

Assessing Small Language Models as Decimal-Arithmetic Tutors: A Measurement Framework

Mai Que Vuong¹ Shahana Ahmadli¹ Michelle Zhou¹ Minseok Kim¹ Shruti Mehta¹
Talita de Paula Cypriano de Souza² Seiji Isotani¹

¹University of Pennsylvania, USA

²Federal Institute of São Paulo (IFSP), Brazil;

Institute of Mathematics and Computer Sciences, University of São Paulo (USP), Brazil

{maivuon, ahmadli07, zhouch, mk8393, shrmeht}@upenn.edu

talita@ifsp.edu.br sisotani@upenn.edu

Abstract

Small language models (SLMs) are increasingly proposed for educational use because they promise lower cost, offline deployment, and stronger data privacy. We report an exploratory evaluation of three sub-2B-parameter models on example-based decimal-arithmetic tutoring. Across structured interactions, all three produced fluent, confident output that masked unstable pedagogy and frequent mathematical errors even on elementary decimal addition and place-value tasks. Building on these observations, we describe an emerging interaction-based measurement framework intended to support more defensible readiness decisions about SLMs as math tutors.

1 Introduction

Small language models (SLMs) are increasingly proposed for classroom use because they run locally, often offline, at a fraction of the cost of frontier systems, and keep student data on-device (Lu et al., 2024; Wang et al., 2025). In this study, we focus specifically on instruction-tuned models below two billion parameters, a class of models whose low resource requirements make local educational deployment particularly feasible. These properties make them especially attractive to under-resourced schools, which are also the settings serving students with the fewest alternative supports. If a small model tutors poorly, the cost falls hardest on learners least able to absorb it. Whether these models are ready to teach is thus an equity question as much as a technical one.

Current evaluation cannot answer this question. Assessments of SLMs emphasize benchmark accuracy and single-response correctness, but teaching is a sustained, contingent process: explaining, recognizing confusion, and adjusting in response (Wood et al., 1976; Puntambekar, 2022). A model can be correct yet pedagogically useless, or well-intentioned yet mathematically wrong; single-turn

metrics are blind to both. Fluency compounds the problem. SLMs produce confident, well-formed explanations that read like good teaching even when the mathematics is wrong, because training rewards confident guessing over expressed uncertainty (Kalai et al., 2025) and a beginner cannot tell the difference. Recent systems do achieve aligned pedagogical guidance, but with far larger 7–17B models supported by retrieval scaffolding (Katharakis et al., 2025). How unscaffolded sub-2B models behave as tutors remains underexamined.

We study three small instruction-tuned models tutoring middle-school-level decimal arithmetic. Decimal addition and place value demand both procedural accuracy and conceptual understanding, making them a sensitive test of teaching rather than computation. Student errors in these domains are extensively catalogued. This allows us to ask not whether a model is wrong, but which documented misconception it reproduces (Isotani et al., 2010). This study is guided by the following research questions:

RQ1. What happens when small language models tutor middle-school-level decimal arithmetic?

RQ2. What mathematical and instructional failures do they exhibit, particularly when a student expresses confusion across multiple turns?

These two questions address a gap that standard model evaluation leaves open. Benchmark scores and single-response accuracy tell us whether a model can produce a correct answer; tutoring, however, is not a single output but an interactive, contingent process (Wood et al., 1976). It is sustained and adaptive: explaining a concept, recognizing when a student is confused, and adjusting support accordingly through ongoing diagnosis (Puntambekar, 2022). RQ1 therefore asks the broad, still largely unexamined question of how small models actually behave when placed in a tutoring role for decimal arithmetic: whether they teach the concept, default to supplying answers, or drift off task.

RQ2 narrows the focus to the moments where tutoring is most consequential and where we expect small models to be weakest: the multi-turn exchanges that follow a student's confusion. Documenting these failures concretely matters because small models are fluent: they generate confident, well-formed explanations that can sound like good teaching even when the mathematics is wrong, and a beginning learner is poorly positioned to detect the difference.

We contribute (1) a two-axis framework separating mathematical validity from pedagogical appropriateness, with each rubric dimension grounded in learning-sciences research; (2) a structured multi-turn evaluation of three SLMs across 72 interactions conducted by four research-team raters; and (3) a taxonomy of recurring failures: premature answer-giving, rigid strategy application, and non-contingent response to confusion that single-turn evaluation cannot detect.

2 Theoretical Framework

Judging whether a tutoring response is adequate requires two distinct standards, because mathematical error and pedagogical failure are evaluated against different bodies of evidence. We treat adequacy as resting on two axes: mathematical validity, whether the content is correct, and pedagogical appropriateness, whether the instructional behavior supports learning. A response can fail on either axis independently, as a model can compute correctly while teaching poorly, or teach with good instructional form while getting the mathematics wrong, so an adequate evaluation must assess both.

2.1 Mathematical Validity

We treat mathematical validity not as a binary check on the final answer but as correctness across the result, the procedure, and the underlying place-value representation (Hiebert and Wearne, 1986). This distinction matters because a model can arrive at a correct numeric answer through an incorrect procedure, or state a correct procedure while misrepresenting what the digits mean, which are failures a result-only check would miss.

Decimal errors are also not idiosyncratic. Synthesizing findings across an extensive body of research, Isotani et al. (2010) organize decimal errors into a taxonomy of persistent misconceptions rooted in learners' prior knowledge of whole numbers and fractions; for example, treating a longer

decimal as larger because it has more digits, treating a shorter decimal as larger, or treating decimals with different numbers of trailing digits (e.g., .07 and .7) as equal. This taxonomy gives us a coding scheme for the models' mathematical failures: rather than asking only whether a model is wrong, we ask which documented misconception its error instantiates. This reframing is central to our argument, because it shows that when a small model errs, it is reproducing a recognizable pattern of novice reasoning, and doing so with a confidence that a real novice would not have earned.

2.2 Pedagogical Appropriateness

Pedagogical appropriateness draws on three complementary strands of learning-sciences research, each targeting a different way instruction can fail even when the underlying mathematics is not at issue.

Example quality. Because the tutoring task we study relies on concrete examples to teach decimal concepts, the quality of a model's examples is central to evaluating it. Example-based learning (teaching through illustrative cases rather than abstract rules alone) is well established as effective for novices, but its benefit depends on the design and relevance of the example rather than following automatically from offering one (van Gog and Rummel, 2010). An effective instructional example must instantiate the target concept: an example drawn from a different operation, or an analogy with no structural relationship to the concept, does not teach it and may mislead. A good example also connects the concrete case to the underlying principle so that understanding transfers to new problems, and a varied set of examples supports learners in abstracting the general structure rather than fixating on one surface case. Because our target learners are beginners, who depend most heavily on the quality of the examples they receive, an irrelevant or mathematically flawed example does not merely fail to teach; it actively models a misconception.

Cognitive load. Cognitive Load Theory holds that working memory is severely limited and that learning falters when demands exceed capacity. The theory distinguishes intrinsic load (the inherent difficulty of the material), extraneous load (strain from poor instructional design), and germane load (productive effort toward schema construction) (Sweller et al., 1998). For novice learners, effective instruction minimizes extraneous load

so that limited capacity is spent on the target concept. This gives us a direct lens on one recurring failure mode: when a model answers a basic decimal-alignment question by introducing rounding, floating-point precision, and base conversion, it loads a beginner’s working memory with content irrelevant to the task, impeding rather than supporting learning. Because germane load supports schema construction only when intrinsic and extraneous load are kept within capacity (Sweller et al., 1998), a model that piles on unclear or irrelevant content does not create productive difficulty; it pushes a novice past the point where learning is possible, and the pedagogical consequence is serious: misconceptions go undetected until they surface later as systematic failures.

Scaffolding and feedback. Effective tutoring is not the transmission of correct information but the contingent management of a learner’s effort. Wood et al. (1976), who introduced the term, defined scaffolding as the process by which a more capable other controls the elements of a task that exceed the learner’s current capacity, allowing the learner to concentrate on the parts within reach: a process rooted in Vygotsky’s (1978) zone of proximal development and later specified as requiring ongoing diagnosis of the learner’s understanding and fading, the gradual, readiness-responsive withdrawal of support as competence grows (Puntambekar, 2022). Reiser and Tabak (2014) further specify what scaffolding does beyond securing a correct answer: it focuses attention on critical features and prompts explanation and reflection rather than simply delivering solutions. A tutor that supplies the answer short-circuits this function, removing the productive work through which understanding is built; a tutor that repeats an unchanged explanation when a learner signals confusion fails the diagnostic requirement, having not identified the difficulty at all.

These three strands converge on two instructional failures we treat as central to this study. First, premature answer-giving: a model that produces the final result despite an explicit instruction to withhold it collapses the scaffold and eliminates the learner’s opportunity to reason. Second, non-contingent response to confusion: a model that repeats, substitutes an irrelevant example, or escalates to more advanced content when a learner expresses difficulty has not diagnosed the specific misconception. Both failures are invisible to single-

response, accuracy-only evaluation, which is precisely why the multi-turn interaction protocol described in Section 3 is necessary to detect them.

3 Methods

3.1 Sample

We evaluated three small, instruction-tuned language models: LLaMA-3.2-1B-Instruct, Qwen2.5-0.5B-Instruct, and Gemma-3-1B-IT (GGUF), ranging from roughly 0.5 to 1 billion parameters. Four raters interacted with each model and recorded structured observations. The analytic focus is decimal addition and decimal place-value tasks, chosen because they require both procedural accuracy and conceptual explanation and so test whether a model can teach rather than merely compute.

3.2 Tutoring Prompt

Each model received an identical system prompt establishing a tutoring role with two binding constraints: never give the final answer, and teach using short examples, together with one in-context demonstration of the intended Socratic style:

You are a helpful math tutor. Rules: NEVER give the final answer. Give short examples to explain the concept. Example — Student: “Is 0.5 bigger than 0.05?” Tutor: “Let’s try a different comparison. Compare 0.4 and 0.35. Write them with the same decimal places: 0.40 and 0.35. 40 hundredths is greater than 35 hundredths, so 0.40 is bigger than 0.35. Now, can you answer your exercise?” Task: Help the student with the question: [question inserted here]

This prompt fixes the instructional strategy across models, so that observed differences reflect model behavior rather than prompt variation, and it makes the central tension of the task explicit: a model that supplies the answer, however correct, has violated the tutoring constraint.

3.3 Procedure

Using the shared tutoring prompt, four research-team raters conducted multi-turn interactions with each of the three models on decimal addition and place-value tasks. The protocol standardized the tutoring role, task domain, and initial instructional constraints, while allowing follow-up turns to probe how models responded when a student expressed confusion, questioned an example, or requested an alternative explanation. This controlled flexibility was intended to capture contingent tutoring behavior rather than performance on a single predetermined response.

3.4 Rubric

Raters scored each interaction on five quality dimensions and two binary behavioral flags, based on Souza et al. (2026). The dimensions were grounded in the theoretical framework. The five quality dimensions were: (1) relevance, (2) correctness, (3) level of detail, (4) variety, and (5) abstract-concrete connections. *Relevance* assessed whether the model's examples directly addressed the question rather than drifting off-topic. *Correctness* assessed whether examples were mathematically and procedurally accurate, including appropriate use of place value. *Level of detail* assessed whether examples provided enough steps for a student to follow and reproduce the procedure independently. *Variety* assessed whether the model used different examples and levels of complexity rather than repeatedly relying on the same pattern. *Abstract-concrete connections* assessed whether the model connected decimal notation to underlying place-value meaning rather than providing procedural instructions alone. Together, these dimensions provided an initial operationalization of the framework's two axes: correctness primarily captured mathematical validity, whereas the remaining four dimensions captured complementary aspects of pedagogical appropriateness.

Two binary flags capture critical failures that override surface quality. The *immediate-answer* flag records whether the model supplied a complete solution at any point in the interaction, in violation of the tutoring constraint and of the scaffolding principle that support should prompt reasoning rather than deliver the result. The *hallucination* flag records whether the model produced content that was unrelated to the task or mathematically fabricated. Raters additionally noted response latency and an overall judgment of the interaction's instructional quality (satisfactory, partially satisfactory, or unsatisfactory).

3.5 Coding

Beyond the rubric scores, raters qualitatively coded recurring failure types observed across interactions, including direct answer-giving, hallucinated or fabricated reasoning, place-value confusion (such as treating 0.5 and 0.05 as equal), rigid application of a single reasoning strategy across task types, and failure to adapt when a student expressed confusion. These codes form the basis for the failure analysis reported under RQ2.

interactions ($n = 24$ per model), organized around the two research questions introduced above. We first present graded quality ratings and binary failure flags addressing RQ1, characterizing overall response quality and the prevalence of critical instructional failures across models. We then turn to RQ2, drawing on qualitative coding of multi-turn exchanges to characterize how these failures manifest specifically when a student expresses confusion. Together, the quantitative and qualitative results point to a consistent picture: models varied in surface-level quality but converged on the same underlying failures of mathematical validity and contingent scaffolding.

RQ1: What happens when small language models are used to tutor middle-school-level decimal arithmetic?

Across the 72 rated interactions ($n = 24$ per model), the three small language models showed low- to moderate response quality but pervasive instructional and mathematical failures. Mean ratings on the five-point rubric ranged from 1.63 to 3.17 (Table 1). LLaMA-3.2-1B-Instruct received the highest scores overall, particularly for relevance ($M = 3.17$, $SD = 1.01$) and correctness ($M = 2.17$, $SD = 1.31$), while Gemma-3-1B-IT-GGUF and Qwen2.5-0.5B-Instruct scored lower and more similarly to each other ($M = 1.88$, $SD = 1.12$). Across models, relevance was the strongest dimension and abstract-concrete connections and variety the weakest, indicating that models more often stayed on topic than they linked examples to underlying place-value concepts or varied their explanations.

Additionally, models were evaluated for their propensity to produce hallucinated and immediate outputs (Table 2). Hallucinated or fabricated content appeared in 87.5% of responses for every model (21 of 24), with no differentiation across models. Immediate-answer cases, supplying the final result despite explicit instructions to withhold it, occurred in 70.8% to 83.3% of interactions, highest for Qwen2.5-0.5B-Instruct and lowest for LLaMA-3.2-1B-Instruct.

4 Results

RQ2: What mathematical and instructional failures do small LLMs exhibit, particularly when a student expresses confusion across multiple turns?

Three failure patterns recurred across raters and models. First, fluent output did not track mathematical accuracy: models frequently confused tenths and hundredths, treated unequal decimals as equal, or fabricated intermediate values, while still sounding confident. Second, models applied a single reasoning template, most often numeric comparison, regardless of whether the task called for addition or place-value decomposition. Third, multi-turn responsiveness was the weakest link: when a rater expressed confusion, models typically repeated the prior explanation, substituted an unrelated example, or escalated to advanced, off-task content rather than identifying the specific misconception. Sample interactions across models illustrating all three cases above are reported in Appendix A. Stylistic differences were also evident: LLaMA-3.2-1B-Instruct was more exploratory but unstable, Qwen2.5-0.5B-Instruct somewhat more accurate and detailed, and Gemma-3-1B-IT-GGUF was weakest overall, though all three exhibited substantive limitations. Taken together, these results suggest that unscaffolded small language models can produce locally fluent, plausible-sounding tutoring language while failing the two fundamental properties, mathematical validity and contingent scaffolding, that represent adequate tutoring.

4.1 Anticipated Contributions

This work makes two main contributions. Empirically, our experiment offers concrete evidence that current sub-2B models are not reliable as tutors for beginning decimal learners, posing real risks if deployed in under-resourced classrooms precisely because they sound helpful. Methodologically, we describe how literature-informed constructs, standardized interaction tasks, shared transcript scoring, and disagreement analysis can be combined into a more transparent readiness assessment. Such decisions matter most in under-resourced settings, where low-cost SLMs are most attractive and struggling learners are least able to detect errors.

5 Discussion and Future Work

This study shows an important difference between fluent responses and effective tutoring. The three models often produced explanations that sounded

clear and helpful. However, many responses contained mathematical errors or provided weak instructional support. This suggests that fluent language alone does not mean that a model is ready to tutor students. This is especially concerning for beginning learners, because they may not recognize an incorrect explanation.

The findings also show why mathematical and instructional quality should be evaluated separately. A response can be mathematically correct but still be poor instruction. For example, a model may give the final answer, use an irrelevant example, or fail to address the student's confusion. A response may also sound supportive while presenting incorrect mathematics. Therefore, benchmark accuracy and the correctness of a single response are not enough to evaluate an AI tutor. Evaluation should also examine how the model responds during an interaction and whether it adjusts its support when a student expresses confusion.

These findings do not mean that all open-source language models are unsuitable for tutoring. The models in this study had fewer than two billion parameters. They were also tested without retrieval tools or other forms of external support. Their limited performance may be partly related to their small size. Future research should compare these models with larger open-source models, including models with 10 to 20 billion parameters. The same tasks, prompts, and rubric should be used across models. This comparison can help us understand whether model size is related to tutoring performance. Future research can also examine whether retrieval tools, more structured prompts, or training for specific tasks can improve mathematical accuracy and instructional support.

Overall, this study presents an interaction-based approach to evaluating small language models as tutors. Low cost, privacy, and offline access make these models attractive for educational use. However, these advantages do not show that a model is ready for the classroom. Before using these models as tutors, researchers should evaluate both their mathematical accuracy and their ability to provide support based on the student's needs.

Limitations

This study has several limitations. First, members of the research team interacted with the models and also rated their responses. Although this dual role may have introduced potential bias, the structured

evaluation protocol was designed to minimize its influence. Second, this study included only three models and focused on decimal arithmetic. The findings may not apply to other models or mathematical topics. Third, the raters used selected responses to represent student confusion. These responses may not represent how actual middle-school students interact with an AI tutor. Future research should develop and test the interaction protocol using real student dialogue or research on how students express confusion and ask for help.

References

- James Hiebert and Diana Wearne. 1986. Procedures over concepts: The acquisition of decimal number knowledge. In James Hiebert, editor, *Conceptual and Procedural Knowledge: The Case of Mathematics*, pages 199–223. Lawrence Erlbaum.
- Seiji Isotani, Bruce M. McLaren, and Miriam Altman. 2010. [Towards intelligent tutoring with erroneous examples: A taxonomy of decimal misconceptions](#). In Vincent Aleven, Judy Kay, and Jack Mostow, editors, *Intelligent Tutoring Systems (ITS 2010)*, volume 6095 of *Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg.
- Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and Edwin Zhang. 2025. [Why language models hallucinate](#). *arXiv preprint arXiv:2509.04664*.
- Konstantina Katharakis, Silvia Rossi, and Raghava Rao Mukkamala. 2025. [Small language models for curriculum-based guidance](#). *arXiv preprint arXiv:2510.02347*.
- Zhenyan Lu, Xudong Li, Dongqi Cai, Rongjie Yi, Fangming Liu, Xiwen Zhang, et al. 2024. [Small language models: Survey, measurements, and insights](#). *arXiv preprint arXiv:2409.15790*.
- Sadhana Puntambekar. 2022. [Distributed scaffolding: Scaffolding students in classroom environments](#). *Educational Psychology Review*, 34(1):451–472.
- Brian J. Reiser and Iris Tabak. 2014. Scaffolding. In R. Keith Sawyer, editor, *The Cambridge Handbook of the Learning Sciences*, 2nd edition, pages 44–62. Cambridge University Press.
- John Sweller, Jeroen J. G. van Merriënboer, and Fred G. W. C. Paas. 1998. [Cognitive architecture and instructional design](#). *Educational Psychology Review*, 10(3):251–296.
- Souza, T. d. P. C. de, Mehta, S., Uema, M. A., de Paula, L. B., & Isotani, S. (2026). Can AI be a teaching partner? Evaluating ChatGPT, Gemini, and DeepSeek across three teaching strategies. *arXiv preprint arXiv:2603.26673*.
<https://arxiv.org/abs/2603.26673>
- Tamara van Gog and Nikol Rummel. 2010. [Example-](#)

[based learning: Integrating cognitive and social-cognitive research perspectives](#). *Educational Psychology Review*, 22(2):155–174.

Lev S. Vygotsky. 1978. *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press. Edited by Michael Cole, Vera John-Steiner, Sylvia Scribner, and Ellen Souberman.

Fali Wang, Zhiwei Zhang, Xianren Zhang, Zongyu Wu, Tzuhao Mo, Qiuha Lu, et al. 2025. [A comprehensive survey of small language models in the era of large language models: Techniques, enhancements, applications, collaboration with llms, and trustworthiness](#). *ACM Transactions on Intelligent Systems and Technology*, 16(6):1–87.

David Wood, Jerome S. Bruner, and Gail Ross. 1976. [The role of tutoring in problem solving](#). *Journal of Child Psychology and Psychiatry*, 17(2):89–100.

Appendix A: Illustrative Interaction Artifacts

The excerpts below illustrate the three recurring failure patterns described in the Results section: mathematically incorrect output delivered with confidence, rigid reliance on a single reasoning template, and non-contingent responses to expressed student confusion.

Excerpt 1: Gemma-3-1B-IT (GGUF)

Prompt: Compare the hundredths place in 4.35 and 4.38.

Student: “Is 4.35 bigger than 4.38?”

Model Output: “Let’s look at the ones place. 5 is greater than 8. So, 4.35 is bigger than 4.38.”

Student: “I don’t understand.”

Model Output: “Imagine you’re comparing apples and oranges.

Apples: Have a rounder shape – more consistent.

Oranges: Have a more irregular shape – they have a lot of variation.

4.35 is closer to being a perfectly rounded apple, while 4.38 is a little bit more spread out and less uniform. That difference in shape is reflected in the ones place.”

Conclusion: When students expressed confusion, models exhibited several recurring instructional failures: they sometimes provided incorrect mathematical answers or reasoning, rarely diagnosed or addressed the specific source of the student’s confusion, introduced analogies such as blocks, LEGO, or pizza that were not always relevant to the underlying concept, repeated previous explanations without simplifying or restructuring them, and rarely changed their instructional strategy in response to persistent student difficulty.

Excerpt 2: Qwen2.5-0.5B-Instruct

Prompt: Judge whether aligning decimals is necessary for accurate results.

Model Output: “The concept of decimal alignment is essential in many areas of mathematics and computer science. In scientific notation, 3.14 can be written as 3.14×10^2 . The significance is that it helps maintain precision when performing arithmetic operations. For example, $3.14 + 0.5 = 3.64$, but if written as 3.14×10^2 , rounding to the nearest integer gives 3.14.”

Student: “Give me simple samples to explain what occurs if we misalign decimals.”

Model Output: “Incorrect rounding — a calculator may round away from zero, giving a smaller value than actual. Floating-point precision issues — operations on very close numbers may not give the expected result. Incorrect sign convention — mistakenly assigning a negative value to a positive number. Incorrect base conversion — performing operations in bases other than 10 without converting first.”

Conclusion: The model’s explanation of decimal alignment is overly complex and inaccessible for a student who is still developing basic math skills. Instead of providing a clear, foundational definition such as “digits must be placed in the correct position (ones under ones, tenths under tenths, and so on) to ensure accurate results,” the model drifts into advanced topics such as rounding, floating-point precision, and base conversion. These concepts are far beyond the scope of the task and would likely confuse rather than support a beginner. Compounding this, the model never offers a single concrete, practical example to ground its explanation. A student is left with abstract terminology and no actionable understanding of what decimal alignment actually means or why it matters.

Excerpt 3: LLaMA-3.2-1B-Instruct

Prompt: Solve $8.09 + 1.5$.

Model Output: “ $8.09 + 1.5 = 9.59$ ”

Student: “Why did you give the final answer?”

Model Output: “Now, let’s focus on one column at a time. $2 + 4 = 6$. So, the result so far is: $8.09 + 1.5 = 9.59$.”

Student: “I think the example you have provided is not relevant.”

Model Output: “Let’s focus on the hundredths place: $2 + 4 = 6$. So, the result so far is: $8.09 + 1.5 = 9.59$.”

Conclusion: The model supplied an immediate answer following the initial prompt; although the result was numerically correct, the response conflated the tenths and hundredths place values. Furthermore, the examples provided were not relevant to the task: despite⁴⁰³ the operation being addition, the model offered an example involving subtraction and integers,

which increased rather than reduced the complexity of the explanation. Subsequent attempts to redirect the model using the prompts “I think you are wrong,” “give me a relevant example,” and “I cannot understand you” did not resolve these issues; the model continued to hallucinate throughout the interaction.

Appendix B: Tables

Model	Relevance	Correctness	Detail	Variety	Abstract-concrete
Gemma-3-1B-IT-GGUF	2.83 (1.05)	1.88 (1.12)	2.58 (1.21)	2.00 (1.10)	2.21 (1.28)
LLaMA-3.2-1B-Instruct	3.17 (1.01)	2.17 (1.31)	2.50 (1.02)	1.92 (1.14)	1.67 (0.87)
Qwen2.5-0.5B-Instruct	2.88 (1.15)	1.88 (0.99)	2.38 (1.24)	1.75 (0.94)	1.63 (0.92)

Table 1: Descriptive statistics for SLM response quality ratings by model and metric. $N = 72$, $n = 24$ per model. Values are means with standard deviations in parentheses. Ratings were made on a 1–5 scale, with higher scores indicating higher quality.

Model	Halluc.	Imm. Answer
Gemma-3-1B-IT-GGUF	21 (87.5%)	17 (70.8%)
LLaMA-3.2-1B-Instruct	21 (87.5%)	19 (79.2%)
Qwen2.5-0.5B-Instruct	21 (87.5%)	20 (83.3%)

Table 2: Frequency and percentage of hallucination and immediate answer across SLM models. $N = 72$, $n = 24$ per model. Values represent the number and percentage of responses within each model classified as containing a hallucination or providing an immediate answer.