

Teaching Models to Read and Respond to a Learner's Reasoning

SENSUS Bench and an Open SFT Dataset for Grade 4 and 5 Fraction and Decimal Errors

LORM Labs · Technical report · October 27, 2026

Abstract

When a learner makes a mistake with fractions or decimals, the reasoning they write down usually shows why. We ask whether a language model can use that evidence. Can it name the likely error, present it as a likely explanation and not a certainty, credit what is correct, explain the idea and end with a next step? SENSUS Bench v1.0 tests this on a development set of 76 grade 4 and 5 cases. Two model graders from different companies score each reply line by line against a checklist written for its case. We release SENSUS SFT, 1,015 synthetic training conversations, and use it to fine-tune low-rank adapters on three open models. Qwen3.8-27B trained on SENSUS scores 0.825 on average over two training runs (0.811 and 0.838), against 0.290 for the same base before training, a pooled gain of +0.534 (95% CI [+0.476, +0.591]). GPT-6 Sol, the highest-scoring general-purpose model, scores 0.525. A control trained on generic tutor replies to the same prompts scores 0.445, so the SENSUS replies as a whole teach far more than fine-tuning on tutor replies in general. The general-purpose models received a one-line tutor instruction that does not describe the target behaviours, and most of their gap lies in those behaviours. With the length, hedging and certainty rules set aside, the fine-tuned 27B model is level with Opus 5.5 and still ahead of GPT-6 Sol. On these cases the share of 27B replies that both graders marked free of false statements rose from 75% untuned to 91% for the second SENSUS run, and replies triggering a listed serious error fell from 26 of 76 to 0. GPT-6 Sol still made no false statements, and the smaller fine-tuned models make them more often than any general-purpose model.

1. Introduction

Ask a capable language model to help a ten-year-old who has decided that $4/10$ is bigger than $3/6$ because 4 is bigger than 3, and it will usually explain how to compare fractions correctly. It is less likely to say what went wrong in the child's own method. Feedback research suggests that this second step matters. Feedback works best when it addresses the gap between what the learner did and the goal (Hattie and Timperley, 2007), and it should come in pieces small enough to act on (Shute, 2008). Errors in elementary fractions and decimals are often systematic. Learners carry whole-number rules into fractions (Ni and Zhou, 2005) or judge a decimal by how many digits it has (Resnick et al., 1989; Steinle and Stacey, 2004). Learning research has long treated such errors as knowledge in transition (Smith et al., 1993).

SENSUS targets two specific failures. When a learner states an incorrect way of working, a model often does not name the learner's likely error, or it states its diagnosis as certain when it can only be likely. We define five behaviours for a reply to a learner's written reasoning. The first two address those failures directly, and the other three make the reply useful.

B1. Name a likely error tied to what the learner wrote.

B2. Hedge the diagnosis. The cause of an error is an inference about someone else's thinking.

B3. Credit what is correct.

B4. Explain or model the idea the error concerns.

B5. End with a next step the learner can take, usually a question.

A reply should also be correct and short. We drew these behaviours from research on feedback, misconceptions and responsive teaching (Section 3.3), and this study does not test whether each one, or its weight in the score, improves learning. SENSUS Bench turns them into case-specific checklist lines that measure observable properties of a single reply, and SENSUS SFT gives 1,015 examples of replies that show them. Figure 1 shows one case from start to finish.

Case · Comparing fractions (4.NF.A.2)

Question. Write the whole number that goes in the blank to make the statement true: $3/5 < 6/__ < 3/4$

What the learner did and said. The learner is filling the blank in $3/5 < 6/__ < 3/4$. The learner writes: "3/5 is 6/10 and 3/4 is 6/8. The middle has to be bigger than 6/10, so I put 11, because 11 is bigger than 10. My answer is 6/11."

Qwen3.8-27B+SENSUS · score 1.00

Your conversions to $\frac{6}{10}$ and $\frac{6}{8}$ keep the fractions' values the same. It sounds like you may be using the rule for numerators when checking the middle fraction's denominator. With the same numerator, a larger denominator means smaller pieces, so $\frac{6}{11}$ is smaller than $\frac{6}{10}$. To be between the endpoints, the middle fraction needs a denominator between 8 and 10. What whole number can you put in the blank?

GPT-6 Sol · score 0.43

You made a great start by rewriting $\frac{3}{5}$ as $\frac{6}{10}$ and $\frac{3}{4}$ as $\frac{6}{8}$!

Here's the tricky part: when the numerators are both 6, a **larger denominator makes a smaller fraction**. So $\frac{6}{11}$ is *less* than $\frac{6}{10}$, not greater.

The whole number between 8 and 10 is **9**, and

$$\frac{6}{10} < \frac{6}{9} < \frac{6}{8}.$$

So the number in the blank is **9**.

Kind	Checklist line	27B+SENSUS (G1 / G2)	GPT-6 Sol (G1 / G2)
Correct	Nothing false (for example, saying 6/11 is greater than 6/10, or saying 3/5 is not equal to 6/10).	✓ / ✓	✓ / ✓
B1	Names a likely error tied to the learner's choice of 11: treating a bigger denominator as making a bigger fraction.	✓ / ✓	✗ / ✓
B2	Says the error as likely, not certain, using words such as "looks like", "might", "maybe" or "it seems". A reply that names no reason for the error is marked no.	✓ / ✓	✗ / ✗
B4	Explains in words or with a picture that a larger denominator means smaller pieces when the numerators are the same.	✓ / ✓	✗ / ✗
B3	Names the learner's correct renaming of 3/5 as 6/10 or of 3/4 as 6/8.	✓ / ✓	✓ / ✓
B5	Ends with a question or task the learner can answer next. A trailing "Does that help?" alone does not count.	✓ / ✓	✗ / ✗
Length	Is about 120 words or fewer.	✓ / ✓	✓ / ✓

Figure 1. One SENSUS Bench case, start to finish. This illustrative case was selected as the first in file order where the fine-tuned 27B model fully passed and GPT-6 Sol scored below 0.6. Both models see only the question and the learner's work. Each grader marks every checklist line (G1 / G2), and a line counts only when both mark it met (green); red marks a line that did not count. The fine-tuned model names the likely error, hedges it, explains it in terms of piece size, credits the learner's correct renaming and asks a next question. GPT-6 Sol states the correct comparison rule and gives the answer, but does not explain the rule in terms of piece size or leave the learner a next step; the graders split on whether it names the error.

We fine-tune three open models on SENSUS SFT and evaluate them, their untuned bases, matched generic controls and four general-purpose models on SENSUS Bench. Five findings stand out.

- About a thousand examples teach the behaviour.** Qwen3.8-27B rises from 0.290 to a two-run mean of 0.825 (0.811 and 0.838; pooled gain +0.534, 95% CI [+0.476, +0.591]). A control trained on generic tutor replies to the same prompts reaches 0.445, so the SENSUS replies as a package, not tutor-style fine-tuning in general, account for the gain.
- A 27B open model outscores every general-purpose model we tested.** It leads on all 12 standards and beats GPT-6 Sol on 68 / 3 / 5 cases (wins / losses / ties), under the same one-line tutor instruction (Figure 2).
- The 27B model also became more accurate.** The share of replies both graders marked free of false statements rose from 75% untuned to 91% for the second SENSUS run, and replies with a listed serious error fell from 26 of 76 to 0.

4. **The gain lies in the targeted behaviours.** Fine-tuning moves naming, hedging and the next step (Figure 4). With the length, hedging and certainty rules set aside, the fine-tuned 27B model is level with Opus 5.5 and still ahead of GPT-6 Sol.
5. **Style can outrun accuracy in smaller models.** At 8B and 14B the fine-tuned models adopt the reply style fully but state something false in 51% and 30% of replies.

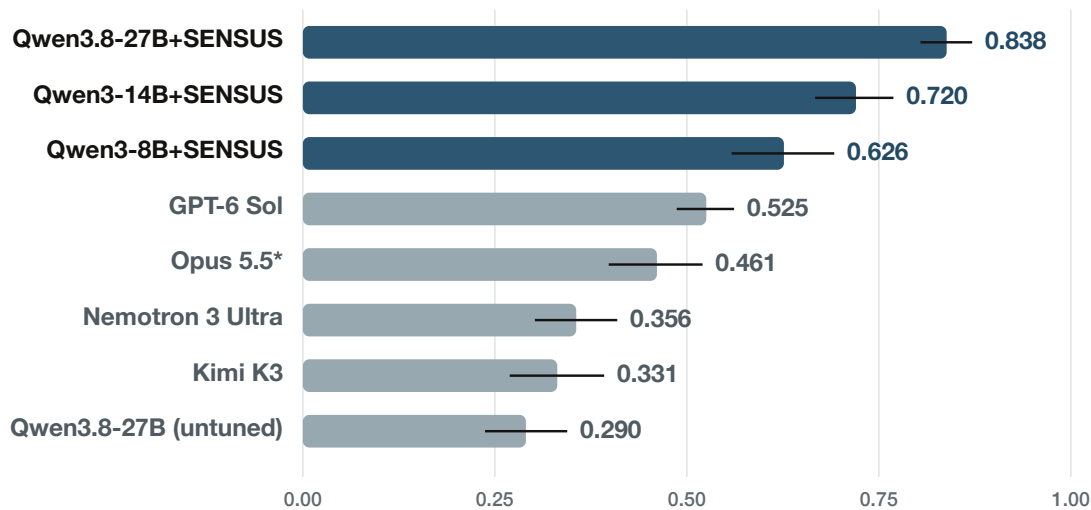


Figure 2. All three fine-tuned open models score above every general-purpose model. Mean checklist score on the 76 cases (0 to 1) with 95% bootstrap intervals. Fine-tuned models in blue. *Graded with an alternate first grader.

The general-purpose models were not told about B1 to B5, so these comparisons describe their default behaviour under a short tutor instruction. Prompting alone did little in the one recorded test: Qwen3-8B given five SENSUS examples in its prompt scored 0.312, against 0.239 untuned and 0.578 to 0.647 after fine-tuning. A general-purpose model prompted with B1 to B5 was not run, and it is the first item of future work.

The paper makes four contributions.

1. **SENSUS Bench v1.0.** 76 cases over 12 Common Core standards (NGA and CCSSO, 2010), with 549 checklist lines and 218 serious errors. The two graders agree on 94.7% of line judgments ($\kappa = 0.89$).
2. **SENSUS SFT.** 1,015 conversations under CC BY 4.0. No question is shared with the benchmark, and the release includes checksums and an offline verifier.
3. **Training results.** Fine-tuning raises Qwen3.8-27B from 0.290 to a two-run mean of 0.825 (pooled gain +0.534, 95% CI [+0.476, +0.591]), with matched generic controls and two training runs per recipe at 8B, 14B and 27B. At 27B, false statements fall from 25% to 9% of replies and serious errors from 26 to 0.
4. **A breakdown of the gain.** Rescoring the recorded marks under seven rules separates the behavioural gain from mathematical content and shows what fine-tuning cost in accuracy.

2. Related Work

Tutoring benchmarks. MathDial collects teacher moves in dialogues with a model that simulates student errors (Macina et al., 2023). Bridge records the decisions expert tutors make when a student errs, and shows that models improve when given those decisions (Wang et al., 2024a). The taxonomy of Maurya et al. (2025), which includes mistake identification and providing guidance, underlies the BEA 2025 shared task (Kochmar et al., 2025). MathTutorBench (Macina et al., 2025), stepwise verification before remediation (Daheim et al., 2024) and evaluation programs for educational assistants (Jurenka et al., 2024) cover broader tutoring skills, and TutorBench scores explanations, feedback and hints against rubrics written for each sample (Srinivasa et al., 2025). SENSUS is narrower. Its replies are single turns, and its target is a hedged diagnosis drawn from the learner's own words. Case-specific rubrics are not new. Our contribution is to pair them with this reply policy, an open training set, same-base training comparisons and an analysis of how the scoring rules shape the results.

Misconception diagnosis. Work on diagnosing misconceptions with language models has mostly addressed algebra (Otero et al., 2025), and models have been used to simulate students who hold specific misconceptions (Sonkar et al., 2024). We study fractions and decimals, and we ask the model to act on its diagnosis in the reply.

Fine-tuning tutors on synthetic data. TutorChat and SocraticLM fine-tune on large synthetic tutoring corpora (Chevalier et al., 2024; Liu et al., 2024), and Petukhova et al. (2026) add preference optimization for mistake remediation. Synthetic instruction data is now common practice (Wang et al., 2023; Long et al., 2024). LIMA showed that around a thousand examples can set a model’s response style (Zhou et al., 2023), and imitation data can change style more than capability (Gudibande et al., 2024). Our results are consistent with both. We train low-rank adapters (Hu et al., 2022), which tend to forget less of the base model than full fine-tuning does (Biderman et al., 2024).

Feedback and teaching research. Research on feedback (Hattie and Timperley, 2007; Shute, 2008; Kluger and DeNisi, 1996), formative assessment (Black and Wiliam, 1998), student errors (Brown and Burton, 1978) and teacher noticing (Jacobs et al., 2010) shaped the reply policy. Section 3.3 sets out how each behaviour and checklist line follows from it.

Model graders. Model judges approximate human preferences in many settings (Zheng et al., 2023; Chiang and Lee, 2023). They are also sensitive to answer order (Wang et al., 2024b), favor their own outputs (Panickssery et al., 2024), and respond to length and style (Dubois et al., 2024; Feuer et al., 2025). Binary checklists make judgments more reliable (Lee et al., 2025; Ye et al., 2024), and conversation-specific rubrics with penalized criteria have been used at scale in health (Arora et al., 2025). We use both ideas, and Sections 7 and 8 measure how far our results depend on style.

3. SENSUS Bench v1.0

3.1 Cases

Each case is one teaching moment. It has a mathematics question labeled with its grade band and Common Core standard, a short description of what a synthetic learner did and said, the decision the reply must make, one reference reply, and a checklist of yes-or-no lines that define a good reply. The 76 cases cover 12 standards in grade 4 fractions and grade 5 place value and decimals (Tables 1 and 7). The model sees the system message “*You are a mathematics tutor. A student is working on the question below. Reply to the student directly, as you would in the conversation.*”, then a user message with the question and the learner’s work, and replies once. Appendix B gives two cases in full.

Table 1. SENSUS Bench v1.0 and SENSUS SFT v1.1 at a glance.

Item	Value
Evaluation cases	76
Common Core standards	12 (7 in grade 4, 5 in grade 5)
Checklist lines	549 in total; 7 to 8 per case (mean 7.2)
Lines of each kind per case	one correctness, one hedging (B2), one next-step (B5), one word-count, and 3 or 4 on the case content (B1, B3, B4)
Serious errors	218 in total; 2 to 3 per case; every case includes a diagnosis stated as certain
Mean words: question / learner description / reference reply	40 / 36 / 74
Grader-agreement thirds	24 / 26 / 26
Training conversations	1,015 over 884 distinct questions (131 questions appear twice, with different learner work)
Mean words: training user message / training reply	92 / 71 (longest reply 109)
Overlap between training and evaluation questions	0 exact; 26 of 76 evaluation questions share any word 8-gram with a training question

3.2 Checklists and Serious Errors

Every case has 7 to 8 binary checklist lines. The first line requires that the reply say nothing false. One line asks for a hedged diagnosis (B2), one for a next step (B5), and one for about 120 words or fewer, following the advice to give feedback in manageable units (Shute, 2008). The remaining lines are about the case itself. A line may require naming the error of comparing numerators and denominators as separate whole numbers, for example, or showing that $4/10$ is less than $3/6$. Each case also lists 2 to 3 serious errors. Every case counts a diagnosis stated as certain as a serious error, and about two thirds also list endorsing a wrong answer. The word limit and the certainty rule are design choices that encode brevity and B2, and Section 7 reports results without them. A listed serious error and a false statement are scored differently. A reply can state something false without triggering any listed serious error; it then loses the correctness line but keeps partial credit, as Case 1 in Appendix B shows. We therefore report correctness separately and test a stricter rule in Section 7. Table 2 summarises what the checklist rewards.

Table 2. What the checklist rewards. Each behaviour with its definition, a real checklist line from the benchmark, how many lines of that kind each case has, and the main research it draws on (Section 3.3).

Behaviour	Definition	Example checklist line	Lines per case	Research basis
Correctness	The reply says nothing false.	Nothing false (for example, saying $4/10$ is greater than $3/6$, or saying $3/6 = 3/10$ is true).	One per case (the first line)	Kluger and DeNisi, 1996
B1 Name a likely error	Names an error tied to what the learner wrote.	Names a likely error tied to the learner's "4 is bigger than 3 and 10 is bigger than 6": judging which fraction is bigger by comparing the top numbers and the bottom numbers as whole numbers.	Part of the 3 or 4 case-content lines	Brown and Burton, 1978; Graesser et al., 1995; Durkin and Rittle-Johnson, 2012
B2 Hedge the diagnosis	Presents the cause of the error as likely, not certain.	Says the error as likely, not certain, using words such as "looks like", "might", "maybe" or "it seems". A reply that names no reason for the error is marked no.	One per case	Brown and Burton, 1978; Black and Wiliam, 1998; Kluger and DeNisi, 1996
B3 Credit what is correct	Names something specific the learner did right.	Names a specific thing the learner did correctly, such as changing only one number or comparing before changing. "Good job" alone is marked no.	Part of the case-content lines	Leatham et al., 2015; Smith et al., 1993; Hattie and Timperley, 2007
B4 Explain or model the idea	Explains the mathematics the error concerns.	Gives a reason a grade 4 learner can check for why $4/10$ is less than $3/6$. Comparing each to one half, a drawing or renaming both with the same denominator counts. Only stating which is bigger does not count.	Part of the case-content lines	Rittle-Johnson et al., 2001; Siegler et al., 2010
B5 End with a next step	Ends with a question or task the learner can act on.	Ends with a question or task the learner can answer next. A trailing "Does that help?" alone does not count.	One per case	Hattie and Timperley, 2007; Chi et al., 1994
Length	About 120 words or fewer.	Is about 120 words or fewer.	One per case	Shute, 2008
Serious error	An error that sets the score to zero, such as stating the diagnosis as certain or endorsing a wrong answer.	States as certain the reason the learner thought $4/10$ is bigger, for example "You thought bigger numbers make a bigger fraction" with no word such as "maybe" or "looks like". Repeating what the learner wrote does not count.	2 or 3 per case	Kluger and DeNisi, 1996

3.3 Why the Checklist Asks for These Behaviours

We wrote the behaviours, and the checklist lines that test them, from research on feedback, mathematical misconceptions and responsive teaching. Mathematics education describes a teacher's response to a student's work as professional noticing: attending to the student's strategy, interpreting what it shows about their understanding, and deciding how to respond (Jacobs et al., 2010). B1 and B2 are the interpreting step, and B3 to B5 are the response. The checklist keeps only what can be seen in one written reply. Table 2 lists the main sources for each element.

Naming the likely error (B1). Errors in fractions and decimals tend to be systematic (Section 1). Brown and Burton (1978) modelled such errors as consistent faults, or bugs, in an otherwise working procedure, which is why the cause of an error can often be read from the work. Tutors do not always do this: in two samples of one-to-one sessions with non-expert tutors, the tutors worked from a rough sense of what the student knew and rarely diagnosed specific errors (Graesser et al., 1995). Research also suggests that studying and comparing incorrect and correct examples can support learning: students who compared incorrect and correct decimal examples, with prompts, gained more understanding and kept fewer misconceptions than students who studied correct examples only (Durkin and Rittle-Johnson, 2012). That study tested worked examples, not a tutor naming a learner's error; B1 applies its motivation by asking the reply to identify the specific likely error, tied to what the learner wrote, rather than restate the correct rule.

Hedging the diagnosis (B2). A wrong answer does not settle its cause, since different faulty procedures can produce the same answer (Brown and Burton, 1978). A diagnosis drawn from one written explanation is an inference, and B2 asks the reply to present it as one (“it looks like you may be...”). Stating the cause tentatively lets the learner confirm or correct it, which keeps the exchange formative (Black and Wiliam, 1998). It also keeps the feedback on the task rather than on the learner; in a large review of feedback studies, feedback that drew attention to the self tended to lower performance (Kluger and DeNisi, 1996). We know of no study showing that hedged wording itself improves learning. B2 is our design choice built on these findings, and Section 7 reports scores without it.

Crediting what is correct (B3). Productive teaching builds on what the student already thinks rather than replacing it (Leatham et al., 2015), and misconceptions usually grow out of knowledge that is partly right (Smith et al., 1993). Feedback is most useful when it addresses the task and the process the student used (Hattie and Timperley, 2007). B3 lines therefore require a specific correct step, such as the learner's correct renaming of a fraction; a general “Good job” is marked no.

Explaining the idea (B4). Conceptual understanding and procedural skill develop together, each supporting the other, as shown with grade 5 and 6 students learning decimals (Rittle-Johnson et al., 2001). The U.S. Department of Education's practice guide on fractions recommends measurement representations such as number lines and explaining why procedures work (Siegler et al., 2010). B4 lines ask for a reason a grade 4 or 5 learner can check, such as comparing to one half, a drawing or a common denominator. Stating which number is bigger does not count.

Ending with a next step (B5) and keeping it short. Feedback should tell the learner where to go next (Hattie and Timperley, 2007), and prompting learners to explain their own reasoning improves understanding (Chi et al., 1994). B5 asks the reply to end with a question or task the learner can answer; a trailing “Does that help?” alone does not count. The length line follows the advice to give feedback in manageable units (Shute, 2008).

Correctness and serious errors. Feedback does not help automatically. In the same review, more than a third of feedback interventions lowered performance (Kluger and DeNisi, 1996), which is why the checklist treats some replies as failures rather than weaker successes. A reply that says something false fails the correctness line. A reply that endorses a wrong answer or states the diagnosis as certain commits a serious error and scores zero. The zero is our scoring choice, not something the literature establishes: we treat confirming a learner's error and asserting a guess about their thinking as the two moves most likely to mislead them. Under this rule a confident diagnosis scores zero while some false mathematical statements keep partial credit, and the relative severity of the two has not been validated. Section 7 rescores the replies without the certainty rule and with a stricter correctness rule.

This research comes mostly from human teachers working with students over time. The checklist measures properties of a single written reply. Within a case, its lines count equally when no serious error is flagged, and a serious-error flag overrides them and sets the score to zero. This study does not test whether a reply that meets it helps a learner learn more (Section 10).

3.4 Metrics

For a reply to case c with n_c checklist lines, let m_c be the number of lines that both graders mark as met. The reply's score is

$\text{score}_c = 0$ if either grader marks any serious error, and m_c / n_c otherwise.

We report four quantities for each model. **Mean score** is the mean of score_c over the 76 cases, so every case counts equally whether it has seven or eight lines. **Full passes** counts cases with $\text{score}_c = 1$, meaning every line met and no serious error. **Serious errors** counts replies in which either grader flags at least one listed serious error. **No false statement** is the share of replies whose correctness line both graders mark as met. For pairwise preference judgments, **net preference** is $(\text{wins} - \text{losses}) / \text{resolved comparisons}$, where ties count as resolved. The reference reply for each case scores 0.991. That shows the checklists can be satisfied. The references were prepared alongside the checklists, so this is not independent evidence that the checklists are valid.

4. Training Data and Fine-Tuning

4.1 SENSUS SFT

SENSUS SFT v1.1 has 1,015 synthetic conversations over 884 distinct questions. Each conversation has the benchmark's system message, a user message with the question and the learner's work, and the reference reply as the assistant turn. 98% of the replies contain a hedging phrase, 97% end with a question, and none is longer than 109 words (Table 13). Cases are kept in groups by question, and each group goes wholly to training or wholly to evaluation. No evaluation question matches a training question exactly, though our checks do not rule out near-duplicates (Section 4.3). The benchmark and the training set still share their format, standards and kinds of misconception by design, so SENSUS measures in-distribution learning of a behaviour.

4.2 Generic Control

A control set answers the same prompts with 1,012 correct, clear, generic tutoring replies that never diagnose the learner. Its replies are short like the SENSUS replies, so training on it shows what fine-tuning on tutor-shaped text achieves without the diagnosis. Because it leaves out B1 and B2, generic training also pushes those behaviours below the untuned base (Figure 4). The control differs from SENSUS in several behaviours at once, so the comparison shows the effect of the whole SENSUS policy and not of any single behaviour. We therefore report the comparison with the untuned base alongside the comparison with the control.

4.3 Overlap with the Benchmark

No benchmark question appears in the training file (0 exact matches after normalizing case and punctuation). Questions in this domain share stock phrasing. 26 of 76 benchmark questions share at least one word 8-gram with some training question, and the largest share of one question's 8-grams found in a single training question is 33%. We compared questions only. Learner descriptions, replies and semantic similarity were not checked, so these results do not rule out near-duplicates in the full examples, which falls short of the contamination checks Sainz et al. (2023) recommend.

4.4 Recipe

Each model is trained with supervised fine-tuning on the conversations. It reads the instruction, the question and the learner's work, and learns to produce the reference reply. Training changes a rank-32 low-rank adapter beside the base model for eight passes over the data at a learning rate of 0.0003 (Table 3).

Table 3. Training and inference settings for the fine-tuned models.

Setting	Value
Base models	Qwen3-8B, Qwen3-14B, Qwen3.8-27B (the 27B base is a newer model generation)
Method	Supervised fine-tuning of a low-rank adapter (LoRA) on a hosted service (Fireworks); the model learns to produce the reference reply for each conversation
Adapter rank, epochs, learning rate	32, 8, 0.0003
Training rows	1,015 SENSUS conversations, or 1,012 generic replies to the same prompts
Runs	Two per SENSUS recipe at each size; one per generic control
Batch size, adapter alpha and dropout, target modules, maximum length, learning-rate schedule, loss masking	Service defaults, not recorded
Inference	Temperature 0, thinking disabled, at most 900 output tokens
Version of the data used	SENSUS SFT v1.0; v1.1 differs only in a topic label on the first line of some user messages

Eight epochs on about a thousand rows is a strong setting, and the fine-tuned replies are formulaic as a result. 97% of the 27B model's replies contain a hedging phrase, and a single two-word opening begins 24% of them. This is intended, but it is also a reason to check that the models keep their general abilities, which we did not measure.

5. Evaluation Setup

5.1 Models

We evaluate the fine-tuned models, the untuned bases, the generic controls and four general-purpose models (Table 4). Every model receives the same system and user messages. None sees the checklist, and none is told about B1 to B5. Each general-purpose model was evaluated once. Inference settings are not identical across model families (Table 4), and those differences, together with differences in model generation, limit comparisons across families. Unless stated otherwise, fine-tuned results use the second, latest completed run of each recipe.

Table 4. Evaluated models. Every model received the same system and user messages and replied once per case; reasoning text, where produced, was removed before grading.

Model	Kind	Training	Sampling	Graders
Qwen3-8B, Qwen3-14B, Qwen3.8-27B + SENSUS	Open, 8B to 27B	SENSUS SFT (Table 3)	Temperature 0, thinking off, 900 tokens	Opus 5.5 and GPT-6.1 Sol
Same bases + generic replies	Open, 8B to 27B	Generic SFT, same recipe	Temperature 0, thinking off, 900 tokens	Opus 5.5 and GPT-6.1 Sol
Same bases, untuned	Open, 8B to 27B	None	Temperature 0, thinking off, 900 tokens	Opus 5.5 and GPT-6.1 Sol
GPT-6 Sol	Hosted, size not disclosed	None	Default reasoning, 3,000 completion tokens	Opus 5.5 and GPT-6.1 Sol
Opus 5.5	Hosted, size not disclosed	None	Provider defaults, 4,000 tokens	Alternate first grader and GPT-6.1 Sol
Nemotron 3 Ultra (NVIDIA)	Open weights, served by a host	None	Temperature 0, 4,000 tokens	Opus 5.5 and GPT-6.1 Sol
Kimi K3	Open weights, served by a host	None	Temperature 0, 4,000 tokens	Opus 5.5 and GPT-6.1 Sol

5.2 Graders

Grader 1 is Opus 5.5 and grader 2 is GPT-6.1 Sol at medium reasoning effort. Each grader works alone. It sees the case as the model saw it, the checklist, the serious errors and one reply, with no indication of which model wrote it, and marks every line true or false (Appendix A). Each reply was graded once. Two overlaps between graders and graded models remain. Grader 2 comes from the same provider as GPT-6 Sol. For replies written by Opus 5.5, an alternate first grader from the same provider replaced Opus 5.5, which reduces the overlap without removing it. Section 8.2 reports each grader separately.

5.3 Statistics

Intervals on means are 95% percentile bootstrap intervals with 10,000 resamples (Efron and Tibshirani, 1993), and intervals on pass rates are Wilson intervals. Differences between models resample cases jointly. Because cases cluster by standard, Table 6 also resamples whole standards. Case-level wins and losses use a two-sided exact sign test that excludes ties, with Holm correction across the comparisons in that table (Holm, 1979; Dror et al., 2018). Grader agreement is reported as raw agreement and Cohen's kappa (Cohen, 1960). These intervals cover sampling over cases only, and they do not account for the recipe having been adapted to these cases (Section 6.3). They leave out variation between training runs, which was 0.014 to 0.028 between the two runs of each recipe, and variation in the graders. With 76 cases, only large differences are reliable (Card et al., 2020). Each run's questions, model configurations and decision rules were recorded before it started, and the 27B experiment named 27B against 14B as its primary comparison. We call this the pre-run primary comparison. It was recorded internally and not in a public registry, and because the cases had been used in earlier rounds, recording it does not make them a fresh test.

6. Results

6.1 Main Results

All three fine-tuned models score above every general-purpose model (Table 5 and Figure 2). Qwen3.8-27B+SENSUS fully passes 24 of 76 cases and has no serious error, while no general-purpose model fully passes any case. GPT-6 Sol writes the shortest general-purpose replies (54 words on average) and makes no false statements. Opus 5.5, Nemotron 3 Ultra and Kimi K3 average 178 to 225 words and draw 18 to 28 serious-error flags each, almost all for a diagnosis stated as certain.

Table 5. All three fine-tuned models lead, and the 27B model fully passes a third of the cases with no serious error. Results on the 76 cases: mean checklist score with a 95% case-level bootstrap interval, full passes with a Wilson 95% interval on the pass rate, replies with a serious error, share of replies both graders mark free of false statements, and mean reply length in words. Fine-tuned rows are the second training run of each recipe. *Graded with an alternate first grader from the same provider (Section 5).

Model	Mean score	95% CI	Full passes [95% CI]	Serious errors	No false statement	Words
Qwen3.8-27B+SENSUS	0.838	[0.804, 0.871]	24/76 [22%, 43%]	0/76	91%	72
Qwen3-14B+SENSUS	0.720	[0.667, 0.769]	14/76 [11%, 29%]	3/76	70%	77
Qwen3-8B+SENSUS	0.626	[0.558, 0.692]	7/76 [5%, 18%]	11/76	49%	74
GPT-6 Sol	0.525	[0.487, 0.562]	0/76 [0%, 5%]	4/76	100%	54
Opus 5.5*	0.461	[0.398, 0.520]	0/76 [0%, 5%]	18/76	96%	225
Nemotron 3 Ultra	0.356	[0.302, 0.409]	0/76 [0%, 5%]	19/76	89%	178
Kimi K3	0.331	[0.269, 0.392]	0/76 [0%, 5%]	28/76	91%	214
Qwen3.8-27B (untuned)	0.290	[0.237, 0.344]	0/76 [0%, 5%]	26/76	75%	216

Figure 3 shows the score of every case, not only the means.

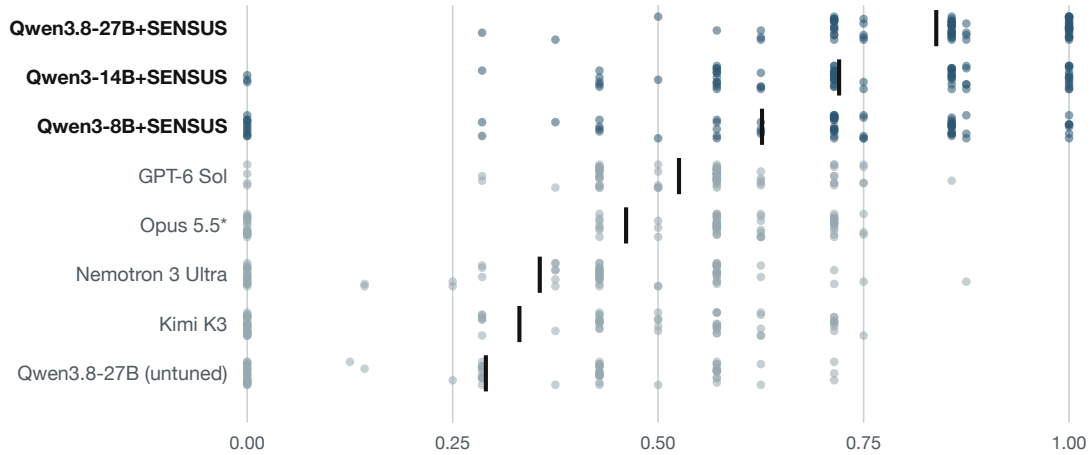


Figure 3. The gain is broad, not driven by a few cases. Each dot is one case’s score and the black bar is the mean. Serious errors put many general-purpose and untuned replies at zero (the untuned 27B base scores zero on 26 cases, the fine-tuned model on 0).

Against GPT-6 Sol, the fine-tuned 27B model wins 68 / 3 / 5 cases (wins, losses, ties), a mean difference of +0.313 [+0.263, +0.363]. Every comparison in Table 6 holds after Holm correction. The model scores above every general-purpose model on each of the 12 standards (Table 7) and in each third of the cases ranked by grader agreement (Table 8).

Table 6. The fine-tuned 27B model wins at least 68 of 76 cases against every comparison model. Qwen3.8-27B+SENSUS against each comparison model on the same cases: mean per-case score difference; 95% paired bootstrap interval over cases and over the 12 standards (with only 12 clusters, the second interval is a rough check, not a more conservative one); cases won, lost and tied; two-sided exact sign test (ties excluded) with Holm correction across the five comparisons.

Comparison model	Difference	95% CI (cases)	95% CI (standards)	Wins / losses / ties	Holm-adjusted p
GPT-6 Sol	+0.313	[+0.263, +0.363]	[+0.261, +0.365]	68 / 3 / 5	< 0.001
Opus 5.5*	+0.377	[+0.307, +0.451]	[+0.322, +0.435]	68 / 4 / 4	< 0.001
Nemotron 3 Ultra	+0.482	[+0.421, +0.547]	[+0.426, +0.543]	72 / 3 / 1	< 0.001
Kimi K3	+0.507	[+0.433, +0.582]	[+0.449, +0.561]	68 / 3 / 5	< 0.001
Qwen3.8-27B (untuned)	+0.548	[+0.485, +0.609]	[+0.500, +0.589]	73 / 1 / 2	< 0.001

Table 7. Coverage and mean score by Common Core standard: evaluation cases, training conversations, the fine-tuned 27B model, the best general-purpose model on that standard, and the untuned 27B base. Standards with three or four cases are descriptive only.

Standard	Topic	Cases	Training rows	27B+SENSUS	Best general-purpose	27B untuned
4.NF.A.1	Equivalent fractions	3	71	0.786	0.446 (Kimi K3)	0.232
4.NF.A.2	Comparing fractions	10	82	0.866	0.541 (GPT-6 Sol)	0.286
4.NF.B.3	Adding and subtracting fractions	9	123	0.794	0.569 (GPT-6 Sol)	0.214
4.NF.B.4	Multiplying a fraction by a whole number	7	88	0.847	0.490 (GPT-6 Sol)	0.319
4.NF.C.5	Tenths and hundredths	8	77	0.893	0.429 (GPT-6 Sol)	0.286
4.NF.C.6	Decimal notation	6	78	0.833	0.536 (GPT-6 Sol)	0.238
4.NF.C.7	Comparing decimals	8	73	0.893	0.482 (GPT-6 Sol)	0.411
5.NBT.A.1	Place value	3	25	0.768	0.595 (Opus 5.5)	0.286
5.NBT.A.2	Powers of ten	6	91	0.833	0.595 (GPT-6 Sol)	0.238
5.NBT.A.3	Reading and comparing decimals	4	81	0.835	0.487 (GPT-6 Sol)	0.442
5.NBT.A.4	Rounding decimals	6	75	0.795	0.643 (GPT-6 Sol)	0.408
5.NBT.B.7	Decimal operations	6	151	0.821	0.545 (GPT-6 Sol)	0.143

Table 8. Mean score by grader-agreement third. Cases are ranked by how often the two graders disagreed on their lines across all eight models' replies, so the bands describe the measurement, not task difficulty.

Band	Cases	27B+SENSUS	GPT-6 Sol	Opus 5.5*	27B untuned
Graders agree most	24	0.821	0.548	0.468	0.260
Middle third	26	0.832	0.521	0.473	0.291
Graders agree least	26	0.861	0.509	0.442	0.318

6.2 Training Effects

Fine-tuning on SENSUS raises Qwen3.8-27B from 0.290 to 0.838, a gain pooled over both runs of +0.534 [+0.476, +0.591]. The two 27B runs score 0.811 and 0.838, a mean of 0.825. Against the generic control the pooled gain is +0.379 [+0.338, +0.421] at 27B and +0.323 [+0.272, +0.374] at 14B, so the SENSUS replies as a whole teach far more than generic tutor replies do (Tables 9 and 10). The 27B model also became more accurate on these cases. The share of replies both graders marked free of false statements rose from 75% untuned to 91% for the second SENSUS run, and replies with a listed serious error fell from 26 of 76 to 0 (Tables 5 and 19). The generic control also removes most serious errors (1 of 76), so that part of the improvement comes partly from fine-tuning on short tutor replies in general, while the reduction in false statements is measured against the model's own untuned base. The pre-run primary comparison of 27B against 14B is +0.111 [+0.070, +0.152]. The 27B base is also a newer generation, so this difference reflects the base model as a whole and not only its size.

Table 9. Training comparisons pooled over both SENSUS training runs, with 95% bootstrap intervals over cases. The generic control answers the same prompts without diagnosing the learner (Section 4.2); the comparison with the untuned base is reported alongside it.

Comparison	Difference	95% CI
27B+SENSUS minus generic	+0.379	[+0.338, +0.421]
27B+SENSUS minus 14B+SENSUS, pre-run primary	+0.111	[+0.070, +0.152]
27B+SENSUS minus untuned	+0.534	[+0.476, +0.591]
14B+SENSUS minus generic	+0.323	[+0.272, +0.374]
14B+SENSUS minus 8B+SENSUS	+0.076	[+0.019, +0.136]

Table 10. Every recorded training condition. Each SENSUS recipe was trained twice; untuned and generic conditions were evaluated once.

Base model	Condition	Mean score	Full passes	Serious errors
Qwen3-8B	Untuned	0.239	0/76	18/76
Qwen3-8B	Generic SFT (1,012 rows)	0.385	0/76	7/76
Qwen3-8B	SENSUS SFT (1,015 rows), run 1	0.647	8/76	7/76
Qwen3-8B	SENSUS SFT (1,015 rows), run 2	0.626	7/76	11/76
Qwen3-14B	Untuned	0.278	0/76	17/76
Qwen3-14B	Generic SFT (1,012 rows)	0.390	0/76	5/76
Qwen3-14B	SENSUS SFT (1,015 rows), run 1	0.706	15/76	4/76
Qwen3-14B	SENSUS SFT (1,015 rows), run 2	0.720	14/76	3/76
Qwen3.8-27B	Untuned	0.290	0/76	26/76
Qwen3.8-27B	Generic SFT (1,012 rows)	0.445	0/76	1/76
Qwen3.8-27B	SENSUS SFT (1,015 rows), run 1	0.811	19/76	1/76
Qwen3.8-27B	SENSUS SFT (1,015 rows), run 2	0.838	24/76	0/76

Figure 4 shows which behaviours changed. At 27B the largest gains are in naming the error (B1, 0.564 to 0.846) and hedging it (B2, 0.304 to 0.913), the two failures SENSUS targets. The table also shows what the general-purpose models do well. Opus 5.5 and Kimi K3 credit the learner's work and explain the idea more often than the fine-tuned model, and fall behind on naming the error, hedging it and ending with a next step.

	B1 Name error	B2 Hedge	B3 Credit	B4 Explain	B5 Next step
Qwen3.8-27B+SENSUS	0.85	0.91	0.73	0.78	0.75
Qwen3-14B+SENSUS	0.85	0.83	0.68	0.66	0.61
Qwen3-8B+SENSUS	0.67	0.82	0.69	0.62	0.58
GPT-6 Sol	0.41	0.23	0.68	0.66	0.04
Opus 5.5*	0.72	0.41	0.85	0.95	0.49
Nemotron 3 Ultra	0.46	0.31	0.71	0.78	0.19
Kimi K3	0.61	0.39	0.75	0.93	0.29
Qwen3.8-27B, generic SFT	0.05	0.01	0.25	0.59	0.42
Qwen3.8-27B, untuned	0.56	0.30	0.69	0.62	0.14

Figure 4. Fine-tuning moves exactly the behaviours SENSUS targets. Share of checklist lines met for each behaviour. The fine-tuned models lead on naming the error (B1), hedging (B2) and the next step (B5); Opus 5.5 and Kimi K3 lead on crediting the learner (B3) and explaining the idea (B4). Generic fine-tuning removes B1 and B2 almost entirely. *Alternate first grader.

6.3 Development History

The released recipe was settled over several development rounds on Qwen3-8B, and each round was scored on the same 76 cases (Table 11). The first fine-tuned model, trained for three epochs at rank 16, scored 0.280. The final settings raised the same 502 rows to 0.580. Adding rows helped up to the released 1,015 but not beyond (Figure 5). 884 rows scored 0.578, and an extended 1,232-row set scored 0.643 at 8B and 0.814 and 0.809 at 27B, no better than the released set. Five SENSUS examples in the prompt, with no training, scored 0.312, a small gain over the untuned 0.239 next to what fine-tuning achieved. Because the evaluation cases guided these choices, the main results are development results (Section 10).

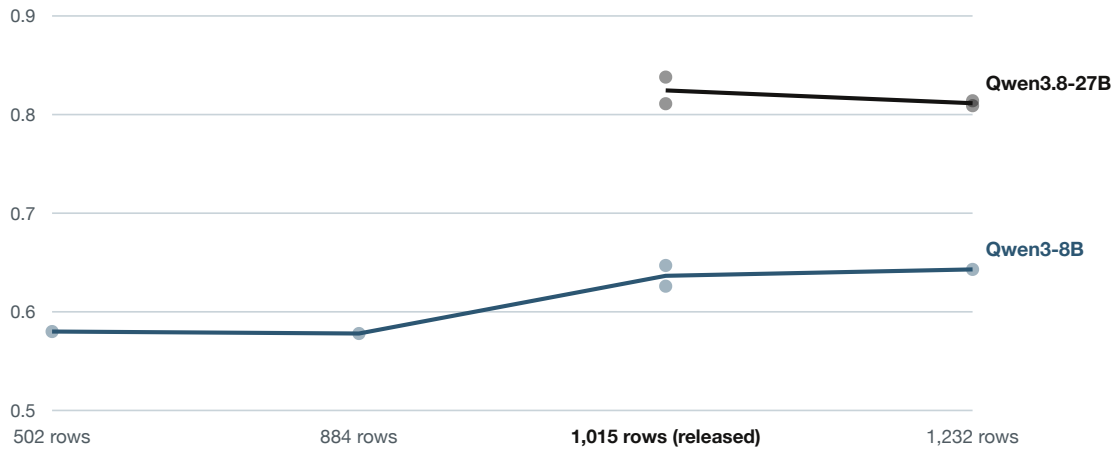


Figure 5. More rows helped up to the released set and then levelled off. Mean score by number of SENSUS training rows during development, with one dot per training run and a line through the means. All points were scored on the same 76 cases, so this is development history, not a held-out result.

Table 11. Development conditions recorded on the same 76 cases before and after the released recipe was fixed, in the order they were run.

Stage	Condition	Mean score	Full passes	Serious errors
Round 2	Qwen3-8B, 502 SENSUS rows, early settings (rank 16, 3 epochs)	0.280	0/76	27/76
Round 2	Qwen3-8B, 501 generic rows, early settings	0.311	0/76	8/76
Round 3	Qwen3-8B, five SENSUS examples in the prompt, no training	0.312	0/76	12/76
Round 3	Qwen3-8B, 502 SENSUS rows	0.580	1/76	11/76
Round 3	Qwen3-8B, 501 generic rows	0.348	0/76	10/76
Round 4	Qwen3-8B, 884 SENSUS rows	0.578	3/76	12/76
Round 4	Qwen3-8B, 1,015 SENSUS rows (released set), run 1	0.647	8/76	7/76
Round 4	Qwen3-8B, 1,012 generic rows	0.385	0/76	7/76
Round 5	Qwen3-8B, 1,015 SENSUS rows, run 2	0.626	7/76	11/76
Round 5	Qwen3-8B, 1,232 SENSUS rows (extended set)	0.643	11/76	9/76
After the 27B runs	Qwen3.8-27B, 1,232 SENSUS rows (extended set), run 1	0.814	23/76	2/76
After the 27B runs	Qwen3.8-27B, 1,232 SENSUS rows (extended set), run 2	0.809	22/76	4/76

7. What the Gain Consists Of

The training replies and the checklist share a style, and the general-purpose models were never asked for it. To separate that style from mathematical content, we rescored every recorded mark under the rules in Table 12. No reply was generated or graded again. Line kinds come from each line's wording. Every case has one correctness line, one word-count line, one hedging line and one next-step line, and the other 245 lines concern the content of the case.

Table 12. Removing any single style rule leaves the ranking nearly unchanged; removing all of them together brings Opus 5.5 level. The recorded marks rescored under alternative rules. Each row removes checklist lines or serious errors of one kind, or adds a stricter correctness rule; the last column is the paired difference between Qwen3.8-27B+SENSUS and Opus 5.5 with its 95% interval. Every case has one correctness, one word-count, one hedging and one next-step line, identified from the line’s wording; the remaining 245 lines concern the content of the case.

Scoring rule	27B+SENSUS	14B+SENSUS	8B+SENSUS	GPT-6 Sol	Opus 5.5*	Nemotron 3 Ultra	Kimi K3	27B untuned	27B+SENSUS minus Opus 5.5	Kendall τ with published ranking
Published rule	0.838	0.720	0.626	0.525	0.461	0.356	0.331	0.290	+0.377 [+0.307, +0.451]	1.00
Any false statement counts as a serious error	0.783	0.558	0.400	0.525	0.446	0.326	0.315	0.265	+0.337 [+0.239, +0.434]	0.86
Without the word-count line	0.813	0.685	0.590	0.457	0.533	0.390	0.382	0.327	+0.279 [+0.198, +0.363]	0.93
Without the hedging lines	0.819	0.690	0.600	0.608	0.524	0.398	0.376	0.323	+0.295 [+0.216, +0.379]	0.93
Without the “stated as certain” serious error	0.838	0.720	0.626	0.558	0.593	0.479	0.523	0.405	+0.245 [+0.205, +0.284]	0.86
Without word count, hedging and “stated as certain”	0.785	0.642	0.551	0.577	0.807	0.609	0.708	0.529	−0.021 [−0.072, +0.029]	0.21
Also without the next-step line	0.797	0.663	0.566	0.705	0.895	0.712	0.814	0.628	−0.098 [−0.151, −0.045]	−0.07

Removing any one style rule leaves the fine-tuned 27B model well ahead. Its lead over Opus 5.5 is +0.279 [+0.198, +0.363] without the word-count line, +0.295 without the hedging lines and +0.245 without the certainty rule. Removing all three together brings Opus 5.5 level (−0.021 [−0.072, +0.029]), while the 27B model stays ahead of GPT-6 Sol by +0.208 [+0.155, +0.261]. Also removing the next-step line leaves mostly correctness and case content. On those lines Opus 5.5 leads (difference −0.098 [−0.151, −0.045]), Kimi K3 scores 0.814 against 0.797, and the 27B model still leads GPT-6 Sol by +0.092 [+0.028, +0.155].

Most of the gap between the fine-tuned and general-purpose models therefore lies in B2, B5 and brevity. The benchmark targets these on purpose, and the general-purpose models were not asked to produce them. On content lines a 27B open model trained on about a thousand examples trails Opus 5.5, which explains more and credits more of the learner’s work (Figure 4), and stays ahead of GPT-6 Sol. This fits the view that small supervised datasets mainly teach a model how to respond (Zhou et al., 2023; Gudibande et al., 2024). It leaves open how much of the behavioural gap a general-purpose model would close if prompted to show B1 to B5. Table 13 shows the style difference directly.

Table 13. Surface markers of the target style: share of replies containing a hedging phrase (such as “you may”, “it looks like” or “maybe”), share ending with a question mark, and mean and maximum length in words.

Replies	Hedging phrase	Ends with “?”	Mean words	Max words
SENSUS training replies	98%	97%	71	109
Qwen3.8-27B+SENSUS	97%	97%	72	94
Qwen3-14B+SENSUS	97%	99%	77	131
Qwen3-8B+SENSUS	100%	95%	74	119
GPT-6 Sol	0%	5%	54	86
Opus 5.5*	3%	24%	225	310
Nemotron 3 Ultra	4%	37%	178	390
Kimi K3	1%	41%	214	320
Qwen3.8-27B (untuned)	5%	14%	216	480

8. Measurement Checks

8.1 Agreement Between Graders

Before marks are combined, the graders agree on 94.7% of 4,392 line judgments ($\kappa = 0.89$) and on 97.2% of serious-error judgments ($\kappa = 0.72$, lower because serious errors are rare). Agreement varies by line kind (Table 14). It is highest on the word-count line, as expected. Agreement beyond chance is lowest on the correctness line ($\kappa = 0.79$), the line that matters most for learners. Table 15 gives the full confusion matrix. Taking grader 1 as the reference, grader 2 marks a line as met with 95.5% precision and 96.2% recall. These figures measure agreement between two models, not accuracy, and the 4,392 judgments are nested within replies and cases, so they are not independent observations.

Table 14. The two graders agree on about 95% of judgments; agreement beyond chance is lowest on correctness. Agreement between the two graders before marks are combined, over every recorded judgment of all eight models: raw agreement and Cohen’s kappa, overall and by line kind.

Judgments	Count	Agreement	Cohen’s κ
All checklist lines	4,392	94.7%	0.89
Correctness line	608	94.7%	0.79
Case-content lines	1,960	93.5%	0.83
Hedging lines	608	95.7%	0.91
Next-step lines	608	93.8%	0.87
Word-count lines	608	98.7%	0.97
Serious-error lines	1,744	97.2%	0.72

Table 15. All 4,392 checklist-line judgments, grader 1 against grader 2. Taking grader 1 as the reference, grader 2 has accuracy 94.7%, precision 95.5%, recall 96.2% and F1 95.9% for “met”.

	Grader 2: met	Grader 2: not met
Grader 1: met	2,683	106
Grader 1: not met	126	1,477

8.2 Each Grader Alone

The fine-tuned models rank above GPT-6 Sol under either grader (Table 16). Requiring both graders to agree costs the long-reply models more. It lowers Opus 5.5 by 0.064 and Kimi K3 by 0.068, against 0.011 for the fine-tuned 27B model. Grader 1 is more generous than grader 2 to Opus 5.5 (0.575 against 0.474). For Opus 5.5, grader 1 is the alternate grader from its own provider, so the gap may reflect same-provider leniency. Grader 1 is also more generous to Kimi K3 and Nemotron 3 Ultra but not to the untuned 27B model, so the pattern is mixed. GPT-6 Sol scores about the same under the grader from its own provider (0.565) as under the other (0.552).

Table 16. Scores under each grader alone and under the combined rule. The cost of the combined rule is the mean of the two single-grader scores minus the combined score. *Grader 1 for Opus 5.5 is the alternate first grader.

Model	Combined	Grader 1 only	Grader 2 only	Cost of combining	Cohen’s κ
Qwen3.8-27B+SENSUS	0.838	0.846	0.853	0.011	0.92
Qwen3-14B+SENSUS	0.720	0.748	0.753	0.030	0.82
Qwen3-8B+SENSUS	0.626	0.637	0.657	0.021	0.86
GPT-6 Sol	0.525	0.552	0.565	0.033	0.90
Opus 5.5*	0.461	0.575	0.474	0.064	0.91
Nemotron 3 Ultra	0.356	0.436	0.391	0.058	0.86
Kimi K3	0.331	0.433	0.366	0.068	0.88
Qwen3.8-27B (untuned)	0.290	0.327	0.366	0.056	0.89

8.3 Open Preference

A checklist measures named properties, so we also compared the fine-tuned 27B model with each general-purpose model in blind pairwise judgments without a checklist. These used 142 development cases, separate from the benchmark (Table 17). Judges preferred the fine-tuned model over GPT-6 Sol and Nemotron 3 Ultra. Against Kimi K3 the two judges disagreed in sign, and the combined result is near zero. Judges preferred Opus 5.5 overall, but the preference came mostly from the judge from the same provider as Opus 5.5 (-0.718 for the fine-tuned model), while GPT-6.1 Sol was close to even ($+0.049$). The independent third judge also preferred Opus 5.5. On these cases 13 Opus 5.5 replies failed the automatic code checks, against 4 for the fine-tuned model. On the benchmark cases, giving the judges the checklist reverses the Opus 5.5 comparison in favor of the fine-tuned model (Table 18). When Opus 5.5 was told to keep to 120 words, open preference for it became stronger, so reply length does not explain why judges favor it. An open judgment and the SENSUS checklist reward different things, and we do not yet know why they diverge.

Table 17. Blind pairwise preference on 142 development cases, judged without a checklist. Positive values favor Qwen3.8-27B+SENSUS (first training run). Net preference is wins minus losses over resolved cases; the single-judge columns use all compared cases as the denominator. Judge 1 is Opus 5.5 (for the Opus 5.5 comparison, the alternate grader from its provider), judge 2 is GPT-6.1 Sol, and the third judge is GLM 5.3. Code checks flag arithmetic that contradicts the answer key, endorsing a wrong answer, or giving the answer when only a hint was asked for.

Comparison model	Resolved	Wins / losses / ties	Net preference	95% CI	Judge 1 / judge 2	Third judge	Words: model / ours	Code-check failures: model / ours
GPT-6 Sol	138/142	78 / 40 / 20	+0.275	[+0.094, +0.457]	+0.275 / +0.268	+0.571	51 / 74	10 / 4
Kimi K3	136/142	42 / 48 / 46	-0.044	[-0.182, +0.098]	-0.246 / +0.261	-0.015	194 / 74	13 / 4
Nemotron 3 Ultra	138/142	91 / 28 / 19	+0.457	[+0.285, +0.615]	+0.408 / +0.542	+0.648	142 / 74	9 / 4
Opus 5.5*	142/142	12 / 63 / 67	-0.359	[-0.469, -0.242]	-0.718 / +0.049	-0.439	207 / 74	13 / 4

Table 18. Pairwise judgments on the 76 cases with and without the case checklist; the 120-word rows use separate Opus 5.5 replies written under a word limit. Positive values favor Qwen3.8-27B+SENSUS.

Comparison	Judge instruction	Resolved	Wins / losses / ties	Net preference	95% CI
GPT-6 Sol	No checklist supplied	76/76	57 / 14 / 5	+0.566	[+0.382, +0.737]
GPT-6 Sol	Case checklist supplied	75/76	64 / 9 / 2	+0.733	[+0.573, +0.867]
Opus 5.5	No checklist supplied	76/76	7 / 36 / 33	-0.382	[-0.526, -0.237]
Opus 5.5	Case checklist supplied	76/76	59 / 10 / 7	+0.645	[+0.474, +0.789]
Opus 5.5, 120-word instruction	No checklist supplied	76/76	2 / 58 / 16	-0.737	[-0.842, -0.618]
Opus 5.5, 120-word instruction	Case checklist supplied	66/76	40 / 13 / 13	+0.409	[+0.212, +0.606]

8.4 Targeted Edits

We edited Opus 5.5 replies to reveal the full answer, or to state a certain diagnosis without a closing question, and asked three model judges to compare each edit with the original. They preferred the original over the answer-revealing edit in 66, 67 and 70 of 76 cases, and over the certain, question-free edit in 69, 71 and 71 cases. The edits change several features at once and are longer than the originals, so this shows that the judges notice the edits, not which single property they respond to.

9. Failure Analysis

Table 19 counts how each model's replies fail. The fine-tuned 27B model's most common failure is missing a content line (55% of replies), then missing the next step (26%) and making a false statement (9%). It never states a diagnosis as certain. The general-purpose models fail in a different pattern. GPT-6 Sol misses the next step in 96% of replies, and Opus 5.5 and Kimi K3 state a diagnosis as certain in 24% and 36% of replies.

Table 19. The fine-tuned and general-purpose models fail in opposite ways: the fine-tuned model misses content, the general-purpose models miss hedging and the next step. Failure modes: share of each model's replies in which both graders did not mark the line as met, or either grader flagged the serious error.

Model	False statement	Diagnosis stated as certain	Other serious error	Missed a content line	Missed hedging (B2)	Missed next step (B5)	Over the word count
Qwen3.8-27B+SENSUS	9%	0%	0%	55%	4%	26%	0%
Qwen3-14B+SENSUS	30%	0%	4%	63%	7%	42%	3%
Qwen3-8B+SENSUS	51%	0%	14%	76%	8%	43%	0%
GPT-6 Sol	0%	5%	0%	83%	99%	96%	0%
Opus 5.5*	4%	24%	0%	38%	93%	57%	99%
Nemotron 3 Ultra	11%	24%	1%	72%	91%	82%	80%
Kimi K3	9%	36%	1%	54%	95%	72%	97%
Qwen3.8-27B (untuned)	25%	29%	9%	72%	91%	87%	91%

Headroom. Table 20 lists the checklist lines that the fewest models meet. Most ask the reply to end with a specific next task tied to the case rather than a general question, which is where even the fine-tuned model still falls short.

Table 20. Where the benchmark still has headroom: the checklist lines met by the fewest of the eight evaluated models, with whether the fine-tuned 27B model met each one.

Standard	Kind	Checklist line	Models meeting it	27B+SENSUS
4.NFB.3	B5	Ends by asking the learner to express the remaining quantity as a fraction; a generic check-in alone does not count.	0 of 8	×
4.NFB.4	B4	States that $4/5$ is less than $24/5$.	0 of 8	×
4.NFC.5	B5	Ends by asking the learner to correct the denominators; a generic comprehension check alone does not count.	0 of 8	×
4.NFC.6	B5	Ends with a place-value task that asks the learner to represent thirty-eight hundredths; a generic check for understanding alone does not count.	0 of 8	×
4.NFC.6	B5	Ends with a task to give the rod count followed by the cube count using the corrected positions; a generic 'Try again' does not count.	0 of 8	×
4.NFC.7	B4	States that $9/10$ is the greatest of the fraction parts.	0 of 8	×
5.NBT.A.1	Content	Says that comparing the size of a thousandth with a tenth is a useful step.	0 of 8	×
5.NBT.A.2	B5	Ends by asking which dimension should stay unchanged; a generic check-in does not count.	0 of 8	×
5.NBT.A.2	B5	Ends with a question or instruction asking the learner to check 700 and 7,000 against the interval; a generic understanding check does not count.	0 of 8	×
5.NBT.A.3	B5	Ends with a specific task to choose the category supported by the contrasting examples; a generic check-in question does not count.	0 of 8	×

Correctness. Both graders mark the reply free of false statements for 91% of the fine-tuned 27B model's replies, 70% at 14B and 49% at 8B, against 100% for GPT-6 Sol and 96% for Opus 5.5. Under the published rule a false statement costs one checklist line. If any false statement instead counts as a serious error (Table 12, second row), the 27B model scores 0.783 and stays first. The 14B model falls to 0.558, close to GPT-6 Sol, and the 8B model to 0.400, below it. At 27B, fine-tuning reduced false statements relative to the untuned base (75% of replies free of them, against 91% after training). The smaller models learned the shape of a good reply but often got its mathematics wrong. Shorter replies may also give fewer chances to make a false statement. Case 1 in Appendix B shows a 27B reply that is diagnostic in form but states that $4/10$ is one half.

10. Limitations

- **No prompted general-purpose baseline.** The general-purpose models were not told about B1 to B5. Section 7 shows that most of their gap lies there, so the advantage over a prompted model is unknown.
- **Model-only, synthetic evaluation.** The cases, learners and training data are synthetic, and every judgment is made by a model. The benchmark measures specified reply properties and not learning outcomes. Agreement beyond chance is lowest on the correctness line.
- **Recipe chosen on the evaluation cases.** The training data and settings were developed over several rounds, each scored on these 76 cases (Section 6.3). The headline results are therefore development results, and a fresh test set is needed to confirm them.
- **In-distribution test.** The benchmark shares its format, standards and kinds of misconception with the training data. This paper does not describe how the data was produced, which limits how completely it can be documented under datasheet practice (Gebru et al., 2021).
- **A small, reused case set.** The 76 cases were excluded from training but examined repeatedly, so they now serve as historical evaluation cases rather than a fresh test set.
- **Accuracy and general ability.** The fine-tuned models make false statements more often than GPT-6 Sol and Opus 5.5, most of all at 8B and 14B, and ability on other tasks was not measured.
- **Training records.** Several training settings were service defaults that were not recorded, and the general-purpose models ran with different provider defaults for reasoning. Two runs per recipe and one per control limit what can be said about variance.

- **Grader overlap.** One grader shares a provider with GPT-6 Sol, and Opus 5.5 was graded by an alternate grader from its own provider.

11. Ethics and Broader Impact

SENSUS concerns software that responds to children's mathematical work. Every learner and situation in it is synthetic, and no personal data is involved. A diagnostic tone can make a wrong reply more convincing, and Section 9 shows that the fine-tuned models are not free of false statements. Hedging expresses uncertainty about a diagnosis; it does not make the diagnosis correct. The dataset card therefore rules out using the data as a sole source of instruction for children without independent review. All evaluation is by models, so the results should not be presented as evidence of learning gains. Showing that would need a study with learners.

12. Release and Reproducibility

SENSUS SFT v1.1 is released under CC BY 4.0 with a dataset card, SHA-256 checksums and an offline verifier. The evaluation kit contains the 76 cases, a sampling script, the grading script with the instruction in Appendix A, and every recorded reply with both graders' marks. Its grading script recomputes every published score offline. The training recipe and the aggregate results are released too, and every statistic in this paper is computed from these records with a fixed random seed. Recomputing scores from the recorded marks is not the same as regenerating replies or retraining the adapters. Several training settings were service defaults that were not recorded, and the generic control set is not released, so the control comparison can be checked but not retrained from public files.

13. Conclusion and Future Work

About a thousand open training examples teach a 27B open model to reply to a learner's reasoning in a way that strong general-purpose models seldom do unprompted. The fine-tuned model names the likely error, hedges it, keeps the reply short and asks the next question, and a generic control trained on the same prompts does not learn this. On the content lines of each case it trails Opus 5.5 and stays ahead of GPT-6 Sol. At 27B, the share of replies both graders marked free of false statements rose from 75% to 91% and serious errors fell from 26 to 0, though the smaller models paid for the new style with more false statements. Our next experiments will prompt general-purpose models with B1 to B5 and test on fresh cases from a separate source. We will also adopt the stricter correctness rule of Table 12 for the next version of the benchmark, measure general capability after fine-tuning and add a grader from a third provider.

References

1. Arora, R. K., Wei, J., Hicks, R. S., et al. (2025). HealthBench: Evaluating Large Language Models Towards Improved Human Health. *arXiv preprint*. <https://arxiv.org/abs/2505.08775>
2. Biderman, D., Portes, J., Gonzalez Ortiz, J. J., et al. (2024). LoRA Learns Less and Forgets Less. *Transactions on Machine Learning Research*. <https://arxiv.org/abs/2405.09673>
3. Black, P., and Wiliam, D. (1998). Assessment and Classroom Learning. *Assessment in Education: Principles, Policy & Practice*, 5(1), 7–74. <https://doi.org/10.1080/0969595980050102>
4. Brown, J. S., and Burton, R. R. (1978). Diagnostic Models for Procedural Bugs in Basic Mathematical Skills. *Cognitive Science*, 2(2), 155–192. https://doi.org/10.1207/s15516709cog0202_4
5. Card, D., Henderson, P., Khandelwal, U., Jia, R., Mahowald, K., and Jurafsky, D. (2020). With Little Power Comes Great Responsibility. *Proceedings of EMNLP 2020*, pp. 9263–9274. <https://doi.org/10.18653/v1/2020.emnlp-main.745>
6. Chevalier, A., Geng, J., Wettig, A., et al. (2024). Language Models as Science Tutors. *Proceedings of ICML 2024*, PMLR 235, pp. 8310–8335. <https://arxiv.org/abs/2402.11111>
7. Chi, M. T. H., de Leeuw, N., Chiu, M.-H., and LaVancher, C. (1994). Eliciting Self-Explanations Improves Understanding. *Cognitive Science*, 18(3), 439–477. https://doi.org/10.1207/s15516709cog1803_3
8. Chiang, C.-H., and Lee, H.-y. (2023). Can Large Language Models Be an Alternative to Human Evaluations? *Proceedings of ACL 2023*, pp. 15607–15631. <https://doi.org/10.18653/v1/2023.acl-long.870>
9. Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1), 37–46. <https://doi.org/10.1177/001316446002000104>
10. Daheim, N., Macina, J., Kapur, M., Gurevych, I., and Sachan, M. (2024). Stepwise Verification and Remediation of Student Reasoning Errors with Large Language Model Tutors. *Proceedings of EMNLP 2024*, pp. 8386–8411. <https://doi.org/10.18653/v1/2024.emnlp-main.478>
11. Dror, R., Baumer, G., Shlomov, S., and Reichart, R. (2018). The Hitchhiker's Guide to Testing Statistical Significance in Natural Language Processing. *Proceedings of ACL 2018*, pp. 1383–1392. <https://doi.org/10.18653/v1/P18-1128>
12. Dubois, Y., Galambosi, B., Liang, P., and Hashimoto, T. B. (2024). Length-Controlled AlpacaEval: A Simple Way to Debias Automatic Evaluators. *Conference on Language Modeling (COLM 2024)*. <https://arxiv.org/abs/2404.04475>
13. Durkin, K., and Rittle-Johnson, B. (2012). The Effectiveness of Using Incorrect Examples to Support Learning about Decimal Magnitude. *Learning and Instruction*, 22(3), 206–214. <https://doi.org/10.1016/j.learninstruc.2011.11.001>
14. Efron, B., and Tibshirani, R. J. (1993). An Introduction to the Bootstrap. *Chapman & Hall/CRC*. <https://doi.org/10.1201/9780429246593>
15. Feuer, B., Goldblum, M., Datta, T., et al. (2025). Style Outweighs Substance: Failure Modes of LLM Judges in Alignment Benchmarking. *International Conference on Learning Representations (ICLR 2025)*. <https://arxiv.org/abs/2409.15268>
16. Gebru, T., Morgenstern, J., Vecchione, B., et al. (2021). Datasheets for Datasets. *Communications of the ACM*, 64(12), 86–92. <https://doi.org/10.1145/3458723>
17. Graesser, A. C., Person, N. K., and Magliano, J. P. (1995). Collaborative Dialogue Patterns in Naturalistic One-to-One Tutoring. *Applied Cognitive Psychology*, 9(6), 495–522. <https://doi.org/10.1002/acp.2350090604>
18. Gudibande, A., Wallace, E., Snell, C., et al. (2024). The False Promise of Imitating Proprietary Language Models. *International Conference on Learning Representations (ICLR 2024)*. <https://arxiv.org/abs/2305.15717>
19. Hattie, J., and Timperley, H. (2007). The Power of Feedback. *Review of Educational Research*, 77(1), 81–112. <https://doi.org/10.3102/003465430298487>
20. Holm, S. (1979). A Simple Sequentially Rejective Multiple Test Procedure. *Scandinavian Journal of Statistics*, 6(2), 65–70. <https://www.jstor.org/stable/4615733>
21. Hu, E. J., Shen, Y., Wallis, P., et al. (2022). LoRA: Low-Rank Adaptation of Large Language Models. *International Conference on Learning Representations (ICLR 2022)*. <https://arxiv.org/abs/2106.09685>
22. Jacobs, V. R., Lamb, L. L. C., and Philipp, R. A. (2010). Professional Noticing of Children's Mathematical Thinking. *Journal for Research in Mathematics Education*, 41(2), 169–202. <https://doi.org/10.5951/jresmetheduc.41.2.0169>
23. Jurenka, I., Kunesch, M., McKee, K. R., et al. (2024). Towards Responsible Development of Generative AI for Education: An Evaluation-Driven Approach. *arXiv preprint*. <https://arxiv.org/abs/2407.12687>
24. Kluger, A. N., and DeNisi, A. (1996). The Effects of Feedback Interventions on Performance: A Historical Review, a Meta-Analysis, and a Preliminary Feedback Intervention Theory. *Psychological Bulletin*, 119(2), 254–284. <https://doi.org/10.1037/0033-2909.119.2.254>
25. Kochmar, E., Maurya, K. K., Petukhova, K., Srivatsa, K. A., Tack, A., and Vasselli, J. (2025). Findings of the BEA 2025 Shared Task on Pedagogical Ability Assessment of AI-powered Tutors. *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 1011–1033. <https://doi.org/10.18653/v1/2025.bea-1.77>
26. Leatham, K. R., Peterson, B. E., Stockero, S. L., and Van Zoest, L. R. (2015). Conceptualizing Mathematically Significant Pedagogical Opportunities to Build on Student Thinking. *Journal for Research in Mathematics Education*, 46(1), 88–124. <https://doi.org/10.5951/jresmetheduc.46.1.0088>

27. Lee, Y., Kim, J., Kim, J., et al. (2025). CheckEval: A Reliable LLM-as-a-Judge Framework for Evaluating Text Generation Using Checklists. *Proceedings of EMNLP 2025*, pp. 15771–15798. <https://doi.org/10.18653/v1/2025.emnlp-main.796>
28. Liu, J., Huang, Z., Xiao, T., et al. (2024). SocraticLM: Exploring Socratic Personalized Teaching with Large Language Models. *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*. https://proceedings.neurips.cc/paper_files/paper/2024/hash/9bae399d1f34b8650351c1bd3692aeae-Abstract-Conference.html
29. Long, L., Wang, R., Xiao, R., et al. (2024). On LLMs-Driven Synthetic Data Generation, Curation, and Evaluation: A Survey. *Findings of ACL 2024*, pp. 11065–11082. <https://doi.org/10.18653/v1/2024.findings-acl.658>
30. Macina, J., Daheim, N., Chowdhury, S. P., et al. (2023). MathDial: A Dialogue Tutoring Dataset with Rich Pedagogical Properties Grounded in Math Reasoning Problems. *Findings of EMNLP 2023*, pp. 5602–5621. <https://doi.org/10.18653/v1/2023.findings-emnlp.372>
31. Macina, J., Daheim, N., Hakimi, I., Kapur, M., Gurevych, I., and Sachan, M. (2025). MathTutorBench: A Benchmark for Measuring Open-ended Pedagogical Capabilities of LLM Tutors. *Proceedings of EMNLP 2025*, pp. 204–221. <https://doi.org/10.18653/v1/2025.emnlp-main.11>
32. Maurya, K. K., Srivatsa, K. A., Petukhova, K., and Kochmar, E. (2025). Unifying AI Tutor Evaluation: An Evaluation Taxonomy for Pedagogical Ability Assessment of LLM-Powered AI Tutors. *Proceedings of NAACL 2025*, pp. 1234–1251. <https://doi.org/10.18653/v1/2025.naacl-long.57>
33. National Governors Association Center for Best Practices and Council of Chief State School Officers (2010). Common Core State Standards for Mathematics. Washington, DC. <https://www.corestandards.org/Math/>
34. Ni, Y., and Zhou, Y.-D. (2005). Teaching and Learning Fraction and Rational Numbers: The Origins and Implications of Whole Number Bias. *Educational Psychologist*, 40(1), 27–52. https://doi.org/10.1207/s15326985ep4001_3
35. Otero, N., Druga, S., and Lan, A. (2025). A Benchmark for Math Misconceptions: Bridging Gaps in Middle School Algebra with AI-Supported Instruction. *Discover Education*, 4, Article 277. <https://doi.org/10.1007/s44217-025-00742-w>
36. Panickssery, A., Bowman, S. R., and Feng, S. (2024). LLM Evaluators Recognize and Favor Their Own Generations. *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*. <https://arxiv.org/abs/2404.13076>
37. Petukhova, K., Nguyen, T. D., and Kochmar, E. (2026). Towards Pedagogically Aligned LLM Tutors for Math Mistake Remediation. *arXiv preprint*. <https://arxiv.org/abs/2606.21502>
38. Resnick, L. B., Nesher, P., Leonard, F., Magone, M., Omanson, S., and Peled, I. (1989). Conceptual Bases of Arithmetic Errors: The Case of Decimal Fractions. *Journal for Research in Mathematics Education*, 20(1), 8–27. <https://doi.org/10.5951/jresmetheduc.20.1.0008>
39. Rittle-Johnson, B., Siegler, R. S., and Alibali, M. W. (2001). Developing Conceptual Understanding and Procedural Skill in Mathematics: An Iterative Process. *Journal of Educational Psychology*, 93(2), 346–362. <https://doi.org/10.1037/0022-0663.93.2.346>
40. Sainz, O., Campos, J. A., García-Ferrero, I., Etxaniz, J., Lopez de Lacalle, O., and Agirre, E. (2023). NLP Evaluation in Trouble: On the Need to Measure LLM Data Contamination for Each Benchmark. *Findings of EMNLP 2023*, pp. 10776–10787. <https://doi.org/10.18653/v1/2023.findings-emnlp.722>
41. Shute, V. J. (2008). Focus on Formative Feedback. *Review of Educational Research*, 78(1), 153–189. <https://doi.org/10.3102/0034654307313795>
42. Siegler, R., Carpenter, T., Fennell, F., Geary, D., Lewis, J., Okamoto, Y., Thompson, L., and Wray, J. (2010). Developing Effective Fractions Instruction for Kindergarten Through 8th Grade (NCEE 2010-4039). *National Center for Education Evaluation and Regional Assistance, Institute of Education Sciences, U.S. Department of Education*. https://ies.ed.gov/ncee/wwc/docs/practiceguide/fraction-s_pg_093010.pdf
43. Smith, J. P., III, diSessa, A. A., and Roschelle, J. (1993). Misconceptions Reconceived: A Constructivist Analysis of Knowledge in Transition. *Journal of the Learning Sciences*, 3(2), 115–163. https://doi.org/10.1207/s15327809jls0302_1
44. Sonkar, S., Chen, X., Liu, N., Baraniuk, R. G., and Sachan, M. (2024). LLM-based Cognitive Models of Students with Misconceptions. *arXiv preprint*. <https://arxiv.org/abs/2410.12294>
45. Srinivasa, R. S., Che, Z., Zhang, C. B. C., et al. (2025). TutorBench: A Benchmark to Assess Tutoring Capabilities of Large Language Models. *arXiv preprint*. <https://arxiv.org/abs/2510.02663>
46. Steinle, V., and Stacey, K. (2004). A Longitudinal Study of Students' Understanding of Decimal Notation: An Overview and Refined Results. *Proceedings of the 27th Annual Conference of the Mathematics Education Research Group of Australasia*, Vol. 2, pp. 541–548.
47. Wang, P., Li, L., Chen, L., et al. (2024b). Large Language Models are not Fair Evaluators. *Proceedings of ACL 2024*, pp. 9440–9450. <https://doi.org/10.18653/v1/2024.acl-long.511>
48. Wang, R. E., Zhang, Q., Robinson, C., Loeb, S., and Demsky, D. (2024a). Bridging the Novice-Expert Gap via Models of Decision-Making: A Case Study on Remediating Math Mistakes. *Proceedings of NAACL 2024*, pp. 2174–2199. <https://doi.org/10.18653/v1/2024.naacl-long.120>
49. Wang, Y., Kordi, Y., Mishra, S., et al. (2023). Self-Instruct: Aligning Language Models with Self-Generated Instructions. *Proceedings of ACL 2023*, pp. 13484–13508. <https://doi.org/10.18653/v1/2023.acl-long.754>
50. Ye, S., Kim, D., Kim, S., et al. (2024). FLASK: Fine-grained Language Model Evaluation based on Alignment Skill Sets. *International Conference on Learning Representations (ICLR 2024)*. <https://arxiv.org/abs/2307.10928>
51. Zheng, L., Chiang, W.-L., Sheng, Y., et al. (2023). Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. *Advances in Neural Information Processing Systems 36 (NeurIPS 2023), Datasets and Benchmarks Track*. <https://arxiv.org/abs/2306.05685>

52. Zhou, C., Liu, P., Xu, P., et al. (2023). LIMA: Less Is More for Alignment. *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*. <https://arxiv.org/abs/2305.11206>

Appendix A. Grader Instruction

Each grader receives the instruction below. The four slots in braces are filled in for every reply: {view} with the system and user messages the model was given, {criteria} and {critical} with the case's numbered checklist lines and serious errors, and {reply} with the model's reply. In the instruction, "criteria" are checklist lines and "critical" lines are serious errors.

You grade one tutor reply to a student, line by line, against a checklist. Work from the checklist lines exactly as written; do not add your own.

Tutor's view (what the tutor was shown):
{view}

Checklist lines the reply should meet ("criteria", numbered from 0):
{criteria}

Serious-error lines ("critical", numbered from 0): mark yes only if the reply commits the error.
{critical}

The tutor's reply:
<reply>
{reply}
</reply>

Return only JSON: {"criteria": [true|false, ...one per criteria line...], "critical": [true|false, ...one per critical line...], "reason": "one short sentence"}

The same instruction filled in for Case 1 of Appendix B and the reply of Qwen3.8-27B+SENSUS (second run), exactly as a grader received it:

You grade one tutor reply to a student, line by line, against a checklist. Work from the checklist lines exactly as written; do not add your own.

Tutor's view (what the tutor was shown):

You are a mathematics tutor. A student is working on the question below. Reply to the student directly, as you would in the conversation.

Question (grades 4 and 5; 4.NF.A.1):

Priya wrote $3/6 = 4/10$, but this is not true. Change exactly one of the four numbers so that the equation is true.

What the student did and said:

The learner is working on making $3/6 = 4/10$ true by changing one number. The learner writes: " $4/10$ is bigger than $3/6$, because 4 is bigger than 3 and 10 is bigger than 6. So I need to make $4/10$ smaller. I changed the 4 to a 3: $3/6 = 3/10$."

Write your reply to the student.

Checklist lines the reply should meet ("criteria", numbered from 0):

0. (critical) Nothing false (for example, saying $4/10$ is greater than $3/6$, or saying $3/6 = 3/10$ is true).
1. Names a likely error tied to the learner's "4 is bigger than 3 and 10 is bigger than 6": judging which fraction is bigger by comparing the top numbers and the bottom numbers as whole numbers.
2. Says the error as likely, not certain, using words such as "looks like", "might", "maybe" or "it seems". A reply that names no reason for the error is marked no.
3. States that $4/10$ is less than $3/6$.
4. Gives a reason a grade 4 learner can check for why $4/10$ is less than $3/6$. Comparing each to one half, a drawing or renaming both with the same denominator counts. Only stating which is bigger does not count.
5. Names a specific thing the learner did correctly, such as changing only one number or comparing before changing. "Good job" alone is marked no.
6. Ends with a question or task the learner can answer next. A trailing "Does that help?" alone does not count.
7. Is about 120 words or fewer.

Serious-error lines ("critical", numbered from 0): mark yes only if the reply commits the error.

0. States as certain the reason the learner thought $4/10$ is bigger, for example "You thought bigger numbers make a bigger fraction" with no word such as "maybe" or "looks like". Repeating what the learner wrote does not count.
1. Says or agrees that $4/10$ is greater than $3/6$.
2. Says or agrees that $3/6 = 3/10$ is true.

The tutor's reply:
<reply>

You compared the top numbers and the bottom numbers separately. It sounds like you may be treating bigger numbers on top and bottom as meaning a bigger fraction. Instead, let's use equal-sized rectangles divided into sixths and tenths. Three sixth-sized pieces cover half a rectangle; four tenth-sized pieces also cover half. So $3/6$ and $4/10$ are already

equal. Your change to $3/10$ makes the right side smaller than the left. If you want to change just one number, which number could you change to make the two sides equal again?

</reply>

Return only JSON: {"criteria": [true|false, ...one per criteria line...], "critical": [true|false, ...one per critical line...], "reason": "one short sentence"}

Appendix B. Two Cases in Full

Case 1 is the first case in file order. Case 2 is the first case in file order that the fine-tuned 27B model passes fully while GPT-6 Sol scores below 0.6. The replies are from Qwen3.8-27B+SENSUS (second run) and GPT-6 Sol as recorded, and each mark is shown as grader 1 / grader 2. All 76 cases, with every model's reply, can be browsed on the evaluation cases page.

Case 1: Equivalent Fractions (4.NE.A.1)

Question. Priya wrote $3/6 = 4/10$, but this is not true. Change exactly one of the four numbers so that the equation is true.

What the learner did and said. The learner is working on making $3/6 = 4/10$ true by changing one number. The learner writes: "4/10 is bigger than 3/6, because 4 is bigger than 3 and 10 is bigger than 6. So I need to make 4/10 smaller. I changed the 4 to a 3: $3/6 = 3/10$."

Line	Kind	Text	Qwen3.8-27B+SENSUS (G1 / G2)	GPT-6 Sol (G1 / G2)
C01	Correctness	Nothing false (for example, saying $4/10$ is greater than $3/6$, or saying $3/6 = 3/10$ is true).	× / ×	✓ / ✓
C02	Case content	Names a likely error tied to the learner's "4 is bigger than 3 and 10 is bigger than 6": judging which fraction is bigger by comparing the top numbers and the bottom numbers as whole numbers.	✓ / ✓	✓ / ✓
C03	Hedging	Says the error as likely, not certain, using words such as "looks like", "might", "maybe" or "it seems". A reply that names no reason for the error is marked no.	✓ / ✓	× / ×
C04	Case content	States that $4/10$ is less than $3/6$.	× / ×	✓ / ✓
C05	Case content	Gives a reason a grade 4 learner can check for why $4/10$ is less than $3/6$. Comparing each to one half, a drawing or renaming both with the same denominator counts. Only stating which is bigger does not count.	× / ×	✓ / ✓
C06	Case content	Names a specific thing the learner did correctly, such as changing only one number or comparing before changing. "Good job" alone is marked no.	× / ×	✓ / ✓
C07	Next step	Ends with a question or task the learner can answer next. A trailing "Does that help?" alone does not count.	✓ / ✓	× / ×
C08	Length	Is about 120 words or fewer.	✓ / ✓	✓ / ✓
S01	Serious error	States as certain the reason the learner thought $4/10$ is bigger, for example "You thought bigger numbers make a bigger fraction" with no word such as "maybe" or "looks like". Repeating what the learner wrote does not count.	- / -	- / -
S02	Serious error	Says or agrees that $4/10$ is greater than $3/6$.	- / -	- / -
S03	Serious error	Says or agrees that $3/6 = 3/10$ is true.	- / -	- / -

Qwen3.8-27B+SENSUS · score 0.50

You compared the top numbers and the bottom numbers separately. It sounds like you may be treating bigger numbers on top and bottom as meaning a bigger fraction. Instead, let's use equal-sized rectangles divided into sixths and tenths. Three sixth-sized pieces cover half a rectangle; four tenth-sized pieces also cover half. So $\frac{3}{6}$ and $\frac{4}{10}$ are already equal. Your change to $\frac{3}{10}$ makes the right side smaller than the left. If you want to change just one number, which number could you change to make the two sides equal again?

GPT-6 Sol · score 0.75

I like that you checked whether the fractions were the same size. But a bigger numerator **and** a bigger denominator don't necessarily make a bigger fraction. If you shade 3 of 6 equal parts, you shade **half** the shape. If you shade 4 of 10 equal parts, you shade **less than half**.

So changing the 4 to a 3 makes the right side even smaller. Instead, change **4 to 5**:

$$\frac{3}{6} = \frac{5}{10}$$

Both fractions show half of a shape shaded.

Case 2: Comparing Fractions (4.NF.A.2)

Question. Write the whole number that goes in the blank to make the statement true: $3/5 < 6/__ < 3/4$

What the learner did and said. The learner is filling the blank in $3/5 < 6/__ < 3/4$. The learner writes: "3/5 is 6/10 and 3/4 is 6/8. The middle has to be bigger than 6/10, so I put 11, because 11 is bigger than 10. My answer is 6/11."

Line	Kind	Text	Qwen3.8-27B+SENSUS (G1 / G2)	GPT-6 Sol (G1 / G2)
C01	Correctness	Nothing false (for example, saying 6/11 is greater than 6/10, or saying 3/5 is not equal to 6/10).	✓ / ✓	✓ / ✓
C02	Case content	Names a likely error tied to the learner's choice of 11: treating a bigger denominator as making a bigger fraction.	✓ / ✓	× / ✓
C03	Hedging	Says the error as likely, not certain, using words such as "looks like", "might", "maybe" or "it seems". A reply that names no reason for the error is marked no.	✓ / ✓	× / ×
C04	Case content	Explains in words or with a picture that a larger denominator means smaller pieces when the numerators are the same.	✓ / ✓	× / ×
C05	Case content	Names the learner's correct renaming of 3/5 as 6/10 or of 3/4 as 6/8.	✓ / ✓	✓ / ✓
C06	Next step	Ends with a question or task the learner can answer next. A trailing "Does that help?" alone does not count.	✓ / ✓	× / ×
C07	Length	Is about 120 words or fewer.	✓ / ✓	✓ / ✓
S01	Serious error	States as certain the reason the learner chose 11, for example "You thought a bigger bottom means a bigger fraction" with no word such as "maybe" or "looks like". Repeating what the learner wrote does not count.	- / -	- / -
S02	Serious error	Says or agrees that 11 makes the statement true.	- / -	- / -
S03	Serious error	Says that renaming 3/5 as 6/10 or renaming 3/4 as 6/8 was a mistake.	- / -	- / -

Qwen3.8-27B+SENSUS · score 1.00

Your conversions to $\frac{6}{10}$ and $\frac{6}{8}$ keep the fractions' values the same. It sounds like you may be using the rule for numerators when checking the middle fraction's denominator. With the same numerator, a larger denominator means smaller pieces, so $\frac{6}{11}$ is smaller than $\frac{6}{10}$. To be between the endpoints, the middle fraction needs a denominator between 8 and 10. What whole number can you put in the blank?

GPT-6 Sol · score 0.43

You made a great start by rewriting $\frac{3}{5}$ as $\frac{6}{10}$ and $\frac{3}{4}$ as $\frac{6}{8}$!

Here's the tricky part: when the numerators are both 6, a **larger denominator makes a smaller fraction**. So $\frac{6}{11}$ is

less than $\frac{6}{10}$, not greater.

The whole number between 8 and 10 is **9**, and

$$\frac{6}{10} < \frac{6}{9} < \frac{6}{8}.$$

So the number in the blank is **9**.

Appendix C. Datasheet

SENSUS SFT v1.1 and the SENSUS Bench v1.0 evaluation kit, summarised in the style of a datasheet.

Question	Answer
What do the instances represent?	One teaching moment each: a grade 4 or 5 mathematics question, what a synthetic learner did and said, and a tutor reply (training) or a checklist, serious errors and reference reply (evaluation).
How many instances are there?	1,015 training conversations over 884 distinct questions; 76 evaluation cases with 549 checklist lines and 218 serious errors.
What fields does each record have?	Training: chat messages (system, user, assistant). Evaluation: an identifier, the Common Core standard, the messages shown to the model, the checklist, the serious errors and the reference reply.
Is any personal information included?	No. Every learner and situation is synthetic.
How was the data collected?	The data is synthetic. Its production process is not described in this report.
How is the data split?	By question group: each group is wholly in training or wholly in evaluation, and no evaluation question appears in the training file.
What are the intended uses?	Supervised fine-tuning of models to give short, diagnostic replies to a learner's work; reproducing and extending these results; research on tutoring behaviour and misconception diagnosis.
What uses are out of scope?	Use as a sole source of instruction for children without independent review; claims about learning gains; treating the reference replies as the only acceptable teaching approach.
How is it distributed, and under what licence?	The data under CC BY 4.0 and the scripts under the MIT licence, with SHA-256 checksums, a manifest and an offline verifier.
What version is it?	SENSUS SFT v1.1, identical to the v1.0 file used for the recorded runs except for a topic label on the first line of some user messages; evaluation kit v1.0.