GRADE: Generalizable Reasoning-Aware Dialogue Evaluation for AI Tutors

Anonymous ACL submission

Abstract

Evaluating AI tutor responses requires more than factual correctness: tutors must identify mistakes, locate errors, provide guidance, and offer actionable next steps. We present GRADE, a systematic study of open-source models for pedagogical ability assessment in student-tutor dialogues. Building on the BEA 2025 TutorMind setting, we evaluate 120 configurations across five language models, zero-shot inference, LoRA fine-tuning, synthetic augmentation, CoT+Reasoning, and single-task versus multitask formulations. Gemma3-12B performs best for single-task evaluation, while Gemma3-27B in 8-bit precision is more reliable for multitask prediction. We find that augmentation helps models that struggle with the original data, verification adds limited gains despite higher cost, and CoT+Reasoning is more useful for synthetic data generation than direct classification. We further show that LoRA fine-tuning on structured classification objectives interferes with instruction-following behavior under thinking mode, redirecting generation away from the required evaluation format. Carbon analysis shows that model choice and reasoning mode substantially affect emissions. Overall, GRADE shows that carefully selected open-source LoRA pipelines can match or surpass proprietary and ensemble-based systems on key pedagogical dimensions, with code and data available at `https://anonymous.4open.science/r/GRADE-D0B3`.

1 Introduction

Large language models are increasingly deployed as AI tutors, yet evaluating whether they teach well requires more than factual correctness. Effective tutors must identify student mistakes, locate errors, provide meaningful guidance, and offer actionable next steps (Tack and Piech, 2022; Maurya et al., 2025), but varied evaluation criteria across prior work make cross-system comparison difficult, motivating standardized automatic assessment (Tack and Piech, 2022; Tack et al., 2023).

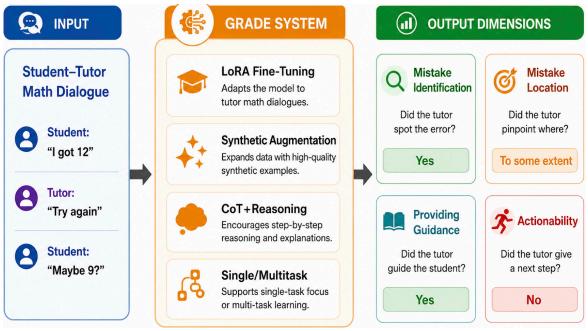

Figure 1: Overview of the GRADE pipeline: a student-tutor math dialogue is processed through LoRA fine-tuning, synthetic augmentation, CoT+Reasoning, and single/multitask formulations, producing evaluations across four pedagogical dimensions.

The BEA 2025 Shared Task on Pedagogical Ability Assessment of AI-powered Tutors (Kochmar et al., 2025) addresses this through a unified framework grounded in learning science (Maurya et al., 2025), using math dialogues from MathDial (Macina et al., 2023) and Bridge (Wang et al., 2024). The task evaluates four dimensions: mistake identification, mistake location, providing guidance, and actionability. While parameter-efficient fine-tuning (Hu et al., 2022) and synthetic augmentation improved performance, most work focused on individual dimensions and standard non-reasoning models, leaving reasoning, multitask learning, and efficiency tradeoffs underexplored (Dekmak et al., 2025; Kochmar et al., 2025).

In particular, the role of chain-of-thought reasoning (Wei et al., 2022) remains underexplored across two stages: as a direct training and inference strategy, and as a tool for generating higher-quality synthetic data. Models such as Qwen3 (Yang et al., 2025) support both thinking and non-thinking modes, enabling isolation of reasoning's contribution at each stage. Whether jointly modeling all four dimensions in a multitask formulation

outperforms separate task-specific models also remains open. Finally, despite growing interest in sustainable NLP, the environmental cost of these choices has not been studied in this setting (Strubell et al., 2019).

We build on the strongest open-source BEA 2025 baseline and present GRADE (Figure 1), conducting 120 experimental runs across model scale, training method, augmentation strategy, and task formulation. Understanding which modeling choices lead to reliable pedagogical evaluation is essential for building trustworthy AI tutoring systems and guiding future work in educational NLP. Our work is guided by three research questions:

RQ1 How do model scale and fine-tuning strategy affect pedagogical ability assessment in educational dialogue evaluation?

RQ2 Does chain-of-thought reasoning help more as a tool for synthetic data generation, as a direct training and inference strategy, or both?

RQ3 Does jointly training one model in a multi-task formulation across all four pedagogical dimensions yield more balanced and competitive performance than independently trained single-task models?

Our contributions are threefold: (1) a systematic study across all four BEA 2025 dimensions covering model scale, training method, augmentation, and task formulation; (2) chain-of-thought reasoning analysis at both inference and data generation time, including self-verification filtering; and (3) the first carbon emissions analysis of pedagogical evaluation systems using CodeCarbon (Courty et al., 2024), with practical recommendations for compute-efficient model selection.

2 Related Work

2.1 AI Tutor Evaluation in Educational Dialogues

Evaluating AI tutor responses has become an important problem, but early work used varied criteria that made systems difficult to compare. Tack and Piech (2022) proposed an early framework for measuring whether generative models could speak like teachers, understand students, and provide help, while the BEA 2023 Shared Task (Tack et al., 2023) further emphasized response generation in educational dialogues. More recent work argues that evaluation should be grounded in learning science and move beyond surface level metrics (Jurenka et al., 2024).

Maurya et al. (2025) addressed this gap by introducing a unified taxonomy for pedagogical ability assessment, which later formed the basis of the BEA 2025 Shared Task (Kochmar et al., 2025). The task operationalizes four dimensions, mistake identification, mistake location, providing guidance, and actionability, using educational math dialogues from MathDial (Macina et al., 2023) and Bridge (Wang et al., 2024). Our work builds on this shared task by extending evaluation across open-source models, reasoning-capable systems, and multitask training across all four dimensions.

2.2 Fine-Tuning and Parameter-Efficient Adaptation of LLMs

Fine-tuning is widely used for downstream NLP tasks, but updating all parameters is computationally expensive. LoRA (Hu et al., 2022) reduces this cost by freezing pretrained weights and training small rank-decomposition modules, and became a common strategy in BEA 2025, where carefully tuned open-source models competed with larger proprietary systems (Kochmar et al., 2025).

Prior systems explored varied approaches: TutorMind (Dekmak et al., 2025) combined LoRA with synthetic augmentation for minority classes; BJTU (Fan et al., 2025) used task-aware prompt tuning; MSA (Hikal et al., 2025) added disagreement-aware ensemble inference; bea-jh (Roh and Bang, 2025) applied Group Relative Policy Optimization with thinking-based rationales; BLCU-ICALL (An et al., 2025) compared supervised fine-tuning, in-context learning, and reinforcement learning; and K-NLPers (Park et al., 2025) relied on GPT-4.1 prompting without fine-tuning. Together these confirm parameter-efficient fine-tuning as a strong baseline, though reasoning and task formulation remain underexplored across all four dimensions.

2.3 Data Augmentation for Imbalanced Classification

Class imbalance remains a major challenge in NLP classification, as majority labels can cause models to underperform on informative minority categories. Prior augmentation methods include rule based transformations, back translation, and model based generation (Feng et al., 2021). Recent work increasingly uses LLMs as data generators, and educational NLP studies show that open-source LLMs can provide useful feedback and training sig-

nal when guided appropriately (Koutcheme et al., 2024).

In the BEA 2025 Shared Task, imbalance was especially severe, with the majority class covering nearly 78% of annotations in mistake identification (Kochmar et al., 2025). Systems addressed this through synthetic data generation, oversampling, and class weighted losses. TutorMind (Dekmak et al., 2025) generated minority class examples but noted label noise, while TBA (Gombert et al., 2025) explored DARE-TIES model merging and NLIP (Saha et al., 2025) combined oversampling with multi-task learning. Our work builds on these efforts by using a reasoning-capable model for augmentation and adding a self-verification step to filter synthetic examples before training, targeting the most underrepresented minority labels across all four pedagogical dimensions.

2.4 Reasoning and Multi-Task Learning in NLP

Chain-of-thought prompting (Wei et al., 2022) showed that intermediate reasoning can improve performance on complex tasks, motivating models with built-in reasoning capabilities. Qwen3 (Yang et al., 2025) supports both thinking and non-thinking modes, enabling direct comparison between deliberate reasoning and standard inference. Alongside this, open-source models such as LLaMA 3 (Dubey et al., 2024), Mistral (Jiang et al., 2023), and Gemma 3 (Gemma Team, 2025) have made rigorous experimentation increasingly feasible without relying only on proprietary systems.

Multi-task learning offers a complementary direction by training one model across related objectives to exploit shared structure. In BEA 2025, most systems treated each pedagogical dimension separately, while TBA (Gombert et al., 2025) showed that cross-dimension information can be useful through model merging. Our work extends this direction by directly comparing single-task and multi-task LoRA fine-tuning, while also testing whether chain-of-thought reasoning complements multi-task pedagogical evaluation.

3 Dataset

Following (Dekmak et al., 2025), we build on the BEA 2025 TutorMind data for pedagogical evaluation of tutor responses. While prior work evaluated only a single dimension, a comprehensive assessment requires all four pedagogical dimensions jointly, motivating our extended setup. Each example is a tutor–student interaction with a task-specific label: *Yes*, *No*, or *To some extent*. We evaluate four dimensions: *Mistake Identification* (MI), *Mistake Location* (ML), *Providing Guidance* (PG), and *Actionability* (ACT).

Dataset	Train	Val	Total
TutorMind original MI	1,980	496	2,476
TutorMind augmented MI pool	2,180	496	2,676
Deduplicated augmented MI pool	–	–	2,668
Our MI	2,130	538	2,668
Our ML	1,974	494	2,468
Our PG	1,976	494	2,470
Our Act	1,974	494	2,468
Our MT	1,976	494	2,470
Our MI + Gen	3,130	538	3,668
Our ML + Gen	2,974	494	3,468
Our PG + Gen	2,976	494	3,470
Our Act + Gen	2,974	494	3,468
Our MT + Gen	2,976	494	3,470
Our MI + Gen+Verify	3,130	538	3,668
Our ML + Gen+Verify	2,974	494	3,468
Our PG + Gen+Verify	2,976	494	3,470
Our Act + Gen+Verify	2,974	494	3,468
Our MT + Gen+Verify	2,607	494	3,101

Table 1: Summary of Proposed Dataset Splits

TutorMind focused only on MI, using 2,476 instances (1,980 train / 496 val) augmented with 200 minority-class examples. We extend this to all four dimensions and a multitask (MT) formulation predicting all labels jointly. For MI, splits are derived from the augmented release after removing 8 exact duplicates. For ML, PG, ACT, and MT, we retain task-valid examples, remove N/A targets and exact duplicates, producing the splits in Table 1.

To address class imbalance, we construct two augmented variants with Qwen3-14B: *Qwen3 Gen*, where synthetic minority-class examples are used directly, and *Qwen3 Gen+Verify*, where generated examples are first verified by the same model. Each Gen dataset adds 1,000 synthetic examples (500/500 split per task), while MT + Gen+Verify contains 631 examples for reasons detailed in Section 4.4.

4 Methodology

4.1 Task Formulation

Each pedagogical dimension is treated as an independent three-class classification problem over student-tutor math dialogues. Given the full conversational context and a tutor response, a model assigns one of three labels: *Yes*, *No*, or *To some extent*. This three-way distinction is more demanding

than binary classification, as the intermediate class captures responses that partially satisfy a pedagogical criterion without fully meeting it, and accounts for the bulk of minority-class examples.

For single-task experiments, each model is trained and evaluated on one dimension at a time, producing a single `Evaluation:` line. For the multitask setting, all four dimensions are evaluated jointly within a single forward pass, requiring four structured output lines simultaneously. This joint formulation tests whether a single model can internalize cross-dimension relationships without task-specific specialization. Prompts for both settings are in Tables 7 and 8.

4.2 Models and Experimental Design

We evaluate five open-source instruction-tuned models: LLaMA-3.1-8B (Dubey et al., 2024), Mistral-7B (Jiang et al., 2023), Qwen3-14B (Yang et al., 2025), Gemma3-12B (Gemma Team, 2025), and Gemma3-27B (Gemma Team, 2025). LLaMA-3.1-8B and Mistral-7B replicate the TutorMind fine-tuning setup (Dekmak et al., 2025), while larger models test the effect of scale. Qwen3-14B additionally studies chain-of-thought reasoning via its native thinking mode.

Our design, summarized in Table 2, spans 120 runs across model scale, training method, augmentation strategy, reasoning mode, and task formulation. We compare zero-shot inference and LoRA fine-tuning (Hu et al., 2022), isolate chain-of-thought reasoning by toggling Qwen3-14B between Think OFF and Think ON, evaluate generated versus self-verified augmentation, and compare single-task with multitask models.

4.3 Fine-Tuning and Reasoning

All fine-tuning uses LoRA (Hu et al., 2022) with Unsloth (Han et al., 2023), applying adapters to all attention and feed-forward projection matrices with rank $r = 16$, scaling factor $\alpha = 16$, and no dropout. Models are trained in bfloat16, except Gemma3-27B (Gemma Team, 2025), which uses 8-bit quantization due to compute constraints. Training runs for 3 epochs with batch size 2, gradient accumulation over 8 steps, AdamW (Loshchilov and Hutter, 2019), learning rate 2×10^{-4}, 5 warmup steps, and weight decay 0.01. Inference uses greedy decoding with at most 64 new tokens.

Chain-of-thought reasoning is studied via Qwen3-14B (Yang et al., 2025), which supports native thinking mode. Think ON is enabled through `enable_thinking=True` with a reasoning-augmented prompt; Think OFF uses the standard prompt. This controlled toggle isolates reasoning's effect on classification. Full prompts are in Tables 7 and 8.

Experiment	Configuration	Runs
Baseline	2 models, Zero-shot, No Aug	10
	2 models, LoRA, No Aug	10
New Models (Ours)	3 models, Zero-shot, No Aug	15
	3 models, LoRA, No Aug	15
Reasoning (Ours)	Qwen3, Zero-shot, Think ON	5
	Qwen3, LoRA, Think ON	5
Augmentation (Ours)	5 models, LoRA, Generate	25
	5 models, LoRA, Gen+Verify	25
Combined (Ours)	Qwen3, LoRA, Think ON, Gen	5
	Qwen3, LoRA, Think ON, Gen+Verify	5
Total		**120**

Table 2: Summary of experimental configurations. Each run covers all five classification tasks.

4.4 Reasoning-Guided Data Augmentation

The BEA 2025 dataset is highly imbalanced, so we use Qwen3-14B with CoT+Reasoning to generate synthetic examples for minority labels (*No* and *To some extent*). For each task, the model receives a real student-tutor conversation and generates a one-sentence tutor response matching the target label. We compare *Gen*, which uses generated examples directly, with *Gen+Verify* (Gen+Ver. in tables), where Qwen3-14B retains only examples whose self-predicted label matches the intended minority class label. Full prompts are in Tables 9 and 10.

Each Gen dataset adds 1,000 synthetic examples split evenly between *No* and *To some extent*. Gen+Verify uses the same split, while MT + Gen+Verify contains only 631 examples: the verification step requires synthetic responses to satisfy all four dimensions simultaneously at the *To some extent* level, a constraint the model rarely fulfills, resulting in high rejection rates. This corroborates that self-verification introduces substantial overhead without consistent gains.

5 Evaluation & Results

5.1 Evaluation Metrics

We evaluate all systems using strict and lenient macro-averaged F1 and accuracy, following the official BEA 2025 protocol (Kochmar et al., 2025).

Strict evaluation treats *Yes*, *No*, and *To some extent* as three separate classes, while lenient evaluation merges *Yes* and *To some extent*. We use strict macro-averaged F1 as the primary metric as accuracy can be inflated under strong class imbalance (Dekmak et al., 2025). Statistical significance is assessed using 95% confidence intervals via bootstrap resampling. Lenient F1 results are reported in Appendix B.

5.2 Baseline Performance Without Augmentation

Zero Shot Analysis. Figure 2 reports strict F1 scores for the single-task zero-shot setting. MI is the strongest dimension across all models, with Qwen3-14B achieving the highest score of 0.622. ML is consistently the weakest, with Mistral-7B scoring as low as 0.288, confirming that localizing an error is substantially harder than detecting its presence. Gemma3-12B and Gemma3-27B (8-bit) are the most balanced overall, maintaining competitive scores across all four dimensions without any fine-tuning signal.

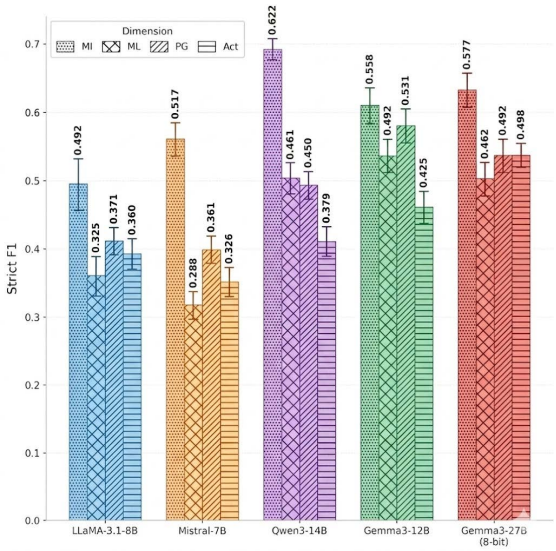

Figure 2: Single task zero shot strict F1 scores without augmentation across all five models and four pedagogical dimensions.

Figure 3 reports multitask zero-shot results. Joint prediction lowers performance for most models, with the largest drop for Mistral-7B, whose MI score falls from 0.517 to 0.329. Gemma3-27B (8-bit) is the most robust, retaining the highest MI score at 0.515 and remaining balanced across PG and Act, suggesting larger models handle joint prediction better under zero-shot inference.

LoRA Analysis. Figure 4 reports strict F1 after LoRA fine-tuning in the single-task setting. Gains are not uniform: Gemma3-12B achieves the strongest result at 0.750 on MI with competitive scores on PG and Act. Qwen3-14B improves substantially to 0.717 on MI, and LLaMA-3.1-8B shows clear gains across dimensions. Mistral-7B remains weak, and Gemma3-27B shows unstable behavior on MI and ML, suggesting 8-bit quantization limits its benefit from parameter-efficient fine-tuning.

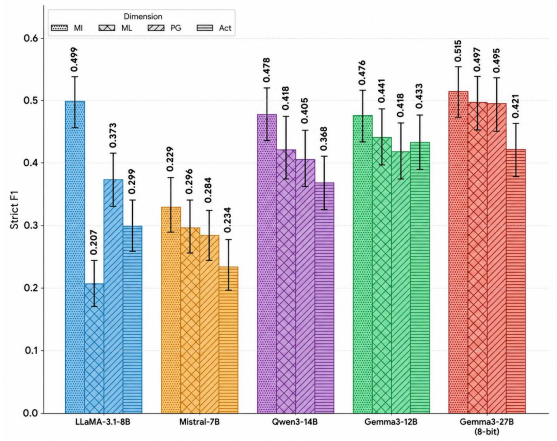

Figure 3: Multitask zero shot strict F1 scores without augmentation across all five models and four pedagogical dimensions.

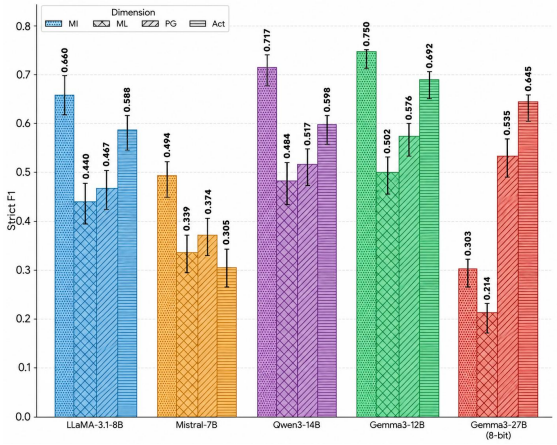

Figure 4: Single task LoRA strict F1 scores without augmentation across all five models and four pedagogical dimensions.

Figure 5 shows the multitask setting reverses this pattern. Gemma3-27B becomes the strongest multitask model, reaching 0.598 on MI and 0.644 on Act with balanced performance across PG and ML, while Gemma3-12B loses its single-task advantage. Overall, LoRA is an effective baseline, but its benefit depends on model scale, quantization, and task formulation: single-task training favors Gemma3-12B, while multitask training better suits Gemma3-27B, with larger models benefiting most from the joint formulation.

5.3 Effect of Data Augmentation & Verification

Table 3 shows that augmentation has highly model-dependent effects. The clearest gain appears for Gemma3-27B (8-bit) in the single-task setting, where MI rises from 0.30 to 0.77 with Qwen3 Gen+Verify, suggesting augmented data substantially recovers performance under the constrained 8-bit LoRA setup. Mistral-7B also benefits, especially in multitask evaluation, where MI improves from 0.31 to 0.48. In contrast, higher-performing baseline models such as Gemma3-12B, Qwen3-14B, and LLaMA-3.1-8B show limited gains or small regressions, suggesting their learning capacity on the original data is already saturated. Synthetic minority-class examples are thus most beneficial when a model has not yet reached its learning capacity, rather than as a universal strategy. Differences between Gen and Gen+Verify are small and inconsistent, indicating no reliable advantage from verification, consistent with the high rejection rates in MT Gen+Verify detailed in Section 4.4.

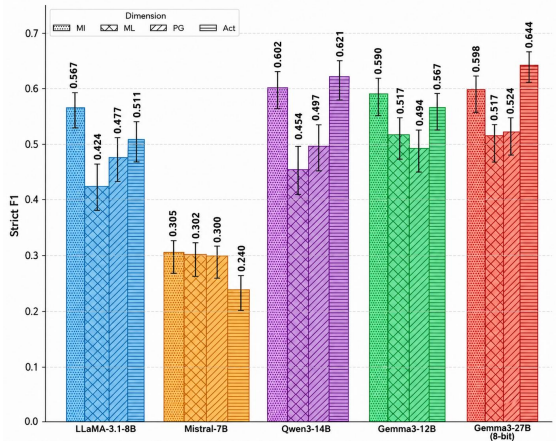

Figure 5: Multitask LoRA strict F1 scores without augmentation across all five models and four pedagogical dimensions.

5.4 Chain-of-Thought and Reasoning Analysis

Tables 4 and 5 show that CoT+Reasoning has a stage dependent effect. Without augmentation, Think ON weakens performance: the effect is negligible under zero shot inference, but severe under LoRA fine tuning, where MI falls from 0.72 to 0.22 with similar collapse across dimensions. This is not due to token budget, since each instance receives 1,024 thinking tokens. Instead, inspection shows an empty reasoning chain, while the model generates free form mathematical solutions rather than structured labels. This suggests that failure comes from the interaction between LoRA fine tuning and thinking mode, rather than thinking mode alone, since zero shot Think ON has negligible unknown rates of ∼1–2%. The result highlights a broader concern for reasoning capable models under parameter efficient fine tuning.

With augmented data, performance recovers sharply (Table 5): single-task MI reaches 0.72 and multitask MI reaches 0.62, with zero unparseable outputs under both strategies. Gen and Gen+Verify remain close with no consistent winner. Overall, CoT+Reasoning is more effective as a data enrichment mechanism than a direct inference strategy — standard inference should be preferred for classification, while reasoning mode is best reserved for data construction.

5.5 Comparison with Published Benchmarks

Table 6 compares our strongest configurations with prior BEA 2025 systems on the same development set and strict macro F1 protocol (An et al., 2025; Dekmak et al., 2025; Roh and Bang, 2025; Hikal et al., 2025). Gemini 2.5 Pro is strongest on ML (0.68) and PG (0.67), while our Gemma3 27B with GenVer reaches the best MI score (0.77). Gemma3 12B NoAug matches MSA on Act (0.69). Overall, open source LoRA can exceed closed source and ensemble based systems on MI while remaining broadly competitive across the other pedagogical dimensions.

5.6 Carbon Emission Analysis

As NLP systems scale, environmental cost becomes essential for sustainable research (Strubell et al., 2019). We track emissions using CodeCarbon v3.2.6 (Courty et al., 2024) on an NVIDIA L40S (GPU via NVML, CPU via TDP, RAM). Figures 6 and 7 show carbon cost is shaped by model choice, augmentation, and reasoning mode. Emissions vary more strongly by base model than augmentation alone: Mistral 7B and LLaMA 3.1 8B remain carbon efficient, while Gemma3 27B becomes the dominant contributor once augmentation is introduced. Qwen3 14B shows a sharper increase under augmentation; Gemma3 12B increases comparatively little, making it a strong middle cost option. Think ON substantially increases emissions, especially in the multitask LoRA setting, and is only justified for data construction given no consistent classification benefit. Pure data generation costs approximately 2.0 kg CO_2 for Gen and 4.2 kg CO_2

Setting	Model	No Aug				Qwen3 Gen				Qwen3 Gen+Verify			
		MI	ML	PG	Act	MI	ML	PG	Act	MI	ML	PG	Act
Single	LLaMA-3.1-8B	0.66	0.44	0.47	0.59	0.63	0.44	0.48	0.55	0.63	0.45	0.48	0.55
	Mistral-7B	0.49	0.34	0.37	0.31	0.51	0.34	0.40	0.43*	0.50	0.37	0.40	0.47*
	Qwen3-14B	0.72	0.48	0.52	0.60	0.70	0.47	0.50	0.57	0.70	0.48	0.53	0.61
	Gemma3-12B	**0.75**	**0.50**	**0.58**	**0.69**	0.73	**0.53**	0.55	0.58†	0.73	**0.51**	**0.58**	**0.64**
	Gemma3-27B (8-bit)	0.30	0.21	0.54	0.65	**0.75***	0.51*	**0.57**	**0.58†**	**0.77***	**0.51***	0.56	0.60
MT	LLaMA-3.1-8B	0.57	0.42	0.48	0.51	0.56	0.42	0.49	0.48	0.56	0.43	0.48	0.48
	Mistral-7B	0.31	0.30	0.30	0.24	0.48*	0.45*	0.50*	0.48*	0.48*	0.48*	0.50*	0.49*
	Qwen3-14B	**0.60**	0.45	0.50	0.62	0.55	0.49	**0.53**	0.57	**0.62**	0.47	0.51	0.58
	Gemma3-12B	0.59	0.52	0.49	0.57	0.57	**0.51**	0.49	**0.60**	0.55	0.50	0.49	0.57
	Gemma3-27B (8-bit)	0.60	**0.52**	**0.52**	**0.64**	**0.60**	0.50	0.53	0.58	0.62	**0.51**	**0.52**	**0.59**

Table 3: Strict F1 scores for LoRA models with Think OFF under no augmentation, Qwen3 generated augmentation, and Qwen3 generated plus verified augmentation. Bold: strongest per setting and augmentation group. * significant improvement over No Aug baseline; † significant decrease .

Setting	MI	ML	PG	Act
ZS, Think OFF	**0.62**	**0.46**	**0.45**	**0.38**
ZS, Think ON	0.52†	0.46	0.45	0.36
LoRA, Think OFF	**0.72**	**0.48**	**0.52**	**0.60**
LoRA, Think ON	0.22†	0.28†	0.24†	0.29†

Table 4: Strict F1 for Qwen3-14B with Think OFF and Think ON without augmentation. † significant decrease relative to Think OFF.

Set.	Aug.	MI	ML	PG	Act
ST	No Aug	0.22	0.28	0.24	0.29
	Gen	**0.72***	0.50*	**0.51***	0.62*
	Gen+Ver.	0.71*	**0.50***	0.49*	**0.64***
MT	Gen	**0.62***	**0.50***	**0.52***	0.56*
	Gen+Ver.	0.57*	0.48*	0.51*	**0.59***

Table 5: Strict F1 for Qwen3-14B with LoRA and Think ON across augmentation strategies. No Aug MT omitted (outputs collapsed to Unknown). * significant improvement over No Aug.

Paper	Method	MI	ML	PG	Act
BLCU	GPT4o ICL	0.46	0.45	0.49	0.44
BLCU	o3-mini ICL	0.43	0.47	0.49	0.45
BLCU	Gemini 2.5 Pro ICL	0.62	**0.68**	**0.67**	0.56
BLCU	Grok ICL	0.48	0.39	0.43	0.42
BLCU	DeepSeek ICL	0.51	0.51	0.54	0.47
BLCU	Claude ICL	0.45	0.49	0.63	0.43
TutorMind	GPT4o Aug	0.70	–	–	–
bea-jh	GLM SFT	0.72	0.48	0.59	0.54
bea-jh	GLM GRPO	0.56	0.57	0.57	0.66
MSA	Mathstral Ens	0.72	0.57	0.58	**0.69**
Ours	G3-12B NoAug	**0.75***	0.50	**0.58**	**0.69**
Ours	G3-12B Gen	**0.74***	**0.53**	0.55	0.58
Ours	G3-27B GenVer	**0.77***	0.51	0.57	**0.60**

Table 6: Strict F1 comparison against prior BEA 2025 systems. * significant improvement over the best prior systems.

for Gen+Verify, confirming verification roughly doubles the augmentation footprint with limited performance gains.

6 Lessons Learned

1. **Model choice depends on task formulation.** Gemma3-12B is strongest for single-task settings, while Gemma3-27B (8-bit) is most consistent in multitask settings, where larger capacity outweighs quantization costs (Figures 4 and 5). Single-task and multitask objectives favor different model capacities.

2. **Pedagogical dimensions differ in difficulty.** MI is the easiest and most learnable; ML remains hardest despite fine-tuning, augmentation, and scale, as pinpointing error location in multi-step solutions is fundamentally harder than detecting or describing the error. Act starts weak under zero-shot but improves substantially with LoRA, while PG stays stable in the middle (Figures 2–5).

3. **Multitask training improves balance.** Single-task training peaks on individual dimensions, but multitask training yields more balanced predictions by sharing signal across labels (Figures 4 and 5), promoting label consistency at the cost of peak performance on individual objectives.

4. **Augmentation helps selectively.** It strongly benefits constrained models such as Gemma3-27B (8-bit) and Mistral-7B, but yields limited gains for stronger models like Gemma3-12B and Qwen3-14B that already saturate the training signal (Table 3). Augmentation is most valuable when model capacity is constrained relative to the data distribution.

5. **Verification doubles cost without consistent gains.** Gen and GenVer perform similarly, yet GenVer raises generation carbon cost from 2.0 to 4.2 kg CO_2 (Table 3), making self-verification an unnecessary overhead that does not generalize as a reliable quality control mechanism.

6. **CoT+Reasoning is better for data generation than classification.** Think ON causes collapse under LoRA without augmentation, but recovers performance when used to generate data (Tables 4 and 5). Reasoning mode is better exploited as a data construction tool than as a direct classification strategy under parameter-efficient fine-tuning.

7. **Carbon cost shapes practical recommendations.** Larger models and Think ON introduce substantial emissions (Figures 6 and 7). Gemma3 12B has the smallest emissions spike across augmentation conditions, making it the strongest single task choice when both performance and efficiency matter. These findings highlight the need for carbon aware model selection in sustainable NLP research.

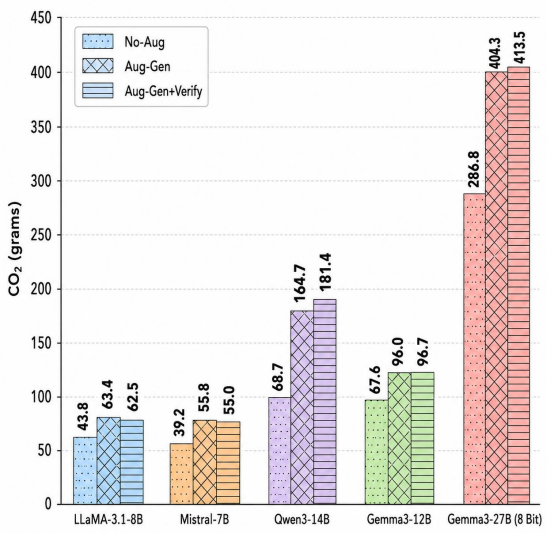

Figure 6: Carbon emissions for LoRA fine-tuning across models and augmentation strategies.

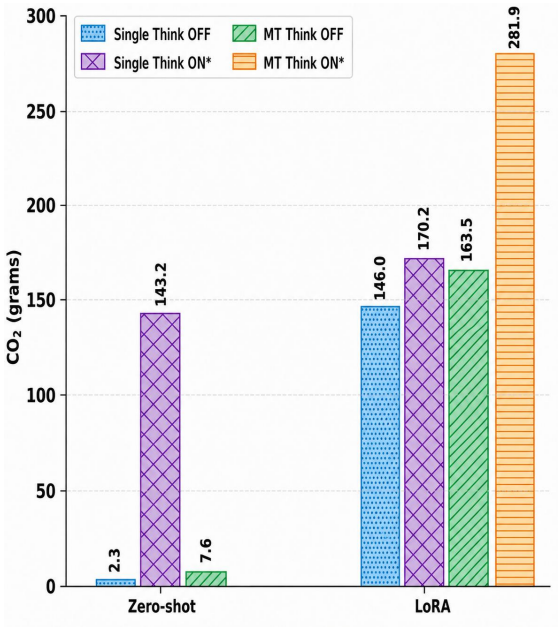

Figure 7: Carbon emissions for Qwen3-14B under different task and reasoning settings.

7 Conclusion

We present a systematic study of open source models for pedagogical ability assessment in AI tutor responses. Across 120 experimental runs covering zero shot inference, LoRA fine tuning, synthetic augmentation, CoT+Reasoning, and single task versus multitask formulations, we show that Gemma3 12B is strongest for single task evaluation, Gemma3 27B (8 bit) is more reliable in multi-task settings, and open source LoRA pipelines can match or surpass proprietary and ensemble based systems on key pedagogical dimensions.

These findings offer practical guidance for building automatic evaluation systems: augmentation, reasoning, and model scale should be used selectively. Augmentation helps constrained models more than those already saturating the training signal, Gen+Ver does not consistently outperform Gen despite higher cost, and CoT+Reasoning is more useful for data generation than direct classification. LoRA fine tuning can also interfere with instruction following under thinking mode, raising broader concerns for reasoning capable models. Carbon analysis shows that stronger performance can carry substantial environmental cost. Future work should explore multilingual data, task weighting, label dependency modeling, and alternative reasoning capable models for augmentation beyond Qwen3. Code and datasets are available at `https://anonymous.4open.science/r/GRADE-D0B3`.

8 Limitations

Dataset size. Although this work expands the original (Dekmak et al., 2025) setting across all four pedagogical dimensions, the dataset remains relatively small for training and evaluating robust LLM based judges. More work is needed from the community to grow this benchmark so that both LLM judges and fine tuned models can learn from broader and more diverse examples.

Using only Qwen3 for augmentation and reasoning. Our augmentation and CoT+Reasoning experiments are centered on Qwen3 14B because it supports both synthetic generation and the Think ON and Think OFF setup used in this study. Since evaluation is performed on untouched validation data, the improvements still reflect genuine downstream gains rather than direct contamination. However, using a single model for both generation and reasoning limits how broadly these findings can be generalized. Future work should test whether similar gains hold with other reasoning capable models, such as DeepSeek R1(DeepSeek-AI, 2025), and with alternative augmentation sources or independent verifier models.

Quantized Gemma3-27B setting. Gemma3-27B is evaluated with 8-bit quantization due to compute constraints. This makes the 27B experiments practical, but it is not a clean full precision comparison against smaller models trained without quantization.

Mistake Location remains challenging. Mistake Location remains the most difficult dimension across our experiments. Its consistently lower scores suggest that the current prompts, training setup, and augmentation strategy still do not fully capture fine grained error localization in student solutions.

Single domain generalizability. Our findings are grounded in math tutoring dialogues (MathDial and Bridge). Whether these conclusions transfer to other educational domains, such as language learning or science tutoring, where error types and pedagogical strategies differ substantially, remains an open question for future work.

References

Jiyuan An, Xiang Fu, Bo Liu, Xuquan Zong, Cunliang Kong, Shuliang Liu, Shuo Wang, Zhenghao Liu, Liner Yang, Hanghang Fan, and Erhong Yang. 2025. BLCU-ICALL at BEA 2025 shared task: Multi-strategy evaluation of AI tutors. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1084–1097.

Benoit Courty, Victor Schmidt, Sasha Luccioni, Goyal-Kamal, Boris Feld, Jérémy Lecourt, Amine Saboni, Mathilde Léval, Luis Blanche, Franklin Zhao, and Aditya Joshi. 2024. CodeCarbon: Estimate and Track Carbon Emissions from Machine Learning Computing.

DeepSeek-AI. 2025. Deepseek-r1. `https://huggingface.co/deepseek-ai/DeepSeek-R1`. Hugging Face model repository.

Fatima Dekmak, Christian Khairallah, and Wissam Antoun. 2025. TutorMind at BEA 2025 shared task: Leveraging fine-tuned LLMs and data augmentation for mistake identification. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, and 1 others. 2024. The Llama 3 herd of models.

Yuming Fan, Chuangchuang Tan, and Wenyu Song. 2025. BJTU at BEA 2025 shared task: Task-aware prompt tuning and data augmentation for evaluating AI math tutors. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 968–988.

Gemma Team. 2025. Gemma 3 technical report.

Sebastian Gombert, Fabian Zehner, and Hendrik Drachsler. 2025. TBA at BEA 2025 shared task: Transfer-learning from DARE-TIES merged models for the pedagogical ability assessment of LLM-powered math tutors. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1173–1179.

Daniel Han, Michael Han, and Unsloth team. 2023. Unsloth.

Baraa Hikal, Mohamed Basem, Islam Abdulhakeem Oshallah, and Ali Hamdi. 2025. MSA at BEA 2025 shared task: Disagreement-aware instruction tuning for multi-dimensional evaluation of LLMs as math tutors. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, and 1 others. 2023. Mistral 7B.

Irina Jurenka, Markus Kunesch, and 1 others. 2024. Towards responsible development of generative AI for education: An evaluation-driven approach.

Ekaterina Kochmar, Kaushal Kumar Maurya, Kseniia Petukhova, KV Aditya Srivatsa, Anaïs Tack, and Justin Vasselli. 2025. Findings of the BEA 2025 shared task on pedagogical ability assessment of AI-powered tutors. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1011–1033.

Charles Koutcheme, Nicola Dainese, Sami Sarsa, Arto Hellas, Juho Leinonen, and Paul Denny. 2024. Open source language models can provide feedback: Evaluating LLMs' ability to help students using GPT-4-as-a-judge.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Jakub Macina, Nico Daheim, Sankalan Chowdhury, Tanmay Sinha, Manu Kapur, Iryna Gurevych, and Mrinmaya Sachan. 2023. MathDial: A dialogue tutoring dataset with rich pedagogical properties grounded in math reasoning problems. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5602–5621.

Kaushal Kumar Maurya, KV Aditya Srivatsa, Kseniia Petukhova, and Ekaterina Kochmar. 2025. Unifying AI tutor evaluation: An evaluation taxonomy for pedagogical ability assessment of LLM-powered AI tutors. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1234–1251.

Geon Park, Jiwoo Song, Gihyeon Choi, Juoh Sun, and Harksoo Kim. 2025. K-NLPers at BEA 2025 shared task: Evaluating the quality of AI tutor responses with GPT-4.1. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1145–1163.

Jihyeon Roh and Jinhyun Bang. 2025. bea-jh at BEA 2025 shared task: Evaluating AI-powered tutors through pedagogically-informed reasoning. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*.

Trishita Saha, Shrenik Ganguli, and Maunendra Sankar Desarkar. 2025. NLIP at BEA 2025 shared task: Evaluation of pedagogical ability of AI tutors. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1242–1253.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, Florence, Italy. Association for Computational Linguistics.

Anaïs Tack, Ekaterina Kochmar, Zheng Yuan, Serge Bibauw, and Chris Piech. 2023. The BEA 2023 shared task on generating AI teacher responses in educational dialogues. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 785–795.

Anaïs Tack and Chris Piech. 2022. The AI teacher test: Measuring the pedagogical ability of blender and GPT-3 in educational dialogues. In *Proceedings of the 15th International Conference on Educational Data Mining*, pages 522–529.

Rose E. Wang, Qingyang Zhang, Carly Robinson, Susanna Loeb, and Dorottya Demszky. 2024. Bridging the novice-expert gap via models of decision-making: A case study on remediating math mistakes. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2174–2199.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, and 1 others. 2025. Qwen3 technical report.

A Prompts

Task	System Prompt

MI	Classify the tutor's response to the student's answer based on whether the tutor has identified a mistake. Use the following labels: *Yes* means the mistake is clearly identified; *No* means the tutor does not recognize the mistake; *To some extent* means the tutor suggests a mistake but is unsure. Respond strictly with exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

ML	Classify the tutor's response to the student's answer based on whether the tutor has correctly located where the student's mistake occurred. Use the following labels: *Yes* means the mistake location is clearly and correctly identified; *No* means the tutor does not locate or pinpoint the mistake at all; *To some extent* means the tutor partially or vaguely locates the mistake. Respond strictly with exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

PG	Classify the tutor's response to the student's answer based on whether the tutor provides useful guidance to help the student correct their mistake. Use the following labels: *Yes* means clear, useful, and relevant guidance is provided; *No* means no meaningful guidance is provided; *To some extent* means some guidance is provided but it is incomplete, vague, or only partially helpful. Respond strictly with exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

Act	Classify the tutor's response to the student's answer based on whether the tutor's response is actionable—i.e., it gives the student something concrete and clear to do next. Use the following labels: *Yes* means the response is clearly actionable with a concrete next step; *No* means the response is not actionable and gives the student nothing to act on; *To some extent* means the response is partially actionable but lacks clarity or completeness. Respond strictly with exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

MT	You are an expert educational evaluator. Given a student-tutor math dialogue and a tutor response, evaluate the tutor response across four pedagogical dimensions. Evaluate each dimension independently:
	`1. Mistake_Identification:` Has the tutor identified the student's mistake?
	`2. Mistake_Location:` Has the tutor correctly located where the mistake occurred?
	`3. Providing_Guidance:` Has the tutor provided useful guidance to correct the mistake?
	`4. Actionability:` Is the tutor's response actionable—does it give the student a concrete next step?
	For each dimension, use one of: *Yes* / *No* / *To some extent*. Respond strictly in the following format:
	`Mistake_Identification: Yes`
	`Mistake_Location: No`
	`Providing_Guidance: To some extent`
	`Actionability: Yes`
	Pick exactly one value per dimension.

Table 7: Standard classification prompts used across all models in zero-shot and LoRA fine-tuning experiments. Single-task prompts produce one line output for the specific task; the multitask prompt produces four dimension-specific lines.

Task	System Prompt

MI	Classify the tutor's response to the student's answer based on whether the tutor has identified a mistake. Think step-by-step: first analyze the student's response to understand what mistake was made, then examine the tutor's response to determine if and how clearly the mistake is identified. Use the following labels: *Yes* means the mistake is clearly identified; *No* means the tutor does not recognize the mistake; *To some extent* means the tutor suggests a mistake but is unsure. After your reasoning, respond strictly with exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

ML	Classify the tutor's response to the student's answer based on whether the tutor has correctly located where the student's mistake occurred. Think step-by-step: first identify where in the student's solution the mistake is, then examine whether the tutor's response pinpoints this exact location. Use the following labels: *Yes* means the mistake location is clearly and correctly identified; *No* means the tutor does not locate or pinpoint the mistake at all; *To some extent* means the tutor partially or vaguely locates the mistake. After your reasoning, respond strictly with exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

PG	Classify the tutor's response to the student's answer based on whether the tutor provides useful guidance to help the student correct their mistake. Think step-by-step: first understand what the student needs to correct their error, then evaluate whether the tutor's response provides relevant and helpful guidance toward that correction. Use the following labels: *Yes* means clear, useful, and relevant guidance is provided; *No* means no meaningful guidance is provided; *To some extent* means some guidance is provided but it is incomplete, vague, or only partially helpful. After your reasoning, respond strictly with exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

Act	Classify the tutor's response to the student's answer based on whether the tutor's response is actionable—i.e., it gives the student something concrete and clear to do next. Think step-by-step: consider what a student would actually do after reading this tutor response—is there a clear next action? Is it specific enough to act on? Use the following labels: *Yes* means the response is clearly actionable with a concrete next step; *No* means the response is not actionable and gives the student nothing to act on; *To some extent* means the response is partially actionable but lacks clarity or completeness. After your reasoning, respond strictly with exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

MT	You are an expert educational evaluator. Given a student-tutor math dialogue and a tutor response, evaluate the tutor response across four pedagogical dimensions. Think step-by-step for each dimension: carefully analyze the dialogue, identify the student's mistake, and then evaluate each dimension independently before giving your final answer.
	`1. Mistake_Identification:` Has the tutor identified the student's mistake?
	`2. Mistake_Location:` Has the tutor correctly located where the mistake occurred?
	`3. Providing_Guidance:` Has the tutor provided useful guidance to correct the mistake?
	`4. Actionability:` Is the tutor's response actionable—does it give the student a concrete next step?
	For each dimension, use one of: *Yes* / *No* / *To some extent*. After your reasoning, respond strictly in the following format:
	`Mistake_Identification: Yes`
	`Mistake_Location: No`
	`Providing_Guidance: To some extent`
	`Actionability: Yes`
	Pick exactly one value per dimension.

Table 8: Chain-of-thought classification prompts used for Qwen3-14B in zero-shot and LoRA with thinking ON. Identical in structure to Table 7 but augmented with explicit step-by-step reasoning instructions before the final label prediction.

Task	Minority Class	Generation Prompt
MI	**No**	You are an expert math tutor generating training data for an AI evaluation system. Given the following student-tutor math conversation, write a single tutor response that does NOT identify the student's mistake at all. The tutor should either ignore the mistake, proceed as if the student is correct, or simply provide the next step without any acknowledgment of an error. The response must be exactly ONE sentence, natural and realistic—something an actual tutor might say. Write only the single-sentence tutor response, nothing else.
MI	**To some extent**	You are an expert math tutor generating training data for an AI evaluation system. Given the following student-tutor math conversation, write a single tutor response that PARTIALLY or VAGUELY suggests the student may have made a mistake, but does not clearly or explicitly identify what the mistake is. The tutor should sound uncertain, exploratory, or cautious. The response must be exactly ONE sentence, natural and realistic—something an actual tutor might say. Write only the single-sentence tutor response, nothing else.
ML	**No**	You are an expert math tutor generating training data for an AI evaluation system. Given the following student-tutor math conversation, write a single tutor response that does NOT locate or pinpoint where the student's mistake occurred. The tutor may acknowledge something is off but gives no indication of where in the solution the error is. The response must be exactly ONE sentence, natural and realistic—something an actual tutor might say. Write only the single-sentence tutor response, nothing else.
ML	**To some extent**	You are an expert math tutor generating training data for an AI evaluation system. Given the following student-tutor math conversation, write a single tutor response that PARTIALLY locates where the student's mistake is, but is vague, imprecise, or only hints at the location without clearly identifying it. The response must be exactly ONE sentence, natural and realistic—something an actual tutor might say. Write only the single-sentence tutor response, nothing else.
PG	**No**	You are an expert math tutor generating training data for an AI evaluation system. Given the following student-tutor math conversation, write a single tutor response that provides NO useful guidance to help the student correct their mistake. The tutor might point out an error but gives the student nothing helpful to act on, or simply restates the problem without direction. The response must be exactly ONE sentence, natural and realistic—something an actual tutor might say. Write only the single-sentence tutor response, nothing else.
PG	**To some extent**	You are an expert math tutor generating training data for an AI evaluation system. Given the following student-tutor math conversation, write a single tutor response that provides SOME guidance but is incomplete, too vague, or only partially helpful. The student would have some direction but not enough to fully correct their mistake. The response must be exactly ONE sentence, natural and realistic—something an actual tutor might say. Write only the single-sentence tutor response, nothing else.
Act	**No**	You are an expert math tutor generating training data for an AI evaluation system. Given the following student-tutor math conversation, write a single tutor response that is NOT actionable—it gives the student nothing concrete to do next. The response might be motivational or general but lacks any specific next step. The response must be exactly ONE sentence, natural and realistic—something an actual tutor might say. Write only the single-sentence tutor response, nothing else.
Act	**To some extent**	You are an expert math tutor generating training data for an AI evaluation system. Given the following student-tutor math conversation, write a single tutor response that is PARTIALLY actionable—it gives the student some direction but the next step is unclear, incomplete, or ambiguous. The response must be exactly ONE sentence, natural and realistic—something an actual tutor might say. Write only the single-sentence tutor response, nothing else.
MT	**No (all dims)**	You are an expert math tutor generating training data for an AI evaluation system. Given the following student-tutor math conversation, write a single tutor response that scores *No* on ALL four pedagogical dimensions: it does not identify the mistake, does not locate it, provides no guidance, and is not actionable. The response must be exactly ONE sentence, natural and realistic—something an actual tutor might say. Write only the single-sentence tutor response, nothing else.

Task	Minority Class	Generation Prompt
MT	**To some extent (all dims)**	You are an expert math tutor generating training data for an AI evaluation system. Given the following student-tutor math conversation, write a single tutor response that scores *To some extent* on ALL four pedagogical dimensions: it vaguely suggests a mistake, partially locates it, gives incomplete guidance, and is only partially actionable. The response must be exactly ONE sentence, natural and realistic—something an actual tutor might say. Write only the single-sentence tutor response, nothing else.

Table 9: Augmentation generation prompts used by Qwen3-14B (thinking ON) to synthesize minority-class examples (*No* and *To some extent*). Responses are constrained to one sentence to match real tutor response style.

Task	Verification Prompt

MI	You are an expert educational evaluator. Given a student-tutor math conversation and a generated tutor response, classify whether the tutor response identifies the student's mistake. Labels: *Yes*: The tutor clearly identifies or recognizes the student's mistake. *No*: The tutor does not recognize or acknowledge any mistake. *To some extent*: The tutor vaguely, cautiously, or partially suggests there may be a mistake but does not clearly identify it. Think carefully, but output only one line using exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

ML	You are an expert educational evaluator. Given a student-tutor math conversation and a generated tutor response, classify whether the tutor response locates where the student's mistake occurred.Labels: *Yes*: The tutor clearly and correctly pinpoints where the mistake occurred. *No*: The tutor does not locate or pinpoint the mistake at all. *To some extent*: The tutor gives only a vague, broad, partial, or imprecise indication of where the mistake may be. Think carefully, but output only one line using exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

PG	You are an expert educational evaluator. Given a student-tutor math conversation and a generated tutor response, classify whether the tutor response provides useful guidance to help the student correct the mistake. Labels: *Yes*: The tutor provides clear, useful, and relevant guidance sufficient to help the student correct the mistake. *No*: The tutor provides no meaningful guidance, hint, strategy, correction, or useful direction. *To some extent*: The tutor provides some guidance, but it is incomplete, vague, under-specified, or only partially helpful. Think carefully, but output only one line using exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

Act	You are an expert educational evaluator. Given a student-tutor math conversation and a generated tutor response, classify whether the tutor response gives the student a concrete next action. Labels: *Yes*: The tutor gives a clear and concrete next step the student can act on. *No*: The tutor gives no concrete action or next step.*To some extent*: The tutor gives a vague, incomplete, or partially actionable next step, but it is not specific enough to be fully actionable. Think carefully, but output only one line using exactly one of the following:
	`Evaluation: Yes`
	`Evaluation: No`
	`Evaluation: To some extent`

Table 10: Self-verification prompts used by Qwen3-14B to filter synthetic examples generated during augmentation. The placeholder `{label}` is replaced at runtime with the intended minority class label. For multitask examples, all four prompts run independently.

B Lenient F1 Results

Tables 11 through 20 report lenient macro-averaged F1 scores across all experimental conditions. G3-12B and G3-27B denote Gemma3-12B and Gemma3-27B (8-bit), respectively. Best value per dimension is bolded in each table.

Model	MI	ML	PG	Act
LLaMA-3.1-8B	0.7662	0.6748	0.6631	0.6265
Mistral-7B	0.7621	0.5393	0.5188	0.4945
Qwen3-14B	0.7747	**0.6813**	0.6332	0.6204
G3-12B	**0.7761**	0.6795	**0.7131**	0.6258
G3-27B	0.7444	0.6645	0.6762	**0.6644**

Table 11: Zero-shot | No-Aug | Think OFF | Single Tasks | Lenient F1.

Model	MI	ML	PG	Act
LLaMA-3.1-8B	0.6962	0.5213	0.5907	0.5067
Mistral-7B	0.5186	0.4821	0.4687	0.4109
Qwen3-14B	0.6990	0.5777	0.7178	0.5766
G3-12B	0.7280	0.6031	0.7234	0.6770
G3-27B	**0.7587**	**0.6630**	**0.7257**	**0.6764**

Table 12: Zero-shot | No-Aug | Think OFF | Multitask | Lenient F1.

Dimension	Think OFF	Think ON
MI	**0.7747**	0.6820
ML	**0.6813**	0.6750
PG	0.6332	**0.6517**
Act	**0.6204**	0.5792

Table 13: Zero-shot | No-Aug | Qwen3-14B | Think OFF vs ON | Single Tasks | Lenient F1. No MT data available for Think ON.

Model	MI	ML	PG	Act
LLaMA-3.1-8B	0.8685	0.7232	0.7307	0.7809
Mistral-7B	0.7056	0.5696	0.5246	0.4702
Qwen3-14B	0.8886	0.7416	0.7400	0.8151
G3-12B	**0.8948**	**0.7605**	**0.7890**	**0.8828**
G3-27B	0.4883	0.3379	0.7443	0.8692

Table 14: LoRA | No-Aug | Think OFF | Single Tasks | Lenient F1.

Model	MI	ML	PG	Act
LLaMA-3.1-8B	0.8455	0.6690	0.7229	0.7767
Mistral-7B	0.4832	0.4897	0.4679	0.4210
Qwen3-14B	0.8081	0.6973	0.7307	0.8372
G3-12B	0.8550	0.7772	0.7734	0.8736
G3-27B	**0.8751**	**0.7848**	**0.7684**	**0.8877**

Table 15: LoRA | No-Aug | Think OFF | Multitask | Lenient F1.

Dimension	Think OFF	Think ON
MI	**0.8886**	0.3373
ML	**0.7416**	0.4368
PG	**0.7400**	0.3763
Act	**0.8151**	0.4300

Table 16: LoRA | No-Aug | Qwen3-14B | Think OFF vs ON | Single Tasks | Lenient F1. No MT data available for No-Aug Think ON.

	No-Aug				Qwen3-Gen				Qwen3-Gen+Verify			
Model	MI	ML	PG	Act	MI	ML	PG	Act	MI	ML	PG	Act
LLaMA-3.1-8B	0.8685	0.7232	0.7307	0.7809	0.8206	0.7131	0.7490	0.7751	0.8297	0.7241	0.7242	0.7692
Mistral-7B	0.7056	0.5696	0.5246	0.4702	0.6882	0.5756	0.6095	0.6018	0.6971	0.6354	0.5943	0.6952
Qwen3-14B	0.8886	0.7416	0.7400	0.8151	0.8576	0.7180	0.7372	0.8137	0.8659	0.7414	0.7406	0.8390
G3-12B	**0.8948**	0.7605	0.7890	**0.8828**	0.8586	0.7637	0.7624	0.8632	0.8766	0.7644	**0.7908**	0.8811
G3-27B	0.4883	0.3379	0.7443	0.8692	0.8734	0.7633	0.7867	0.8484	0.8825	**0.7846**	0.7641	0.8528

Table 17: LoRA | Think OFF | All Augmentation Strategies | Single Tasks | Lenient F1.

	No-Aug				Qwen3-Gen				Qwen3-Gen+Verify			
Model	MI	ML	PG	Act	MI	ML	PG	Act	MI	ML	PG	Act
LLaMA-3.1-8B	0.8455	0.6690	0.7229	0.7767	0.8507	0.6632	0.7395	0.7705	0.8397	0.6698	0.7122	0.7578
Mistral-7B	0.4832	0.4897	0.4679	0.4210	0.7155	0.6346	0.7327	0.6706	0.7135	0.6393	0.7240	0.6732
Qwen3-14B	0.8081	0.6973	0.7307	0.8372	0.8394	0.7162	0.7544	0.8455	0.8569	0.7315	0.7344	0.8381
G3-12B	0.8550	0.7772	**0.7734**	0.8736	0.7793	0.7493	0.7549	**0.8794**	0.8328	0.7870	0.7550	0.8684
G3-27B	**0.8751**	0.7848	0.7684	**0.8877**	0.8564	0.7806	0.7627	0.8735	0.8733	**0.7907**	0.7648	0.8594

Table 18: LoRA | Think OFF | All Augmentation Strategies | Multitask | Lenient F1.

Dimension	No-Aug	Qwen3-Gen	Qwen3-Gen+Verify
MI	0.3373	**0.8699**	0.8598
ML	0.4368	**0.7530**	0.7482
PG	0.3763	0.7544	**0.7581**
Act	0.4300	0.8632	**0.8727**

Table 19: LoRA | Qwen3-14B | Think ON | Single Tasks | Lenient F1. No-Aug Think ON included as baseline.

Dimension	Qwen3-Gen	Qwen3-Gen+Verify
MI	**0.8538**	0.8394
ML	**0.7496**	0.7436
PG	**0.7763**	0.7495
Act	0.8488	**0.8564**

Table 20: LoRA | Qwen3-14B | Think ON | Multitask | Lenient F1. No-Aug MT not available for Think ON.