tools for elementary, middle, and high school math. We hypothesized that increasing a Llama 3.2 model's parameter count would improve step-accuracy more effectively than increasing the number of epochs during fine-tuning. Using Google Colab and Unsloth's open-source code, we fine-tuned the models on 538 math queries. We assessed performance on 126 test prompts based on their ability to avoid revealing answers, provide accurate step-by-step hints and guidance, and achieve high human-rated quality. The 8B model at 5 epochs performed best (93.65% accuracy, 0.946 rating), while the 1B model at 1 epoch lagged in performance (55.56% accuracy, 0.700 rating). These results suggest that small, fine-tuned language models can provide a practical balance between educational quality and efficiency while mitigating cost, making them viable as AI tutors for classrooms.

INTRODUCTION

Math is a core part of education, yet many students continue to struggle with understanding it. In the U.S., 39% of middle schoolers score below the basic level on the National Assessment of Educational Progress (NAEP), and just 28% reach proficiency (1). These numbers point to a long-standing issue in the American education system. Even though the country leads in technology and innovation, students are often not gaining the fundamental skills needed for future academic and career success, especially in fields like science, technology, and engineering, where a strong foundation in math is essential (2).

One potential way to address this gap in student math achievement is through AI-powered math tutoring assistants that provide personalized, step-by-step guidance to students. Teachers are often severely outnumbered by students and unable to provide immediate, individualized guidance to every student who needs help. Unlike traditional math tools such as calculators, modern AI systems can walk students through problems, explain each step, and offer personalized help at any time, leading to the rise of AI tutoring assistants that can help students outside of the classroom. For example, when given a query like "Solve for x: $2x + 5 = 13$", an effective tutoring model would respond with a clear response, without directly providing the answer: "Let's start by isolating the x term. What operation would help you remove the +5 from the left side?"

Despite growing interest in AI tutoring tools, most current systems rely on very large and resource-intensive models that are unaffordable for many schools and damaging to the environment (3). Khan Academy's Khanmigo, initially powered by GPT-4, is designed to guide students through problems interactively rather than just providing solutions (4). Research on intelligent tutoring systems (ITS) has shown that step-by-step support and feedback can improve learning outcomes (5). A meta-analysis of 50 studies found that ITS increased test scores by an average of 0.66 standard deviations, equivalent to moving a student from the 50th to the 75th percentile (5). However, many current AI tutoring systems rely on large commercial models with hundreds of billions, sometimes trillions, of parameters, such as GPT-5, that require significant computational resources, making them impractical for schools with limited budgets and harmful for the environment (3). This access problem is relevant in schools: across OECD countries, nearly one-third of 15-year-old students were in schools where principals reported that internet speed or bandwidth was insufficient for instruction (6). These findings suggest that many schools still face infrastructure limits, which makes access to high-compute AI tools uneven. In contrast, smaller models of under 10 billion parameters require fewer resources, making them more affordable and less harmful for the environment (3). Previous research suggests that both model size and training time can impact how accurately a model responds (7). However, it remains unclear exactly how much is required for effective educational support and the relative effect both have on model performance (7). Large language models such as GPT-5 perform well on a wide range of tasks and generally produce higher-quality responses than smaller models, further supporting the potential of parameter scaling for tutoring efficacy (8). Thus, exploring whether smaller, more affordable models could achieve similar educational impacts is important.

This study explores whether smaller, open-source model configurations, when properly fine-tuned, can achieve math tutoring comparable to large commercial models. We evaluated nine configurations of Meta's open-source language model Llama 3.2, using 1 billion (1B), 3 billion (3B), and 8 billion (8B) parameter versions trained for 1, 3, and 5

https://doi.org/10.59720/25-174

epochs, to see how well they could assist with math instruction. These configurations were fine-tuned on math questions and explanations, and their performance was tested across several training durations (epochs). We hypothesized that increasing parameter count across Llama 3.2 configurations would improve step accuracy more strongly than adding further fine-tuning epochs. Prior research done with scaling-laws shows that accuracy rises almost log-linearly with parameter count when compute is fixed, while additional training steps deliver diminishing returns, indicating the increased effectiveness of larger model size compared to training (9). Since these smaller models require fewer resources to run, they could be a more practical option for schools looking to improve math education with limited technology. Overall, our results showed that larger Llama 3.2 configurations performed notably better than smaller ones, and parameter count had a stronger effect on performance than training duration.

RESULTS

The goal of this study was to determine if smaller, fine-tuned language model configurations could perform as effective math tutors and, specifically, whether the parameter count or training duration affects performance more. We hypothesized that increasing parameter count would improve accuracy more than increasing training epochs. To test this, we evaluated nine Llama 3.2 configurations, using 1B, 3B, and 8B parameter versions trained for 1, 3, and 5 epochs each, using Unsloth's open-source code and an NVIDIA T4 GPU (10, 11, 12). We tested the nine configurations on 126 prompts, measuring three things: if the model reveals the final answer (Gave Answer; lower percentages indicate better pedagogical behavior), whether correct first-step guidance was given (Accurate Step), and a holistic rating based on clarity and helpfulness (Holistic Rating), a score from 0 to 1 assigned by the first author representing the clarity, helpfulness, and educational value of the response, where 0 means an unhelpful or unclear response and 1 indicates a clear and helpful response. The results of our experiments reveal clear and consistent relationships between model size, training duration, and performance. We first tested how training duration (epochs) affected model performance across different parameter sizes.

Increasing the number of fine-tuning epochs improved step accuracy and holistic rating across the different parameter sizes, with the best results for the largest models with the most training. The 1B model's step accuracy rose from 55.6% at one epoch to 64.3% at three epochs, while its holistic rating increased from 0.700 to 0.778 (**Table 1**). However, model performance dipped after five epochs, dropping to 60.3% accuracy with a rating decline to 0.752, indicating a limit to the model's effectiveness. Conversely, the 3B model showed steady growth from 67.5% accuracy and a 0.694 rating at one epoch to 80.2% and 0.849 at three epochs, and 89.7% and 0.930 at five epochs (**Table 1**). Similarly, the 8B model also had notable improvement in model performance as the epochs increased, improving from 80.2% accuracy and a 0.865 rating at one epoch to 88.1% and 0.913 at three epochs, and 93.7% and 0.946 at five epochs (**Table 1**). These patterns demonstrate that while the 1B configuration plateaued early, the 3B and 8B models continue to show better performance from extended training (**Figure 1A**). The standard error bars confirm this, with the 3B model having a rise in step accuracy from 0.675 ± 0.042 standard error of the mean (SEM) at 1 epoch to 0.897 ± 0.027 SEM at 5 epochs (**Table 1**). This is a 0.222 increase, which is more than 2x the combined SEM, indicating a substantial epoch effect for this model. Even so, this is still substantially less than the increase in model performance that occurred as the parameters were

Model (Llama 3.2)	Accurate Step (%)	Accurate Step SEM	Holistic Rating (0-1)	Holistic Rating SEM	Gave Answer (%)	Gave Answer SEM
1B - 1 Epoch	55.6	0.044	0.700	0.025	3.17	0.016
1B - 3 Epoch	64.3	0.043	0.778	0.026	1.59	0.011
1B - 5 Epoch	60.3	0.044	0.752	0.026	1.59	0.011
3B - 1 Epoch	67.5	0.042	0.694	0.022	4.76	0.019
3B - 3 Epoch	80.2	0.036	0.849	0.022	1.59	0.011
3B - 5 Epoch	89.7	0.027	0.930	0.017	1.59	0.011
8B - 1 Epoch	80.2	0.036	0.865	0.022	1.59	0.011
8B - 3 Epoch	88.1	0.029	0.913	0.018	3.97	0.017
8B - 5 Epoch	93.7	0.022	0.946	0.015	3.17	0.016

Table 1: Performance metrics and standard error of the mean (SEM) for Llama 3.2 models across parameter sizes and training epochs. Metrics include Accurate Step: the proportion of responses containing a correctly executed first step; Holistic Rating: an overall quality score (0–1 scale) assessing clarity, helpfulness, and educational appropriateness; and Gave Answer: the proportion of instances where the model directly revealed the final answer rather than guiding through steps. $SEM = \frac{SD}{\sqrt{N}}$, where SD is the standard deviation and N=126 is the number of test prompts per model. Models with larger parameter sizes and more training epochs generally demonstrated higher step accuracy and holistic ratings.

https://doi.org/10.59720/25-174

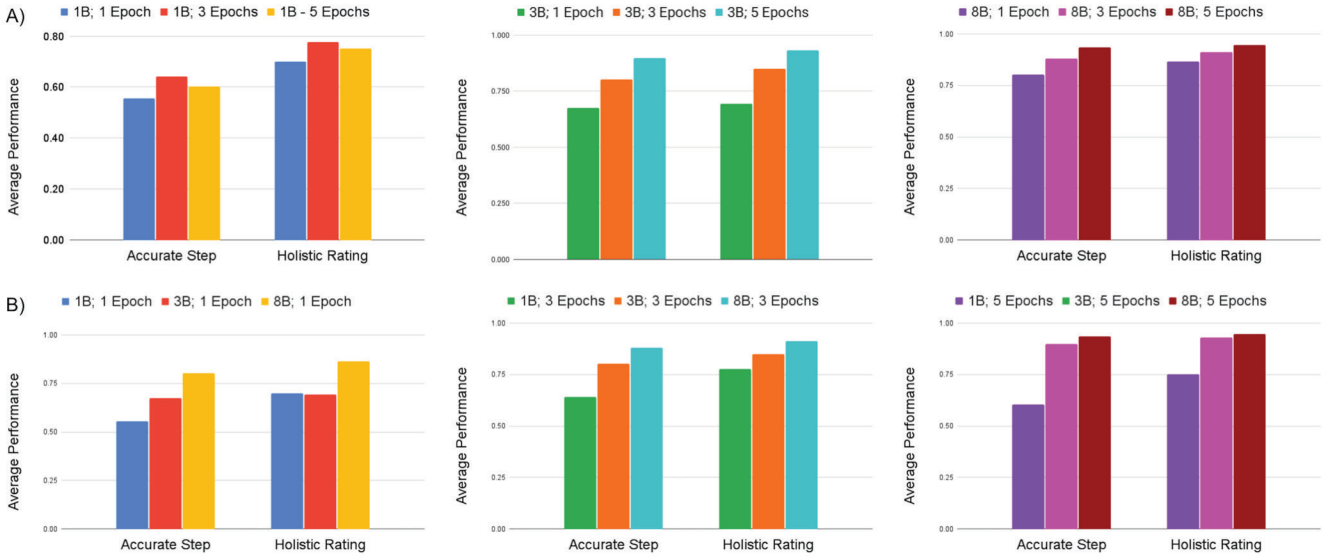


Figure 1: Effects of training epochs and parameter size on Llama 3.2 performance. (A) Average performance is compared for models with 1B, 3B, and 8B configurations for 1, 3, and 5 epochs each. (B) Average performance for models trained for 1, 3, and 5 epochs across parameter sizes of 1B, 3B, and 8B. Performance was evaluated using two metrics: Accurate Step, indicating whether the model provided a correctly explained next step, and Holistic Rating, an overall response-quality score from 0 to 1. Each bar represents the mean performance. Accuracy and holistic rating generally increased with both additional training and larger parameter size.

increasing: step accuracy surged between the 1B model (0.603 at 5 epochs) to the 3B model (0.897 at 5 epochs) and 8B model (0.937 at 5 epochs). Consistent with these trends as the parameters increase, logistic regression confirmed significant effects of both factors on performance (**Table 2**). For Accurate Step, advancing from 1B to 3B parameters increased the chance of correct first-step guidance by 2.57-fold ($z = 5.67$, $p < 0.001$), while advancing to 8B yielded a 4.72-fold increase compared to 1B ($z = 8.21$, $p < 0.001$). Odds of an accurate first step skyrocketed by 2.18-fold ($z = 4.37$, $p < 0.001$) when trained on 5 epochs instead of 1. For Holistic Rating, the 1B to 8B increase improved odds of higher-quality responses by 3.37-fold ($z = 12.66$, $p < 0.001$), while going from 1 to 5 epochs improved odds by 2.27-fold ($z = 9.18$, $p < 0.001$). A significant interaction between parameters and epochs for both metrics (Accurate Step: $X^2 = 10.9$, $p = 0.028$; Holistic Rating: $X^2 = 52.9$, $p < 0.001$) indicated that smaller models showed diminished returns from extended training (**Table 2**).

Furthermore, when the epochs remained constant between tests, the models with a higher number of parameters performed better. With a single epoch of training, step accuracy improved from 55.6% for the 1B model to 67.5% for 3B and 80.2% for the 8B model, with holistic ratings of 0.700, 0.694, and 0.865, respectively (**Table 1**). This pattern held true across all training epochs, with larger configurations consistently outperforming smaller ones (**Figure 1B**). This clearly demonstrates the correlation between performance, underscoring the increased power of larger models as high-quality math tutors. These patterns were also reflected in the holistic rating metric. The 8B model improved from 0.865 ± 0.022 SEM at 1 epoch to 0.946 ± 0.015 SEM at 5 epochs, while the 3B model climbed from 0.694 ± 0.022 SEM at 1 epoch to 0.930 ± 0.017 SEM at 5 epochs. The 3B and 1B models showed similar gains, with 1B climbing from 0.700 ± 0.025 SEM at 1 epoch to 0.752 ± 0.026 SEM after 5 epochs, further confirming a transparent size-related advantage (**Table 1**). The differences in accurate step and holistic rating between the 1B and 8B models at all epoch levels were substantial, with the 8B model consistently outperforming the 1B model by multiple length error bars across all training epochs. The Gave Answer metric showed no significant effects of parameters ($p = 0.49$) or epochs ($p = 0.37$), with all models hiding the correct answer over 96% (**Table 2**). This demonstrates consistent pedagogical behavior across model sizes and training durations.

When comparing step accuracy and holistic rating across all nine models, the highest performance was observed for the 8B model trained for five epochs, with 93.7% accuracy and a 0.946 rating. The next best performances were from the 3B configuration trained for five epochs (89.7% accuracy, 0.930 rating) and the 8B configuration trained for three epochs (88.1% accuracy, 0.913 rating) (**Figure 2**). The lowest performance was observed for the 1B configuration at one epoch, with 55.6% accuracy and a 0.700 rating (**Figure 2**). These results underscore a clear relationship between model size, training amount, and performance, illustrating the importance of both fine-tuning and model size for math tutoring quality (**Figure 2**). Notably, the differences between the average for step accuracy and holistic rating between the 1B and 8B models were typically several times greater than the standard error values, further confirming the statistical significance of these findings. For example, the improvement in step accuracy between the 1B and 8B models at 5 epochs (0.603 vs. 0.937) was over five times the combined standard error (0.044 for 1B and 0.022 for 8B), emphasizing a significant difference between the models' performances. Similarly, the ranking of 8B > 3B > 1B remained clear, reinforcing the idea

https://doi.org/10.59720/25-174

Effect	Accurate Step	Holistic Rating	Gave Answer
3B vs 1B	OR = 2.57 ***	OR = 1.62 ***	OR = 0.80
8B vs 1B	OR = 4.72 ***	OR = 3.37 ***	OR = 0.72
3 vs 1 Epoch	OR = 1.71 **	OR = 1.78 ***	OR = 1.34
5 vs 1 Epoch	OR = 2.18 ***	OR = 2.27 ***	OR = 1.52
Interaction	X^2 = 10.9 p = 0.028 *	X^2 = 52.9 p < 0.001 ***	X^2 = 0.91 p = 0.339

Table 2: Logistic regression results comparing model performance across different parameters and epochs. Odds Ratio (OR) indicates how much more likely a model is to succeed compared to the baseline (1B model at 1 epoch); OR > 1 means better performance than baseline. The Interaction row shows whether the effect of epochs changes depending on model size, with chi-squared (X^2) values indicating the strength of this interaction. Significance levels: *** means p < 0.001 (very strong evidence), ** means p < 0.01 (strong evidence), * means p < 0.05 (moderate evidence), and no asterisk means not significant.

that parameter size, not training epochs, was the primary driver of overall performance (**Figure 2**).

On the other hand, the Gave Answer, the amount that the model revealed the final answer, showed no clear correlation between reveal rate, parameters, and epochs. The lowest reveal rate, 1.59%, was shared by several runs (1B-3 epoch, 1B-5 epoch, 3B-3 epoch, 3B-5 epoch, 8B-1 epoch), while the highest rate of 4.76% was recorded for the 3B model at one epoch. No clear trend was observed linking the reveal rate to either model size or number of epochs, indicating a performance limit with slight variation. Most importantly, all models maintained a reveal rate below 5%, indicating that they reliably withheld final answers in favor of explanatory steps and aligned with critical-thinking objectives (**Figure 2**).

Overall, larger models that were trained with increased epochs demonstrated improved performance in both step accuracy and holistic ratings. The 8B model consistently outperformed the 3B model regardless of the number of epochs, and the 3B model in turn outperformed the 1B model. While the 1B configuration plateaued after three epochs, the 3B and 8B configurations continued to improve, suggesting that larger configurations possess a greater capacity for learning and a higher performance ceiling (**Figure 1A**). The variability in the answer-reveal rate highlights minor uncertainty in controlling specific output behavior; however, these variations are minor because the reveal rate stayed below 5% in all experiments, typically remaining under 2%. These findings emphasize the potential and viability of small language models as a math tutor and emphasize clear trade-offs between model size, training epochs, and model performance.

DISCUSSION

Our study shows that fine-tuning Llama 3.2 configurations greatly improves their capacity to act as math tutoring assistants, with increased parameter size and training epochs leading to increased performance. The 8B 5-epoch model obtained a step accuracy of 93.65% and a holistic rating of 0.946, much higher than the 1B 5-epoch model, which was 60.3% accuracy and 0.752 rating (**Table 1**). This corresponds with scaling laws observed showing that performance improves with increased model size, dataset size, and computation (9). Reaching 89.68% accuracy and a 0.930 rating at 5 epochs, the 3B model provides a strong middle ground between efficiency and performance (**Figure 1B**). These results suggest that larger models with more training epochs outperform smaller, less-trained ones in accuracy and tutoring quality (**Figure 2**). Moreover, these results are proving to be reliable and valid through the consistently low standard error values calculated for each model and metric. For every

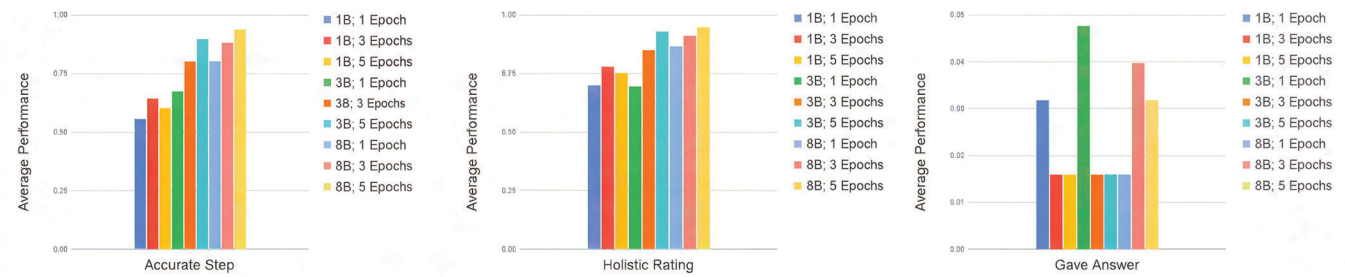


Figure 2: Comparison of Llama 3.2 models by parameter size and training epochs across three metrics. Average performance is compared for all models by the metrics: (A) Accurate Step, (B) Holistic Rating, and (C) Gave Answer. Each bar represents mean scores across all parameter sizes (1B, 3B, and 8B) and epochs (1, 3, and 5). Accurate Step and Holistic Rating improve notably with increased model size and training epochs, while the Gave Answer metric remains low (<5%) across conditions, reflecting a consistent model preference against direct solutions.

https://doi.org/10.59720/25-174

model and epoch, the standard error for step accuracy and holistic rating was less than 0.05 and 0.03, respectively, indicating that the performance of the models is statistically reliable (**Table 1**). The difference in step accuracy between the 8B model at 5 epochs (0.937, SEM = 0.022) and the 1B model at 5 epochs (0.603, SEM = 0.044) is greater than four times the combined SEMs, providing strong evidence that these improvements are not due to random variation. On the other hand, step accuracy increased by just 0.047 between 1 and 5 epochs in the 1B model, yet the combined SEM was 0.088, indicating that this minor increase in performance is due to random variation rather than the increase in training (**Table 1**). Taken together, the error-bar analysis and prior research both confirm the concept that scaling parameters contributes more than extending training in order to increase model accuracy and performance.

The variations in performance can be explained through model capacity and training length. The 1B parameter model showed a decrease in performance between 3 and 5 epochs, going from a step accuracy and holistic rating of 64.3% and 0.778, respectively, to 60.3% and 0.752 after 5 epochs. This plateau is likely due to its limited parameters, capping its ability to learn beyond a given point and introducing overtraining as the epochs continue to increase (**Figure 1A**). Conversely, the 3B and 8B models continuously grew in performance; the 9.5% accuracy gain in the 3B model from 3 epochs (80.2%) to 5 epochs (89.7%) indicates that it has a greater potential to refine mathematical reasoning, compared to 1B, through extended training, making it a strong candidate for further research as a math tutor (**Figure 1A**). This suggests that larger models benefit more from extra epochs than small models, consistent with language model training dynamics (13). These results are also consistent with Chinchilla-style compute-optimal scaling laws, which suggest that performance gains depend not just on model size, but also on a balanced use of compute across training steps and model parameters (14). The Gave Answer rate, where configurations revealed final answers instead of steps, varied across sizes and epochs, ranging from 1.59% at 1B and 3 epochs to 4.76% at 3B and 1 epoch (**Figure 2**). Although variation occurred in the given answer rate, the rate remained below 5%, indicating that our model successfully avoided the direct solution.

The logistic regression results show that parameter count is a more prevalent factor than epochs in tutoring performance. Although both parameters and epochs showed statistically significant effects (all $p < 0.01$), parameter increases produced larger effect sizes (OR = 2.57 for 3B and OR = 4.72 for 8B) compared to epoch increases (OR = 1.71 and 2.18), indicating that parameter count was more impactful for overall model performance. Moreover, the significant interaction term ($X^2 = 10.9$, $p = 0.028$) indicates that epoch effects vary by model size, with smaller models like 1B showing a plateau in performance after 3 epochs, while larger models continued to benefit from more epochs.

With 39% of middle school students in the US scoring below basic proficiency, these findings may have real-world implications for math education (1). The 8B model showed high tutoring quality in our experiments, indicating potential effectiveness in under-resourced classrooms. Conversely, this model remains more expensive, both economically and environmentally, than a similar, smaller model like 3B. Training larger models like 8B requires significantly more computational and energy resources than 1B or 3B, leading to increased carbon emissions and financial costs for hardware and electricity. The negative environmental impact of these language models scales with their size and distribution, leading to challenges with the widespread adoption of increasingly large models (15). Training a single transformer model, with only 213 million parameters, can emit the equivalent of over 626,000 pounds of carbon dioxide, nearly as much carbon as five cars over their lifetimes (16). Now, models are getting exponentially bigger, with models eclipsing one trillion parameters, making optimization key to reduce carbon emissions and protect the environment (17). This trade-off between accuracy, cost, and environmental impact is critical for schools with limited resources, where lower parameter models, despite lower performance, are more efficient and affordable, making them viable solutions for schools. Our results suggest that small language models, balancing efficacy and affordability, could be explored as an effective and scalable solution to students and teachers across the globe (3).

An important question is whether 93.7% accuracy is good enough for actual educational use. These models are not meant to replace human teachers but rather to provide an alternative to the current situation, with students either receiving no help or using general-purpose large language models like ChatGPT that directly reveal answers, consuming large amounts of water and energy while not improving learning outcomes. Our fine-tuned models did not reveal the answer over 96% of the time while providing correct guidance in more than 93% of cases (8B model with 5 epochs), representing a substantial improvement over undertrained or smaller alternatives. Moreover, even the 3B model achieved 89.7% accuracy despite having only a fraction of the parameters, suggesting that small models can provide pedagogical support. Overall, "good enough" AI tutoring that guides students correctly most of the time may be more valuable than perfect tutoring that only a limited number of students can access.

However, several methodological considerations should be noted when discussing the model results and implications. First, our dataset of 538 training queries spans elementary through high school mathematics but does not comprehensively cover all topics across grade-levels, particularly advanced math like calculus. Second, evaluation was conducted using synthetic prompts generated with GPT-4o rather than questions from actual classroom or student interactions, which may not fully capture the complexity of real student needs. Third, the holistic rating metric, while useful for assessing overall response quality, was evaluated by a single rater (the first author), which limits assessment of inter-rater reliability for this subjective measure. Fourth, we tested only the Llama 3.2 model family; evaluating other architectures would help confirm whether parameter scaling effects generalize across different base models. Finally, the base language models were trained on large internet datasets that may contain biases, though our fine-tuning focused specifically on mathematical reasoning rather than general knowledge.

https://doi.org/10.59720/25-174

Despite these limitations, our logistic regression analysis provides strong statistical evidence that parameter count significantly outweighs training duration in tutoring effectiveness. Future work should address these limitations by employing multiple independent raters or a fine-tuned machine learning model, testing multiple model families (e.g., Mistral, Gemma), evaluating performance on standardized benchmarks (GSM8K, MathQA), and validating findings through real-world classroom testing.

Future studies can build on this work by evaluating models using larger datasets and benchmarks, including the GSM8K Benchmark and MathQA, to further evaluate model performance across more diverse datasets (18, 19). The 3B and 8B showed no signs of plateauing during this experiment, indicating that the ceiling for the 8B model performance could be much greater than 5 epochs. By doing this, researchers can ensure that models will be equitable and accurate for all subtopics in elementary, middle, and high school math. Additionally, to address the inconsistency with the model output, future work could adopt verifier models to increase the quality of results. Verifier models have been shown to improve the accuracy and consistency of mathematical reasoning by generating multiple solutions and rating them; however, they also increase the compute and latency as they require additional steps (18). This raises issues regarding efficiency and practicality, as a verifier model would only increase the costs, latency, and environmental damage. Moreover, adjusting other parameters, including learning rate, should be tested as they have the potential to lead to increased performance (20). Finally, incorporating chain-of-thought prompting, shown to increase performance in large models, can increase the accuracy and depth of the responses, making it another possible area of iteration for future work (21). Comparing these models to existing AI tutors, including Khan Academy's Khanmigo tool, would also clarify the real-world impact of small language models acting as AI tutors (4). Ultimately, this study underscores the power of small language models in AI tutoring by showing that models, when fine-tuned properly, have the potential to support scalable, affordable, and effective math learning solutions worldwide.

MATERIALS AND METHODS

We fine-tuned and evaluated several Llama 3.2 configurations as math tutoring assistants using a custom dataset and open-source tools on Google Colab (10, 22). Our goal was to measure each model's ability to generate step-by-step math help across a range of K-12 topics. We curated a dataset of 664 math questions spanning elementary to high school difficulty. Each entry included a student-style prompt and a step-by-step explanation omitting the final answer. Questions were either manually written or synthetically generated using GPT-4o to simulate single-step tutoring interactions. Approximately 30% of the dataset (nearly 200 training prompts and 40 test prompts) were manually created based on common K-12 math topics from Khan Academy and other public educational sources, while about 70% were synthetically generated utilizing GPT-4o with topic—and grade—specific prompts (23, 24). The dataset was split into 538 training and 126 testing questions and uploaded to HuggingFace as JSONL files that are easily accessible for further research and use.

Model training was conducted on Google Colab using an NVIDIA Tesla T4 GPU (12). We used Unsloth's lightweight fine-tuning framework for Llama 3.2 models (1B, 3B, 8B parameters), available at github.com/unslothai/unsloth, using Hugging Face's Supervised Fine-Tuning Trainer, which made training easier and more efficient (11, 25). Each model was trained for 1, 3, and 5 epochs using default hyperparameters, including a learning rate of 2e-4, the AdamW optimizer, and a weight decay of 0.01. Training times varied due to model size and epoch count, ranging from approximately 1 minute 30 seconds (1B model at 1 epoch) to approximately 28 minutes (8B at 5 epochs), underscoring how larger models with more training epochs require significantly increased computational resources. We updated Unsloth's code to use test queries directly from HuggingFace and log all the model responses to a local file, allowing us to analyze the model's performance.

Performance was evaluated by the first author on three metrics: Gave Answer—whether the model gave the user the problem's final answer; Accurate Step—whether the model showed or hinted the mathematically correct first step; and Holistic Rating—a subjective score (0-1) assessing clarity, helpfulness, and educational quality, where 1.0 indicates a clear and helpful response, 0.5 indicates partial guidance with notable issues, and 0.0 indicates a completely unhelpful or complicated response. To ensure consistency, we applied a standardized rubric and rated all 126 test outputs per model in a single session. All 126 test outputs per model were reviewed and scored, with the averages taken as the overall model performance. The Accurate Step metric specifically covered the initial help that the model provided, as the first step is the most critical aspect of scaffolded learning, while the holistic rating covered the overall response quality. For each variant of each model, the standard error of the mean (SEM) was calculated as $\frac{SD}{\sqrt{N}}$ with N = 126 test prompts per model as a measure of variability. To evaluate how model size and training duration influence performance, we performed logistic regression analyses using R statistical software (version 4.5.2; 26). For binary performance metrics, Accurate Step and Gave Answer, we applied binomial logistic regression with counts of successes and failures across 126 tests per configuration. For the Holistic Rating metric (scale from 0-1), we modeled each rating as a proportion of the maximum score using binomial logistic regression. Both Parameters (1B, 3B, 8B) and Epochs (1, 3, 5) were treated as categorical variables to capture possible non-linear effects.

All of the code is publicly available at github.com/yashmahe2020/math-tutor-research, including the modified Unsloth code and customized testing script. The dataset is accessible on HuggingFace at huggingface.co/datasets/yashmahe2018/math-tutor-data, and the experiments can be replicated using Google Colab with a free T4 GPU, requiring only an internet connection. This setup ensures that other researchers or educators interested in AI-driven math tutoring can replicate and expand this research.

Received: May 19, 2025
Accepted: December 3, 2025
Published: November 11, 2026

https://doi.org/10.59720/25-174

REFERENCES

1. "NAEP Mathematics: Mathematics Results." *The Nation's Report Card,* National Center for Education Statistics, 2024, nationsreportcard.gov/reports/mathematics/2024/g4_8/. Accessed 26 Apr. 2025.
2. OECD. *Education at a Glance 2023: OECD Indicators*. OECD Publishing, 2023, https://doi.org/10.1787/e13bef63-en.
3. Bender, Emily M., *et al.* "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?" *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 2021, pp. 610–623. https://doi.org/10.1145/3442188.3445922.
4. "Khanmigo Is Your Always-Available Teaching Assistant." *Khan Academy*. khanmigo.ai/. Accessed 30 Mar. 2025.
5. Kulik, James A., *et al.* "Effectiveness of Intelligent Tutoring Systems." *Review of Educational Research*, vol. 86, no. 1, 2016, pp. 42–78. https://doi.org/10.3102/0034654315581420.
6. OECD. *Shaping Digital Education: Enabling Factors for Quality, Equity and Efficiency*. OECD Publishing, 2023, https://doi.org/10.1787/bac4dc9f-en.
7. "Are Bigger Language Models Always Better?" *IBM*. ibm.com/think/insights/are-bigger-language-models-better. Accessed 26 Apr. 2025.
8. Brown, Tom, *et al.* "Language Models are Few-Shot Learners." *Advances in Neural Information Processing Systems*, vol. 33, Dec. 2020, pp. 1877-1901. https://doi.org/10.48550/arXiv.2005.14165.
9. Kaplan, Jared, *et al.* "Scaling Laws for Neural Language Models." *arXiv preprint arXiv:2001.08361*, 2020, https://doi.org/10.48550/arXiv.2001.08361.
10. "Llama 3.2: Revolutionizing Edge AI and Vision with Open Models." *AI at Meta*, 25 Sept. 2024, ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/. Accessed 20 Feb. 2026.
11. Han, Daniel, *et al.* "Unsloth." *Github*, 2023, github.com/unslothai/unsloth. Accessed 26 Apr. 2025.
12. NVIDIA Corporation. *NVIDIA T4 Tensor Core GPU Datasheet*. NVIDIA, 2019, www.nvidia.com/en-us/data-center/tesla-t4/. Accessed 31 Mar. 2025.
13. Raffel, Colin, *et al.* "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer." *Journal of Machine Learning Research*, vol. 21, no. 140, 2020, pp. 1-67. www.jmlr.org/papers/v21/20-074.html.
14. Hoffmann, Jordan, *et al.* "Training Compute-Optimal LLMs." *arXiv preprint arXiv:2203.15556*, Mar. 2022, https://doi.org/10.48550/arXiv.2203.15556.
15. Ren, Shaolei, *et al.* "Reconciling the Contrasting Narratives on the Environmental Impact of Large Language Models." *Scientific Reports*, vol. 14, no. 1, 1 Nov. 2024, https://doi.org/10.1038/s41598-024-76682-6.
16. Hao, Karen. "Training a Single AI Model Can Emit as Much Carbon as Five Cars in Their Lifetimes." *MIT Technology Review*, 6 June 2019, www.technologyreview.com/2019/06/06/239031/training-a-single-ai-model-can-emit-as-much-carbon-as-five-cars-in-their-lifetimes/. Accessed 3 May 2025.
17. Hudson, Nathaniel, *et al.* "Trillion Parameter AI Serving Infrastructure for Scientific Discovery: A Survey and Vision." *arXiv preprint arXiv:2402.03480,* Feb. 2024, https://doi.org/10.48550/arXiv.2402.03480.
18. Cobbe, Karl, *et al.* "Training Verifiers to Solve Math Word Problems." *arXiv preprint arXiv:2110.14168*, Oct. 2021, https://doi.org/10.48550/arXiv.2110.14168.
19. Amini, Aida, *et al.* "MathQA: Towards Interpretable Math Word Problem Solving with Operation-Based Formalisms." *Proceedings of NAACL-HLT 2019*, 2019, pp. 2357-2367. https://doi.org/10.18653/v1/N19-1245.
20. Smith, Leslie N. "A Disciplined Approach to Neural Network Hyper-Parameters: Part 1 -- Learning rate, Batch size, Momentum, and Weight Decay." *arXiv preprint arXiv:1803.09820*, Mar. 2018, https://doi.org/10.48550/arXiv.1803.09820.
21. Wei, Jason, *et al.* "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models." *Advances in Neural Information Processing Systems*, vol. 35, 2022. proceedings.neurips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.
22. Google. "Google Colaboratory." *Google Colab.* colab.research.google.com. Accessed 20 Feb. 2026.
23. "Math." *Khan Academy*. khanacademy.org/math. Accessed 31 Mar. 2025.
24. OpenAI. "Hello GPT-4o." *OpenAI*, 13 May 2024, openai.com/index/hello-gpt-4o/. Accessed 5 Apr. 2026.
25. "SFT Trainer." *TRL Documentation*, Hugging Face, huggingface.co/docs/trl/en/sft_trainer. Accessed 31 Mar. 2025.
26. R Core Team. "R: A Language and Environment for Statistical Computing" (version 4.5.2) [software]. *R Foundation for Statistical Computing*, 2025. www.R-project.org/.

Copyright: © 2026 Maheshwari and Gupta. All JEI articles are distributed under the Creative Commons Attribution Noncommercial No Derivatives 4.0 International License. This means that you are free to share, copy, redistribute, remix, transform, or build upon the material for any purpose, provided that you credit the original author and source, include a link to the license, indicate any changes that were made, and make no representation that JEI or the original author(s) endorse you or your use of the work. The full details of the license are available at https://creativecommons.org/licenses/by-nc-nd/4.0/deed.en.