Simulated Students in Tutoring Dialogues: Substance or Illusion?

Alexander Scarlatos[1], Jaewook Lee[1], Simon Woodhead[2], Andrew Lan[1]
[1]University of Massachusetts Amherst, [2]Eedi
{ajscarlatos,jaewooklee,andrewlan}@cs.umass.edu , simon.woodhead@eedi.co.uk

Abstract

Advances in large language models (LLMs) enable many new innovations in education. However, evaluating the effectiveness of new technology requires real students, which is time-consuming and hard to scale up. Therefore, many recent works on LLM-powered *tutoring* solutions have used *simulated students* for both training and evaluation, often via simple prompting. Surprisingly, little work has been done to ensure or even measure the quality of simulated students. In this work, we formally define the student simulation task, propose a set of evaluation metrics that span linguistic, behavioral, and cognitive aspects, and benchmark a wide range of student simulation methods on these metrics. We experiment on a real-world math tutoring dialogue dataset, where both automated and human evaluation results show that prompting strategies for student simulation perform poorly; supervised fine-tuning and preference optimization yield much better but still limited performance, motivating future work on this challenging task.[1]

1 Introduction

Private tutoring has been shown to be highly effective for helping students learn. Naturally, with the emergence of powerful large language models (LLMs), significant attention has been put on how to effectively use LLMs as tutors. Many LLM-based tutors have already been deployed by learning platforms and companies (Khan Academy, 2023; Carnegie Learning, 2024; OpenAI, 2025; Google, 2024; Anthropic, 2025), aiming at scaling up the impact of on-demand, real-time tutoring.

Efforts in developing LLM-based tutors can be broadly categorized into two types: alignment-based and student-based. The first type relies on aligning LLMs with well-established pedagogical principles, especially on how to respond to student utterances during tutoring dialogues to maximize learning and engagement (Khan Academy, 2023; Scarlatos et al., 2025c; Sonkar et al., 2024b). The second type resorts to training, often via reinforcement learning (RL), with students in-the-loop; since doing RL training with real human students at large scale is difficult, existing works either ask experts to role-play students (Google, 2024) or prompt LLMs to simulate students (Dinucu-Jianu et al., 2025; He-Yueya et al., 2024a; Li et al., 2025; Macina et al., 2023; Zhang et al., 2025b).

However, both of these student simulation approaches have obvious limitations: the former is still not scalable, while the latter requires LLM-based students to behave realistically. Many recent works have shown that LLMs cannot reliably simulate student behavior simply through prompting (Martynova et al., 2025; Sonkar et al., 2023). Moreover, it is challenging to make LLM-based simulated students follow cognitive characteristics of real human students, like the power law of practice (Scarlatos et al., 2025a; Weitekamp et al., 2025), when learning. In addition to these well-known shortcomings, there do not exist many evaluation metrics for realistically simulating students in dialogues, particularly at the turn-level; recent works focus on linguistic features at the population-level (Perczel et al., 2025), or consistency with synthetic profiles at the dialogue-level (Liu et al., 2024b). However, it is difficult to determine if systems developed using simulated students are reliable without thoroughly verifying the simulated students themselves. We discuss prior work in more detail in Appendix A.

Contributions In this paper, we explore the task of creating more realistic LLM-based simulated students using a two-stage approach. First, we identify six high-level dimensions, identified by the learning sciences research community, that define

[1]Our code and data annotations are available at https://github.com/umass-ml4ed/sim-student-eval

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 42349–42385

student behavior in dialogues: 1) dialogue acts, 2) correctness, 3) error-making, 4) knowledge acquisition, 5) language use, and 6) tutors' responses. For each dimension, we develop reference-based automated evaluations to measure the realism and faithfulness of a simulated student response with respect to a ground-truth response. Second, we benchmark performance on a set of strong simulated student methods, including several prompting-based methods, supervised fine-tuning (SFT), and one using multi-objective RL. To the best of our knowledge, our work is the first to 1) develop reference-based metrics for realistic student simulation in dialogues, and 2) examine the ability of LLMs to replicate real student turns in dialogues. We conduct extensive experiments on a real-world dataset with 2,000 dialogues between human tutors and students. Using both automated metrics and human evaluations, we find that: (various styles of) prompting often fail to capture most dimensions of student behavior, SFT significantly improves the alignment of generated student utterances with these dimensions, and RL leads to further improvements. We further show that our automated metrics show strong agreement with human experts.

2 Methodology

We now detail our notations, methodology for simulating students in dialogues, and evaluating the faithfulness of these simulations. A dialogue $d = (s_0, t_1, s_1, \ldots, t_M, s_M)$ is defined as an alternating sequence of student turns, s_i, and tutor turns, t_i, where M is the number of *turn pairs* in the dialogue; s_0 is present if the student initiates the dialogue, and s_M is present if the student ends the dialogue. For a dataset of N real tutor-student dialogues, $\mathcal{D}$, d^n denotes the n-th dialogue. We refer to the textual content of a turn as an "utterance". Dialogues are often grounded in a question, q, that the student is attempting and the tutor is guiding them through. In these cases, the question is associated with a set of *knowledge components (KCs)*, C, i.e., skills that the student must possess to answer the question correctly.

2.1 LLM-based Student Simulation

We simulate realistic student behavior by using an LLM, $\mathcal{M}_S$, to generate a predicted student utterance, $\hat{s}_i$, at each turn i in the dialogue, conditioned on the current dialogue history and any other relevant context. Formally, this process is summarized as $\hat{s}_i \sim \mathcal{M}_S(s|s_{<i}, t_{\leq i}, q, P)$. P represents a prompt, which includes instructions on simulating student behavior, but can also include additional context on the student, such as a *persona*, if available. $\mathcal{M}_S$ can be a pre-trained LLM, but can also be fine-tuned using SFT on a dataset of existing student utterances, minimizing the negative log likelihood of student utterances, i.e., $\mathcal{L} = -\sum_{n=0}^{N}\sum_{i=0}^{M} \log \mathcal{M}_S(s_i^n|s_{<i}^n, t_{\leq i}^n, q^n, P^n)$. We can further refine $\mathcal{M}_S$ using RL in order to optimize for specific properties of student behavior, which we detail later.

2.2 Evaluating Simulated Student Turns

We ground our evaluation metrics in several high-level measurable properties of student utterances across six dimensions. These metrics cover important aspects of students, such as behavior, knowledge, and language use, that enable us to make meaningful comparisons between real and simulated students. Several of the metrics rely on ground-truth labels for real student turns; we use LLM annotation to provide these labels, via GPT-4.1 (OpenAI, 2025a). We provide these prompts in Appendix G and also show the agreement between the labels and human experts in Table 3. Finally, several metrics require fine-tuned LMs, as detailed below. We provide the prompts for these models in Appendix G and additional implementation details in Appendix B.

Dialogue Acts Our first student behavior measure is dialogue acts, a widely studied approach for classifying actions taken in dialogues. We develop a list of five student dialogue acts in Table 6, rooted in prior work (Vail and Boyer, 2014; Hou et al., 2025; McNichols et al., 2025). We define a_i as the act that the student takes in turn s_i and $\hat{a}_i$ as the act in the simulated student turn $\hat{s}_i$. We define the similarity to be binary-valued, i.e., 1 if $a_i = \hat{a}_i$ and 0 otherwise. We prompt an LLM to provide the ground-truth act labels. To reduce the costs of our evaluation metric, we then fine-tune a local LLM, $\mathcal{M}_{\text{act}}$, to classify dialogue acts, trained on the ground-truth act labels. At evaluation time, we use this model to classify the acts of simulated student turns, i.e., $\hat{a}_i \sim \mathcal{M}_{\text{act}}(a|\hat{s}_i, s_{<i}, t_{\leq i})$.

Correctness Another important aspect of student behavior is whether they are responding correctly to questions (or tasks) raised by the tutor during the dialogue, who often "scaffold" problem solving by asking small sub-questions (Azevedo and

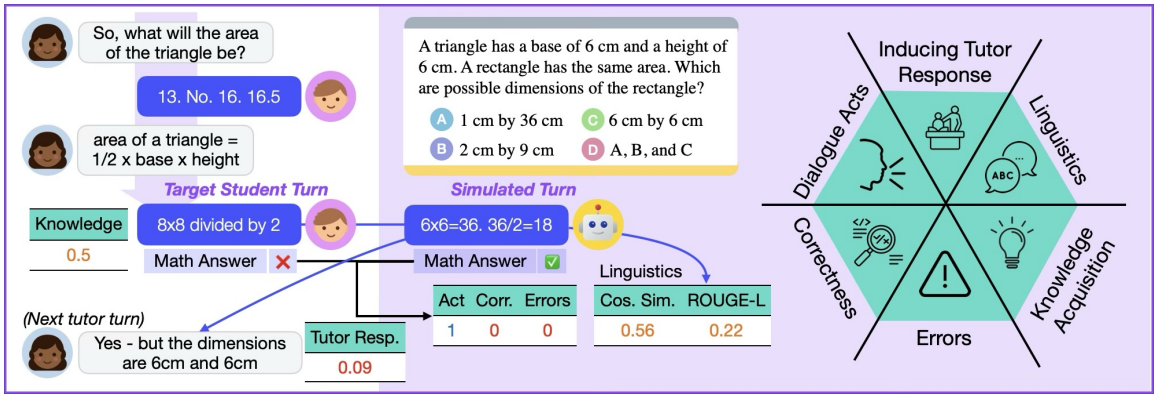

Figure 1: Overview of the seven evaluation metrics for simulated student turn evaluation, with a real paraphrased tutor–student dialogue serving as the reference. In this example, the ground-truth student turn and the simulated turn have the same dialogue act *Math Answer*; however, the target turn is incorrect while the simulated turn is correct.

Hadwin, 2005). Concretely, we define y_i as the correctness of s_i; $y_i = 1$ if s_i is a correct response to the question in the tutor turn t_i, and $y_i = 0$ if it is an incorrect response. Additionally, $y_i = na$ if the tutor does not ask a question in t_i; these turns include off-topic ones or simple utterances that keep the dialogue flowing. Similarly, $\hat{y}_i$ indicates the correctness of the simulated student turn, $\hat{s}_i$. When $y_i \neq na$, we consider the correctness similarity to be 1 if $y_i = \hat{y}_i$ and 0 otherwise. When $y_i = na$, we do not measure correctness similarity. We use an LLM to provide ground-truth correctness labels. We find that classifying correctness by fine-tuning an LLM is much more challenging than doing so for dialogue acts. Instead, we prompt a reasoning-based LLM, GPT-5 mini (OpenAI, 2025b), with low reasoning effort, to evaluate the correctness of simulated student turns, including the ground-truth turn as reference.

Errors In addition to correctness, it is also important to understand the *errors* that students make when they give incorrect answers (King et al., 2024). We define e_i as the error in the ground-truth student turn and $\hat{e}_i$ as the error in the simulated student turn. If the ground-truth turn has an error, we set the error similarity to 1 if $e_i = \hat{e}_i$ and 0 otherwise. We do not evaluate similarity if the ground-truth turn is not incorrect. In practice, we only need to test error equivalence, avoiding the challenge of classifying errors. Therefore, we prompt GPT-5 mini to determine if two turns have the same error. We perform correctness and error evaluation in a single prompt to reduce costs.

Knowledge Acquisition In order for simulated students to be useful for modeling learning outcomes, they need to follow *knowledge acquisition* patterns that are similar to real students (Weitekamp et al., 2025). We formalize knowledge acquisition using knowledge tracing (KT) (Corbett and Anderson, 1994), where knowledge is represented as mastery of KCs. Following Scarlatos et al. (2025a), a framework for KT in tutoring dialogues, we use an LLM-based model to predict student correctness in subsequent turns. Formally, we define $C_i \subseteq C$ as the set of KCs relevant to the tutor turn t_i, which we identify using LLM prompting.

Using these KC labels and the correctness labels, as detailed above, we train a model, $\mathcal{M}_{\text{KT}}$, to estimate a student's current *knowledge state*, which represents the student's current mastery over all KCs in C. We achieve this by training the model to predict if the student will correctly answer the *next* tutor-posed task correctly, conditioned on the dialogue so far. Formally, the estimated probability of a correct answer is $\frac{1}{|C_{i+1}|}\sum_{k=1}^{|C_{i+1}|} z_{i+1,k}$, where $z_{i+1,k} = \mathcal{M}_{\text{KT}}(c_{i+1,k}|s_{\leq i}, t_{\leq i})$ represents the student's estimated mastery of the KC $c_{i+1,k}$.

To measure the *acquisition* of knowledge, we define $\nabla Z_i = \{z_{i,1} - z_{i-1,1}, \ldots, z_{i,|C|} - z_{i-1,|C|}\}$, i.e., how much the student's mastery changes for each KC since the last turn. We similarly compute the change in knowledge for the simulated student turn, $\nabla \hat{Z}_i = \{\hat{z}_{i,1} - z_{i-1,1}, \ldots, \hat{z}_{i,|C|} - z_{i-1,|C|}\}$, where $\hat{z}_{i+1,k} = \mathcal{M}_{\text{KT}}(c_{i+1,k}|\hat{s}_i, s_{<i}, t_{\leq i})$. Since the range of these values can be highly dependent on $\mathcal{M}_{\text{KT}}$ and the data it is trained on, we compare the quantile buckets of mastery deltas rather than the raw values themselves. Specifically, we compute 5 equal sized mass-based quantile bins for all ∇Z_i, where $\text{quant}(z) \in \{0, \ldots, 4\}$ returns the quantile index. We then

define the similarity between ∇Z_i and $\nabla \hat{Z}_i$ as $1 - \sum_{k=1}^{|C|} \frac{|\text{quant}(\nabla Z_{i,k}) - \text{quant}(\nabla \hat{Z}_{i,k})|}{4|C|} \in [0, 1]$, i.e., the inverted average distance between the delta quantiles. We provide a step-by-step computation of knowledge acquisition similarity in Table 12 to illustrate the intuition for this metric.

Linguistics Prior works have found that LLM-based simulated students often fail to reproduce basic linguistic patterns of real students (Martynova et al., 2025). Therefore, we employ existing text-based similarity metrics, specifically the cosine similarity between the text embeddings of the simulated and ground-truth utterances, which measures their semantic similarity, and ROUGE-L (Lin, 2004), which measures word-level recall. Both of these measures capture many aspects of linguistic patterns in student utterances, such as specific word use, utterance length, and sentiment. We use the `Qwen3-Embedding-8B` model (Zhang et al., 2025a) in the cosine similarity calculation.

Inducing Tutor Responses Finally, we measure how well student turns fit into the natural progression of a dialogue. We can examine if the simulated student response is likely to induce the actual response from the tutor, assuming that the tutor would respond similarly if the predicted student utterance is similar to the actual one. Therefore, we leverage the likelihood of the *ground-truth, next tutor turn* conditioned on the simulated student utterance in this turn. We fine-tune an LLM tutor model, $\mathcal{M}_T$, with SFT on the ground-truth tutor utterances and calculate the inverse perplexity of the tutor turn. Formally, this metric is defined as $\exp\big(-\frac{1}{|t_{i+1}|}\log P_{\mathcal{M}_T}(t_{i+1}|\hat{s}_i, s_{<i}, t_{\leq i})\big)$.

2.3 Tuning with RL

Given these automated evaluation metrics, we can create a *reward function* to provide real-time feedback on simulated student responses to improve them via RL. Specifically, we take the average of all metrics to form our final reward, which is possible, since all metrics are in the $[0, 1]$ range; this simple aggregation can form a Pareto-optimal policy on convex Pareto fronts (Lin et al., 2019), and we leave more complex aggregations for future work. In this work, we use offline RL rather than online algorithms to reduce computational burden.

We use a simple four-stage pipeline for offline RL training: we 1) generate n candidate student responses for each dialogue turn in the train set using an SFT-ed student model, 2) evaluate each candidate student response on each of our evaluation metrics, 3) form preference pairs between all possible pairs of the n candidate responses for each turn, where a response is preferred if it has a higher average score across metrics, and 4) train the student model on the resulting preference pairs using direct preference optimization (DPO) (Rafailov et al., 2023). In practice, we employ two techniques for eliminating noisy preference pairs. First, following (Scarlatos et al., 2025b), we only form a pair if the score difference is greater than a threshold, ϵ. Second, we do not train on any of the first 5 turns in a dialogue, since we find that student behavior is highly uncertain early in dialogues, resulting in noisy reward signals.

We note that the purpose of our RL pipeline is to explore the potential of leveraging our metrics to improve simulated student realism. However, we caution that training on LLM-evaluated metrics can lead to reward hacking, where the trained model can learn to overfit to errors in the evaluations, potentially causing them to become unreliable. For this reason, we encourage future works to carry out expert human evaluation in situations when evaluation metrics are leveraged during training.

3 Experiments

We now detail our experimental setup to 1) benchmark a variety of student simulation methods on a real-world dataset and 2) validate our automated evaluation metrics using human evaluation.

3.1 Dataset

We conduct our experiments on Question-Anchored Tutoring Dialogues 2k (Zent et al., 2025), the largest publicly-available dataset of real student-tutor dialogues from the Eedi (2026) learning platform. Each dialogue centers around a middle school student solving a math multiple-choice problem, where a trained tutor guides the student through the process via online chat. Problems span a wide range of topics, including Algebra, Geometry, and Number Sense. After processing, the dataset contains 1,529/382 dialogues in the predefined train/test split, respectively. We further split the train set into a train/validation set with 1,147/382 dialogues, respectively. We train all models on the train set, use the validation set for hyperparameter tuning, and show results on the test set. Additional details are available in Appendix C.

3.2 Simulated Student Benchmarks

We now detail the methods that we benchmark for the student simulation task, including fine-tuning and prompting approaches.

3.2.1 Fine-Tuning

We use both `Llama-3.1-8B-Instruct` and `Llama-3.2-3B-Instruct` (Grattafiori et al., 2024) to examine the impact of model size, and train all models using LoRA (Hu et al., 2022). We discuss training and inference details in Appendix B.1 and show prompts in Appendix G. We perform standard **SFT** on all student turns in the train set, as detailed above in Section 2.1. We then further perform **DPO** training as detailed above in Section 2.3.

3.2.2 Prompting

One of the most common approaches for simulating students is to prompt pre-trained LLMs to behave like real students. Therefore, we test a variety of prompting approaches to see how they compare to fine-tuning. Unless otherwise specified, we prompt GPT-4.1 for each of the following methods. We provide prompts, including persona annotation prompts, in Appendix G, and model decoding details in Appendix B.1. We also provide an example persona, summary, and dialogue retrieval in Table 13.

Zero-Shot We simply instruct the LLM to behave like a real student with a few simple guidelines for student behavior, adapting the prompt from Dinucu-Jianu et al. (2025) with minimal changes.

OCEAN Persona In prompting, we can provide a *persona* to the LLM, instructing it to behave like a student while following the traits described in the persona (He-Yueya et al., 2024a). We use OCEAN (i.e., "Big Five") personas (McCrae and John, 1992), which are commonly used in human user simulation (Liu et al., 2024b; Kim et al., 2025). Specifically, we first prompt an LLM to estimate each of the five OCEAN traits as "low", "neutral", or "high" given the full, ground-truth dialogue. We then provide this estimated persona in the simulated student prompt. We note that this approach does leak information from the ground-truth. If prior student dialogues are available, we can use them to estimate the persona; however, they are not available in the dataset we are using.

Oracle As an upper bound for prompting-based methods, we investigate how well prompted models can simulate student behavior given a summary of the current dialogue. Specifically, we first have an LLM summarize the student's behavior in the dialogue, including OCEAN traits and learning patterns. We then provide this summary in the simulated student prompt, revealing the student's behavior ahead of time.

In-Context Learning (ICL) We also test ICL where an example dialogue is given in the prompt. Adopting the approach in (Lee et al., 2024), we use dialogue summaries to retrieve examples. We first encode the Oracle summaries using `Qwen3-Embedding-8B` and then select the dialogue in the training set whose summary is closest to the summary of the current dialogue to be evaluated. Similar to OCEAN, this approach technically leaks information from the ground-truth and prior dialogues from the same student should be used, if available.

Reasoning Finally, we experiment with a reasoning LLM, GPT-5 mini, with medium reasoning effort. In addition to instructions from the Zero-Shot prompt, we include descriptions of the evaluation criteria that the response will be judged on, letting the model reason about how to generate student-like utterances.

3.3 Evaluation Metrics

We use seven total metrics to evaluate simulated student dialogue turns, as discussed in Section 2.2: **Acts** for dialogue acts, **Corr.** and **Errors** for the correctness and errors in student utterances, **Knowledge** for knowledge acquisition patterns, **Cos. Sim.** (cosine similarity) and **ROUGE-L** for linguistic patterns, and **Tutor Resp.** for the likelihood of inducing tutor responses.

3.4 Human Evaluation

To gauge the validity of our dialogue turn-level student simulation performance metrics, we conduct an IRB-approved human evaluation with three evaluators experienced in math teaching or tutoring, recruited from Upwork[2]. In total, we collect annotations on 190 turns across 38 dialogues, with 20 turns shared across two evaluators for inter-rater agreement. See Appendix E for additional details.

For each dialogue, an evaluator examines five consecutive student turns in a dialogue, first for ground-truth responses and then for simulated ones.

[2] `https://www.upwork.com/`

Method	Acts	Corr.	Errors	Knowledge	Cos. Sim.	ROUGE-L	Tutor Resp.
Fine-Tuning Methods							
SFT (Llama 3.2 3B)	0.6645	0.5557	0.0815	0.8726	0.7331	0.3058	0.2033
DPO (Llama 3.2 3B)	0.6762	0.5748	0.0584	0.8745	0.7345	0.3109	0.2037
SFT (Llama 3.1 8B)	0.6671	0.5670	0.0661	0.8766	0.7383	**0.3212**	0.2038
DPO (Llama 3.1 8B)	**0.6840**	0.5761	0.0529	**0.8787**	**0.7390**	**0.3212**	**0.2039**
Prompting Methods							
Zero-Shot (GPT 4.1)	0.4998	0.5926	0.0220	0.8078	0.5460	0.1648	0.1911
OCEAN (GPT 4.1)	0.5268	0.6039	0.0308	0.8135	0.5739	0.1772	0.1942
ICL (GPT 4.1)	0.5085	0.5991	0.0319	0.8138	0.5919	0.1939	0.1914
Reasoning (GPT 5 Mini)	0.5755	0.5870	0.0088	0.8395	0.5992	0.2170	0.1909
Oracle (GPT 4.1)	0.5097	**0.6755**	**0.1872**	0.8063	0.6032	0.2109	0.1942

Table 1: Results on turn-level student utterance prediction. Prompting-based methods perform well on correctness, while fine-tuned methods perform well on acts, knowledge, linguistic similarity, and tutor responses. All methods, other than Oracle, perform poorly on error prediction. Best method is **bolded** and second best is underlined.

We include simulated responses from DPO trained on the 8B model, Zero-Shot, and Oracle, which cover a wide range of performance levels across the metrics. We inform the evaluator whether a turn is ground-truth, and randomly shuffle the order of simulated turns to not reveal which method produced which response. We collect annotations after the fifth turn in each dialogue to provide sufficient context. For both ground-truth and simulated turns, we ask evaluators to label the dialogue act and correctness. For simulated turns that they label as incorrect, we also ask them to evaluate if they have the same error as the corresponding ground-truth turn. Finally, for simulated turns, we ask evaluators to rate linguistic similarity to the ground-truth on a 5-point Likert scale. We compute 1) the human-evaluated scores for acts, correctness, errors and linguistic similarity, 2) agreement between these four categories and the corresponding automated metrics, 3) agreement between human labels and LLM-assigned labels for acts and correctness on ground-truth turns, and 4) agreement between evaluators on the shared set of turns.

4 Results

We now detail our experimental results, including quantitative results through automated and human evaluation, a qualitative analysis of simulated student utterances, and finally an ablation study.

4.1 Quantitative Results

Table 1 shows quantitative results for turn-level student simulation. We see that fine-tuning methods generally outperform prompting methods, with significant improvements on the Acts, Knowledge, Cos. Sim., ROUGE-L, and Tutor Resp. metrics. They also perform better on Errors, with the exception of Oracle, which contains leaked information on exact errors made by students in its prompt. While prompting methods perform better on the correctness metric, this can be attributed to mostly generating correct responses, the majority class, as seen in Figure 3. These trends indicate that while LLM prompting can anticipate some high-level behavior of the student, fine-tuning is required to capture more nuanced details.

Different prompting-based methods have clearly different strengths and weaknesses. As expected, Zero-Shot performs the worst, showing the importance of context in the prompt. Between OCEAN and ICL, OCEAN performs better on Acts, which can be explained by the OCEAN persona containing high-level behavioral traits for the student. ICL performs better on linguistic metrics, which can be explained by the model reflecting student language patterns in the example dialogues. The Reasoning method performs significantly better than other prompting approaches on several metrics, including Acts, Knowledge, and ROUGE-L. This result shows that reasoning clearly helps, although less so than fine-tuning. Finally, while Oracle outperforms all other methods on Correctness and Errors due to information leakage in the summaries, it does not perform very well on other metrics. This result shows that LLM prompting is highly limited for student simulation; even with a "cheat sheet" in the prompt, the model cannot outperform much smaller, fine-tuned models on most metrics.

Perhaps surprisingly, we find that DPO only slightly outperforms SFT on all metrics, with even

slightly worse performance on Errors, which is possibly due to a sparse reward signal: since candidate turns are sampled from the SFT model, which does not perform well on errors either, there are not enough positive samples in the data for RL training. Similarly, we find that the larger 8B model outperforms the smaller 3B model by small but consistent margins on all metrics except errors. We postulate that this result is due to the inherent difficulty of predicting what students will do next, especially in open-ended settings like dialogues. More advanced techniques, such as reasoning, online RL, or data augmentation, may be needed to further improve the performance of RL, which are possible directions for future work.

4.2 Human Evaluation

Table 2 shows the performance of different simulated student methods using human-evaluated proxies of our evaluation metrics. Mostly, results match the patterns on automated metrics: DPO performs best on Acts and Linguistic, while Oracle performs best on Correctness and Errors. Across Acts, Correctness, and Errors, the values tend to be higher than on automated metrics, though this result is likely due to human evaluation being performed on later dialogue turns; earlier turns are more challenging, as seen in Figure 4. DPO also performs slightly better than Zero-Shot on Correctness, whereas on the automated metrics it is slightly worse. The likely cause is that annotators are more likely to label Zero-Shot responses as na, possibly due to their verbosity. Overall, these results confirm that fine-tuning leads to more realistic student simulations than prompting, although having an Oracle in the prompt helps. However, we note that even the best-performing methods are poor, although it is currently unclear what the theoretical upper bound for performance on our metrics is (the upper bound should be less than 1 on each metric due to inherent randomness in student behavior). Therefore, it is important to study more advanced techniques for realistically simulating students and predicting student behavior in future work.

Metric and Label Agreement Table 3 shows agreement-based reliability results for automated evaluation, LLM annotation, and human evaluation. For Acts, Correctness, and Errors, we report Cohen's Kappa, and for Linguistic, we report Pearson's correlation coefficient. The agreement between human-assigned scores and our automated metrics is very high across labels, indicating that our automated metrics give a reliable measure of student simulation quality. The Errors agreement is slightly lower because 1) it is only computed on incorrect turns, resulting in a smaller sample size, and 2) labels are highly imbalanced, with most simulated turns not making the same error as the ground-truth turn. There is also substantial agreement on the ground-truth turns between human-assigned labels and LLM-assigned labels for Acts and Correctness, showing that our prompting-based method for this annotation is reliable. Finally, we find that inter-rater agreement is very high for Correctness, Errors, and Linguistic. The moderate agreement for acts is primarily due to a single label, Math Answer, being selected much more frequently than the other labels (71- 83% across annotators).

Method	Acts	Corr.	Errors	Linguistic
DPO	**0.7905**	0.6377	0.0612	**0.5405**
Zero-Shot	0.6143	0.6087	0.0408	0.3155
Oracle	0.6476	**0.7101**	**0.2449**	0.4071

Table 2: Human evaluation results, where trends roughly match those on automated metrics.

Pairing	Acts	Corr.	Errors	Linguistic
Hum.-Metric	0.7337	0.6891	0.6127	0.7397
Hum.-Anno.	0.7993	0.7219	–	–
Hum.-Hum.	0.4978	0.7187	0.6154	0.6910

Table 3: Agreement between human evaluators and automated metrics, human evaluators and LLM-assigned annotations, and inter-rater agreement.

4.3 Qualitative Analysis

We perform a qualitative analysis of model outputs to understand the strengths and weaknesses of each simulated student method. We show example simulated utterances from each method in Table 11, along with scores on each metric. We show the distributions of dialogue acts and correctness for each method in Figures 2 and 3, respectively. We also break down the performance of each method over turns as the dialogue progresses in Figure 4.

Generally, there are clear differences between fine-tuning and prompting methods. First, we find that fine-tuned models match the linguistic style of real students much more closely. They produce much shorter outputs than prompting methods (2.28 words on average for DPO 8B vs. 10.89 for

Reward	Acts	Corr.	Errors	Knowledge	Cos. Sim.	ROUGE-L	Tutor Resp.
SFT	0.6795	0.5546	0.0506	0.8723	0.7417	0.3155	0.2102
Average	**0.6962**	0.5699	0.0562	0.8691	0.7433	**0.3181**	0.2104
Acts	0.6949	0.5677	0.0506	0.8695	0.7389	0.3111	0.2101
Corr.	0.6692	**0.5852**	0.0506	0.8652	0.7275	0.3129	0.2081
Errors	0.6795	0.5524	0.0562	0.8708	0.7406	0.3122	0.2105
Knowledge	0.6846	0.5437	**0.0730**	0.8756	**0.7486**	0.3147	**0.2124**
Cos. Sim.	0.6731	0.5371	**0.0730**	0.8713	0.7453	0.2957	0.2116
Tutor Resp.	0.6897	0.5349	0.0618	**0.8763**	0.7447	0.3137	0.2113

Table 4: Results of reward function ablation study. Most reward functions result in high performance on the corresponding metric. Best method is **bolded** and second best is underlined.

ICL and 4.11 for real students), echoing the findings in Perczel et al. (2025). Prompting methods also consistently use typical LLM-like language features such as formal grammar and punctuation; for example, when a tutor asks "I'll leave you to enter your answer now if you're happy with this?", the Reasoning method generates "Yes, thanks — I'll enter D.", while SFT generates "yes thank you" and the real student simply writes "yes". Beyond these surface-level features, we find that fine-tuned responses follow act and correctness distributions that are much more similar to real students; prompting methods overestimate the rate of Seek Information and correct responses, and underestimate the rate of conversational acts, like Acknowledge and Off-Topic. While all methods perform poorly early in dialogues due to limited context, prompting methods suffer much more, while fine-tuning methods perform better by leveraging what they learned from real dialogues through training.

However, there are several limitations to the fine-tuning methods as well. First, they tend to write very short responses, and rarely add in typos or excessive punctuation that are found in real student responses. Moreover, all methods struggle to adapt to individual student behavior. For example, some students give very brief responses, some are very verbose, and some convey a consistent sense of helplessness; however, model response patterns tend to be uniform across students, conveying a lack of diversity. Future work can seek to address these challenges by encouraging diverse outputs in training, or by conditioning on prior dialogues or student personas if they are available, to better capture the nuance in student utterances. See Appendix D for additional discussions.

4.4 Ablation Study

In order to understand how each metric is related to overall student simulation quality, we run an ablation where we train DPO with rewards from only one metric at a time. We use `Llama-3.2-3B-Instruct` for this experiment and evaluate on a random 20% split of the test data to reduce costs. We show results in Table 4. As expected, we see that training on a metric results in high performance on it. The exception is Errors, since the reward signal is likely too sparse to enable learning. Interestingly, training on some metrics induces very high or low performance on completely different metrics. This result may be attributed to the inherent *correlation* between some metrics. For example, training on Knowledge results in the highest performance on Errors, Cos. Sim., and Tutor Resp., yet leads to low performance on Correctness. On the contrary, training on Correctness results in very high performance on Correctness but the lowest performance for most other metrics. While most relationships between aspects of student behavior are as expected, there are several surprising ones, such as between knowledge acquisition and correctness. This result reflects recent findings that training on certain human behavioral data can have unintended effects on seemingly unrelated traits (Chen et al., 2025). Finally, we observe that training on the Average reward generally leads to high performance across metrics, although suffers on Knowledge, Errors, and Tutor Resp. This result suggests that while averaging is a reasonable approach, more advanced techniques for combining metrics could lead to higher overall performance.

5 Conclusions and Future Work

In this paper, we introduce the task of simulating student utterances in tutoring dialogues. We propose a suite of metrics that measure realism of a simulated student turn with respect to a reference student turn. We benchmark a wide range of fine-tuned and prompting-based methods, finding that

while fine-tuning and RL outperform prompting, there is a long way to go before LLMs can fully resemble real student behavior in dialogues.

There are many avenues for future work. First, since we use simple RL methods, it is possible that more advanced techniques leveraging reasoning or data augmentation would result in significantly better performance. Second, due to data constraints, we evaluate each dialogue in isolation. However, with available data, future work could include prior student information, such as prior dialogues or knowledge states, for more context in such simulations. Third, since we only explore simulation realism at the turn-level, future work should further extend our metrics to measure realism at the dialogue-level. Fourth, while we have shown that our metrics are reliable *reference-based* measures of student behavior, future work should investigate *reference-free* metrics for settings where ground-truth data is not available. Finally, since we only explore dialogues based on math problems, future work should extend to dialogues in other domains, such as computer science and language learning.

Acknowledgments

This work is partially supported by Renaissance Philanthropy under the learning engineering virtual institute (LEVI) initiative and the NSF under grants 2153481 and 2237676.

Limitations

There are several practical limitations to our work. First, we only perform our experiments on a single math dialogue dataset, and it remains to be seen if our results generalize to other datasets or domains. Second, our metrics are exclusively reference-based, limiting their use to evaluation settings that are based on existing tutor-student dialogues. Third, we do not evaluate two of our metrics in the human evaluation, Knowledge Acquisition and Inducing Tutor Responses, since we believe they are highly subjective from a human perspective and therefore difficult to reliably measure in a human experiment. However, as a result, we do not have agreement between these metrics and expert ratings. Fourth, we do not measure affect or emotional state in dialogues. While student turns in our dataset give minimal signals on affect, they may be important in other domains, and should be studied in future work. Fifth, we acknowledge that our annotations and our Correctness and Errors metrics rely on proprietary LLMs. We experimented with smaller open-source LLMs in preliminary experiments and found them to be significantly less reliable. Finally, we acknowledge that RL performs only slightly better than SFT, which is perhaps surprising. While we postulate that this result is due to the inherent difficulty of the task, future work will need to develop more advanced RL methods to verify this claim.

Ethical Considerations

There are several potential societal benefits associated with our work. Simulated students have a high potential to improve education broadly; human tutors can use them for low-stakes practice, they can be used in A/B tests to avoid risks to real students and reduce costs, and AI tutors can be trained with simulated students in the loop. Furthermore, reliable *evaluations* will accelerate research on simulated students, allowing researchers to rapidly validate the strengths and weaknesses of their methods and avoid potentially deploying unreliable models in real educational settings. There are also several potential societal risks associated with our work. Simulations may be biased towards certain demographic groups due to bias in training data or inherent bias in language models. As a result, the benefits of simulated students may be diminished for students in underrepresented groups, or worse, cause unintended harm to these groups. We recommend that any simulated student methods used in real educational settings should first be evaluated across demographic groups and thoroughly evaluated for bias. Furthermore, we caution that over-reliance on simulated students could lead to a reduction in the quality of educational tools, for example, if AI tutors are trained using simulated students that do not represent real student distributions. We recommend that any educational AI tools are first thoroughly A/B tested with real students before widespread deployment to avoid harming learning outcomes for students.

References

Marwa Abdulhai, Ryan Cheng, Donovan Clay, Tim Althoff, Sergey Levine, and Natasha Jaques. 2025. Consistently simulating human personas with multi-turn reinforcement learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

John R Anderson, Albert T Corbett, Kenneth R

Koedinger, and Ray Pelletier. 1995. Cognitive tutors: Lessons learned. *The journal of the learning sciences*, 4(2):167–207.

Anthropic. 2025. Introducing claude for education. Online: https://www.anthropic.com/news/introducing-claude-for-education.

Roger Azevedo and Allyson F Hadwin. 2005. Scaffolding self-regulated learning and metacognition–implications for the design of computer-based scaffolds. *Instructional science*, 33(5/6):367–379.

Luca Benedetto, Giovanni Aradelli, Antonia Donvito, Alberto Lucchetti, Andrea Cappelli, and Paula Buttery. 2024. Using LLMs to simulate students' responses to exam questions. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 11351–11368, Miami, Florida, USA. Association for Computational Linguistics.

Jie Cao, Ha Nguyen, Selim Yavuz, Boran Yu, Shuguang Wang, Pavneet Kaur Bharaj, and Dionne Cross Francis. 2026. Developing authentic simulated learners for mathematics teacher learning: Insights from three approaches with large language models. *arXiv preprint arXiv:2604.04361*.

Carnegie Learning. 2024. Livehint overview. Online: https://support.carnegielearning.com/help-center/math/livehint/article/livehint-overview/.

Runjin Chen, Andy Arditi, Henry Sleight, Owain Evans, and Jack Lindsey. 2025. Persona vectors: Monitoring and controlling character traits in language models. *Preprint*, arXiv:2507.21509.

Albert Corbett and John Anderson. 1994. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Model. User-adapted Interact.*, 4(4):253–278.

Nico Daheim, Jakub Macina, Manu Kapur, Iryna Gurevych, and Mrinmaya Sachan. 2024. Stepwise verification and remediation of student reasoning errors with large language model tutors. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8386–8411, Miami, Florida, USA. Association for Computational Linguistics.

David Dinucu-Jianu, Jakub Macina, Nico Daheim, Ido Hakimi, Iryna Gurevych, and Mrinmaya Sachan. 2025. From problem-solving to teaching problem-solving: Aligning LLMs with pedagogy using reinforcement learning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 272–292, Suzhou, China. Association for Computational Linguistics.

Yao Dou, Michel Galley, Baolin Peng, Chris Kedzie, Weixin Cai, Alan Ritter, Chris Quirk, Wei Xu, and Jianfeng Gao. 2025. SimulatorArena: Are user simulators reliable proxies for multi-turn evaluation of AI assistants? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 35212–35290, Suzhou, China. Association for Computational Linguistics.

Zhangqi Duan, Nigel Fernandez, Arun Balajiee Lekshmi Narayanan, Mohammad Hassany, Rafaella Sampaio de Alencar, Peter Brusilovsky, Bita Akram, and Andrew Lan. 2025. Automated knowledge component generation for interpretable knowledge tracing in coding problems. *Preprint*, arXiv:2502.18632.

Eedi. 2026. Eedi labs. https://www.eedi.com/. Accessed: 2026-01-05.

Nigel Fernandez, Alexander Scarlatos, Wanyong Feng, Simon Woodhead, and Andrew Lan. 2024. DiVERT: Distractor generation with variational errors represented as text for math multiple-choice questions. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 9063–9081, Miami, Florida, USA. Association for Computational Linguistics.

Google. 2024. How generative ai expands curiosity and understanding with learnlm. Online: https://blog.google/outreach-initiatives/education/google-learnlm-gemini-generative-ai/.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, and 7 others. 2020. Array programming with NumPy. *Nature*, 585(7825):357–362.

Joy He-Yueya, Noah D. Goodman, and Emma Brunskill. 2024a. Evaluating and optimizing educational content with large language model judgments. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 68–82.

Joy He-Yueya, Wanjing Anya Ma, Kanishk Gandhi, Benjamin W Domingue, Emma Brunskill, and Noah D Goodman. 2024b. Psychometric alignment: Capturing human knowledge distributions via language models. *arXiv preprint arXiv:2407.15645*.

Xinying Hou, Carol Forsyth, Jessica Andrews-Todd, James Rice, Zhiqiang Cai, Yang Jiang, Diego Zapata-Rivera, and Art Graesser. 2025. An llm-enhanced multi-agent architecture for conversation-based assessment. In *Artificial Intelligence in Education: 26th International Conference, AIED 2025, Palermo, Italy, July 22–26, 2025, Proceedings, Part II*, page 119–134, Berlin, Heidelberg. Springer-Verlag.

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*.

Fareya Ikram, Alexander Scarlatos, and Andrew Lan. 2025. Exploring LLMs for predicting tutor strategy and student outcomes in dialogues. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pages 765–779, Vienna, Austria. Association for Computational Linguistics.

Eylül Ipçi, Tanya Nazaretsky, and Tanja Käser. 2025. Leveraging knowledge profiles and generative ai for realistic student response generation. In *Artificial Intelligence in Education. Posters and Late Breaking Results, Workshops and Tutorials, Industry and Innovation Tracks, Practitioners, Doctoral Consortium, Blue Sky, and WideAIED*, pages 143–151, Cham. Springer Nature Switzerland.

Hyoungwook Jin, Seonghee Lee, Hyungyu Shin, and Juho Kim. 2024. Teach ai how to code: Using large language models as teachable agents for programming education. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, pages 1–28.

Damjan Kalajdzievski. 2023. A rank stabilization scaling factor for fine-tuning with lora. *arXiv preprint arXiv:2312.03732*.

Tanja Käser and Giora Alexandron. 2024. Simulated learners in educational technology: A systematic literature review and a turing-like test. *International Journal of Artificial Intelligence in Education*, 34(2):545–585.

Khan Academy. 2023. Supercharge your teaching experience with khanmigo. Online: https://www.khanmigo.ai/.

Jiho Kim, Junseong Choi, Woosog Chay, Daeun Kyung, Yeonsu Kwon, Yohan Jo, and Edward Choi. 2025. Propersim: Developing proactive and personalized ai assistants through user-assistant simulation. *Preprint*, arXiv:2509.21730.

Jules King, L Burleigh, Simon Woodhead, Panagiota Kon, Perpetual Baffour, Scott Crossley, Walter Reade, and Maggie Demkin. 2024. Eedi - mining misconceptions in mathematics. https://kaggle.com/competitions/eedi-mining-misconceptions-in-mathematics. Kaggle.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Daeun Kyung, Hyunseung Chung, Seongsu Bae, Jiho Kim, Jae Ho Sohn, Taerim Kim, Soo Kyung Kim, and Edward Choi. 2025. Patientsim: A persona-driven simulator for realistic doctor-patient interactions. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Seanie Lee, Jianpeng Cheng, Joris Driesen, Alexandru Coca, and Anders Johannsen. 2024. Effective and efficient conversation retrieval for dialogue state tracking with implicit text summaries. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 96–111, Mexico City, Mexico. Association for Computational Linguistics.

Haoxuan Li, Jifan Yu, Xin Cong, Yang Dang, Daniel Zhang-Li, Lu Mi, Yisi Zhan, Huiqin Liu, and Zhiyuan Liu. 2025. Which type of students can llms act? investigating authentic simulation with graph-based human–ai collaborative system. *Investigating Authentic Simulation with Graph-based Human–AI Collaborative System*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Xi Lin, Hui-Ling Zhen, Zhenhua Li, Qing-Fu Zhang, and Sam Kwong. 2019. Pareto multi-task learning. *Advances in neural information processing systems*, 32.

Jiayu Liu, Zhenya Huang, Tong Xiao, Jing Sha, Jinze Wu, Qi Liu, Shijin Wang, and Enhong Chen. 2024a. SocraticLM: Exploring socratic personalized teaching with large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zhengyuan Liu, Stella Xin Yin, Geyu Lin, and Nancy F. Chen. 2024b. Personality-aware student simulation for conversational intelligent tutoring systems. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 626–642, Miami, Florida, USA. Association for Computational Linguistics.

Xinyi Lu and Xu Wang. 2024. Generative students: Using llm-simulated student profiles to support question item evaluation. In *Proceedings of the Eleventh ACM conference on learning@ Scale*, pages 16–27.

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

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, Benjamin Bossan, and Marian Tietz. 2022. PEFT: State-of-the-art parameter-efficient fine-tuning methods. https://github.com/huggingface/peft.

Julia M Markel, Steven G Opferman, James A Landay, and Chris Piech. 2023. Gpteach: Interactive ta training with gpt-based students. In *Proceedings of the tenth acm conference on learning@ scale*, pages 226–236.

Daria Martynova, Jakub Macina, Nico Daheim, Nilay Yalcin, Xiaoyu Zhang, and Mrinmaya Sachan. 2025. Can LLMs effectively simulate human learners? teachers' insights from tutoring LLM students. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pages 100–117, Vienna, Austria. Association for Computational Linguistics.

Noboru Matsuda, William W Cohen, and Kenneth R Koedinger. 2015. Teaching the teacher: tutoring simstudent leads to more effective cognitive tutor authoring. *International Journal of Artificial Intelligence in Education*, 25(1):1–34.

Robert R McCrae and Oliver P John. 1992. An introduction to the five-factor model and its applications. *Journal of personality*, 60(2):175–215.

Hunter McNichols, Fareya Ikram, and Andrew Lan. 2025. The studychat dataset: Student dialogues with chatgpt in an artificial intelligence course. *Preprint*, arXiv:2503.07928.

Bang Nguyen, Tingting Du, Mengxia Yu, Lawrence Angrave, and Meng Jiang. 2025. QG-SMS: Enhancing test item analysis via student modeling and simulation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 26152–26168, Vienna, Austria. Association for Computational Linguistics.

OpenAI. 2025a. Introducing gpt-4.1 in the api. Online: https://openai.com/index/gpt-4-1/.

OpenAI. 2025b. Introducing gpt-5. Online: https://openai.com/index/introducing-gpt-5/.

OpenAI. 2025. Introducing study mode. Online: https://openai.com/index/chatgpt-study-mode/.

Sitong Pan, Robin Schmucker, Bernardo Garcia Bulle Bueno, Salome Aguilar Llanes, Fernanda Albo Alarcón, Hangxiao Zhu, Adam Teo, and Meng Xia. 2025. Tutorup: What if your students were simulated? training tutors to address engagement challenges in online learning. In *Proceedings of the 2025 CHI conference on human factors in computing systems*, pages 1–18.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, and 1 others. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Janos Perczel, Jin Chow, and Dorottya Demszky. 2025. Teachlm: Post-training llms for education using authentic learning data. *arXiv preprint arXiv:2510.05087*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Alexis Ross and Jacob Andreas. 2024. Toward in-context teaching: Adapting examples to students' misconceptions. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13283–13310, Bangkok, Thailand. Association for Computational Linguistics.

Alexis Ross and Jacob Andreas. 2025. Learning to make mistakes: Modeling incorrect student thinking and key errors. *Preprint*, arXiv:2510.11502.

Alexis Ross, Megha Srivastava, Jeremiah Blanchard, and Jacob Andreas. 2025. Modeling student learning with 3.8 million program traces. *arXiv preprint arXiv:2510.05056*.

Alexander Scarlatos, Ryan S. Baker, and Andrew Lan. 2025a. Exploring knowledge tracing in tutor-student dialogues using llms. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, LAK '25, page 249–259, New York, NY, USA. Association for Computing Machinery.

Alexander Scarlatos, Nigel Fernandez, Christopher Ormerod, Susan Lottridge, and Andrew Lan. 2025b. SMART: Simulated students aligned with item response theory for question difficulty prediction. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 25082–25105, Suzhou, China. Association for Computational Linguistics.

Alexander Scarlatos, Naiming Liu, Jaewook Lee, Richard Baraniuk, and Andrew Lan. 2025c. Training llm-based tutors to improve student learning outcomes in dialogues. In *International Conference on Artificial Intelligence in Education*, pages 251–266. Springer.

Robin Schmucker, Meng Xia, Amos Azaria, and Tom Mitchell. 2024. Ruffle&riley: Insights from designing and evaluating a large language model-based conversational tutoring system. In *International Conference on Artificial Intelligence in Education*, pages 75–90. Springer.

Paras Sharma and Qichang Li. 2024. Designing simulated students to emulate learner activity data in an open-ended learning environment. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 986–989, Atlanta, Georgia, USA. International Educational Data Mining Society.

Jiajia Song, Zhihan Guo, and Jionghao Lin. 2026. Simulating novice students using machine unlearning and relearning in large language models. *arXiv preprint arXiv:2603.26142*.

Shashank Sonkar, Xinghe Chen, Naiming Liu, Richard G Baraniuk, and Mrinmaya Sachan. 2024a. Llm-based cognitive models of students with misconceptions. *arXiv preprint arXiv:2410.12294*.

Shashank Sonkar, Naiming Liu, Debshila Mallick, and Richard Baraniuk. 2023. Class: A design framework for building intelligent tutoring systems based on learning science principles. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1941–1961.

Shashank Sonkar, Kangqi Ni, Sapana Chaudhary, and Richard Baraniuk. 2024b. Pedagogical alignment of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 13641–13650.

Alexandria Katarina Vail and Kristy Elizabeth Boyer. 2014. Identifying effective moves in tutoring: On the refinement of dialogue act annotation schemes. In *Intelligent Tutoring Systems*, pages 199–209, Cham. Springer International Publishing.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, and 16 others. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. 2020. Trl: Transformer reinforcement learning. `https://github.com/huggingface/trl`.

Jian Wang, Yinpei Dai, Yichi Zhang, Ziqiao Ma, Wenjie Li, and Joyce Chai. 2025. Training turn-by-turn verifiers for dialogue tutoring agents: The curious case of LLMs as your coding tutors. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 12416–12436, Vienna, Austria. Association for Computational Linguistics.

Junling Wang, Jakub Macina, Nico Daheim, Sankalan Pal Chowdhury, and Mrinmaya Sachan. 2024. Book2Dial: Generating teacher student interactions from textbooks for cost-effective development of educational chatbots. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9707–9731, Bangkok, Thailand. Association for Computational Linguistics.

Daniel Weitekamp, Momin N. Siddiqui, and Christopher J. MacLellan. 2025. Tutorgym: A testbed for evaluating ai agents as tutors and students. In *Artificial Intelligence in Education: 26th International Conference, AIED 2025, Palermo, Italy, July 22–26, 2025, Proceedings, Part III*, page 361–376, Berlin, Heidelberg. Springer-Verlag.

Wes McKinney. 2010. Data Structures for Statistical Computing in Python. In *Proceedings of the 9th Python in Science Conference*, pages 56 – 61.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Tao Wu, Jingyuan Chen, Wang Lin, Mengze Li, Yumeng Zhu, Ang Li, Kun Kuang, and Fei Wu. 2025. Embracing imperfection: Simulating students with diverse cognitive levels using LLM-based agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9887–9908, Vienna, Austria. Association for Computational Linguistics.

Zhihao Yuan, Yunze Xiao, Ming Li, Weihao Xuan, Richard Tong, Mona Diab, and Tom Mitchell. 2026. Towards valid student simulation with large language models. *arXiv preprint arXiv:2601.05473*.

Matthew Zent, Digory Smith, and Simon Woodhead. 2025. PIIvot: A lightweight NLP anonymization framework for question-anchored tutoring dialogues. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 27467–27476, Suzhou, China. Association for Computational Linguistics.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang,

Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025a. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*.

Zheyuan Zhang, Daniel Zhang-Li, Jifan Yu, Linlu Gong, Jinchang Zhou, Zhanxin Hao, Jianxiao Jiang, Jie Cao, Huiqin Liu, Zhiyuan Liu, Lei Hou, and Juanzi Li. 2025b. Simulating classroom education with LLM-empowered agents. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 10364–10379, Albuquerque, New Mexico. Association for Computational Linguistics.

A Related Work

A.1 Student Simulation in Dialogues

Many recent works have explored LLM-based student simulation in educational dialogues, mainly using prompting and conditioning on student personas. MathDial introduces a dialogue dataset with LLM-simulated students exhibiting specific misconceptions (Macina et al., 2023), while other work uses zero-shot prompting to generate student responses for the purpose of downstream tutor training (Dinucu-Jianu et al., 2025). Several works simulate students by conditioning LLMs on cognitive states, misconceptions, and prior knowledge, enabling diverse simulated interactions with tutoring agents (Liu et al., 2024a; Wang et al., 2025, 2024). Other works use LLMs to simulate student learning curves and realistic misconceptions (Jin et al., 2024; Schmucker et al., 2024), behavior that corresponds to different demographic, behavioral, or personality traits (Markel et al., 2023; Liu et al., 2024b; Li et al., 2025), and goals and learning trajectories (Sharma and Li, 2024). One recent work leverages a multi-agent LLM pipeline to refine cognitive and linguistic properties of student responses, and performs DPO on resulting data (Cao et al., 2026). Another work leverages machine unlearning to remove knowledge of specific concepts from LLMs in order to study how they relearn these concepts through dialogue tutoring (Song et al., 2026). Finally, recent works have also used SFT to create simulated students, finding that this approach significantly improves authenticity (Perczel et al., 2025; Cao et al., 2026). In general, dialogue-based simulated students are typically developed for the purpose of generating data for training and evaluating LLM tutoring agents (Dinucu-Jianu et al., 2025; Scarlatos et al., 2025c; Wang et al., 2024), or creating training environments for human tutors (Pan et al., 2025; Markel et al., 2023).

However, these works tend to focus on simulating entire dialogues that are not grounded in real student data. In contrast, our work focuses on predicting student utterances at the turn-level, which we validate by comparing against real student data. While some prior works explicitly predict turn-level knowledge and correctness of student turns (Scarlatos et al., 2025a; Ikram et al., 2025), they do not predict the full text of student utterances, therefore missing many other aspects of student behavior.

A.2 Evaluating Simulated Students

Despite growing interest in student simulation, hardly any existing work thoroughly evaluate the quality of simulated student methods and how closely do they resemble real human students. A recent survey identifies a lack of validation in many works, and proposes a "Turing-like test" to capture realism (Käser and Alexandron, 2024). A recent work uses an LLM-conducted Turing test to measure simulated student realism, although it remains to be seen if LLMs can identify simulated students as well as humans, or if the results of the test are correlated with more reliable measures of simulation realism (Dou et al., 2025). A concurrent work identifies the "competence paradox", where LLMs are tasked with simulating learners that are intrinsically less knowledgeable, and argues that simulated students should be constrained by definitions of student behavior and knowledge (Yuan et al., 2026). Another recent work conducts a comprehensive analysis of simulated student realism, where expert teachers interact with simulated students, and identifies a wide range of behavioral dimensions that indicate realism and reveal weaknesses in current approaches (Martynova et al., 2025). The identified realism indicators include error patterns, consistency in knowledge acquisition, question-asking behavior, and linguistic patterns. However, all of these indicators are evaluated using human experts, leaving the question of how to *automatically* evaluate simulated students unanswered.

A common automated evaluation approach is measuring the consistency between simulated outputs and the personas they were generated on, typically via LLM-as-a-judge, used both in education (Liu et al., 2024b; Wu et al., 2025; Li et al., 2025) and other domains (Abdulhai et al., 2025; Kyung et al., 2025; Kim et al., 2025). However, these evaluations assume access to a ground-truth per-

sona and are often conducted in fully synthetic settings. Alternative approaches use human-evaluated realism scales (Macina et al., 2023; Martynova et al., 2025; Cao et al., 2026), alignment between simulated performance and conditioned cognitive states (Scarlatos et al., 2025b; Wang et al., 2025; Benedetto et al., 2024), accurately predicting correctness (Scarlatos et al., 2025a), and textual similarity between real and simulated student solutions (Ross et al., 2025; Duan et al., 2025). Finally, several works measure population-level statistics of simulated student properties, such as temporal error rates (Weitekamp et al., 2025) or length and frequency of student turns (Perczel et al., 2025; Zhang et al., 2025b), and compare against statistics measured on real student interactions.

Our evaluation metrics build on many of the ideas used in these prior works. By comparing simulated turns to ground-truth ones, we effectively measure the consistency of the simulated student. We explicitly model many of the aspects identified as important in prior works, including correctness, errors, knowledge, behavior (through dialogue acts), and similarity to real student utterances (through cosine similarity and ROUGE-L). Additionally, our work is the first to establish a framework for reference-based evaluation of simulated students in dialogues, complementing prior works that focus on fully synthetic dialogue generation and others that focus on reference-based evaluations in non-dialogue settings.

A.3 Student Simulation in Other Settings

Beyond dialogues, student simulation has been studied in problem-solving and assessment settings. A large body of work has studied student "misconceptions", particularly in math, typically fine-tuning LLMs to reliably predict errors that students will make when solving problems (Ross and Andreas, 2025; Fernandez et al., 2024; Sonkar et al., 2024a; Ross and Andreas, 2024; Daheim et al., 2024). Other works have fine-tuned or prompted LLMs to simulate student responses to exam questions, often for the purpose of evaluating question properties such as difficulty (Benedetto et al., 2024; Scarlatos et al., 2025b; Nguyen et al., 2025; Ipçi et al., 2025; He-Yueya et al., 2024b,a; Lu and Wang, 2024). Another recent area of interest is studying student behavior in programming settings, typically by fine-tuning LLMs on real or simulated student traces (Ross et al., 2025) and conditioning on student knowledge states (Duan et al., 2025). Earlier non-LLM approaches modeled student problem-solving behavior and adaptation to tutor feedback in math learning platforms (Matsuda et al., 2015), with a focus on modeling the cognitive states of students (Anderson et al., 1995).

B Implementation Details

B.1 Models and Hyperparameters

We use `meta-llama/Llama-3.1-8B-Instruct` as the base model for all fine-tuned models, except in the experiments that use a smaller student model, where we use `meta-llama/Llama-3.2-3B-Instruct`. We load all models in floating point 16 precision. For LLM prompting, we use the `gpt-4.1-2025-04-14` version of GPT-4.1 and the `gpt-5-mini-2025-08-07` version of GPT-5 mini.

We conduct preliminary hyperparameter exploration using the validation set, optimizing for our automated metrics with the student models. We use the same hyperparameters, except for epochs (shown in Table 5), across all fine-tuned models. We train using the AdamW optimizer, with learning rate = $5 \cdot 10^{-5}$ with a linear warmup for 10% of training steps, effective batch size = 64 via gradient accumulation, weight decay = $1 \cdot 10^{-2}$, and gradient norm clipping = 1.0. We set LoRA's rank $r = 32$, $\alpha = 64$, and dropout = 0.05, with rank stabilization (Kalajdzievski, 2023) and adapters on all internal weight matrices. For DPO, we set the learning rate to $5 \cdot 10^{-6}$, $\beta = 0.1$, the number of candidate student utterances per turn to $n = 4$, and the reward threshold to $\epsilon = 0.1$. For DPO, we only train on a random 20% of the dialogues in the train set for one epoch to reduce costs, and find performance to be similar to when training on the full set. In total, we form 4,998/4,703 pairs for 8B/3B DPO training, respectively. For all methods, we exclude all prompts longer than 6,000 characters at train-time to avoid memory issues.

At test-time, for all non-reasoning models, we use greedy decoding and set the maximum number of generated tokens to 400; for reasoning models, we use a temperature of 1.0 and 15,000 maximum generated tokens. For data annotation, we use 4,000 maximum generated tokens.

In Table 5, we report the validation loss, task-specific validation performance, and training/testing runtimes for each fine-tuned model used in our work. We do not include model loading in testing time, since it is a constant cost, and do not include

training data evaluation in DPO training time since it is relative to the testing time of other models. We run all experiments on NVIDIA L40S and A40 GPUs, with each experiment running on a single GPU.

B.2 Additional Metric Details

We show our dialogue act definitions in Table 6. To derive our set of acts, we started with definitions from prior work (Vail and Boyer, 2014; Hou et al., 2025; McNichols et al., 2025) and then refined them to fit our setting of one-on-one math tutoring and to retain a level of granularity that balanced low ambiguity and high behavioral descriptiveness. We then internally labeled acts and compared with LLM labels. Afterward, we further refined our definitions to address areas of disagreement and confusion. For example, we originally included an "Other" act, but found that we could achieve better labeling accuracy by explicitly labeling each utterance's act. We also originally included distinct acts for correct and incorrect math answers, but collapsed those to avoid overlapping with the correctness metric.

For the acts and correctness models, we train using a simple SFT objective where the output is the text of the corresponding label, and use greedy decoding for inference. We clarify that the correctness model is only used for the analysis in Figure 3, and not the correctness metric which uses LLM prompting. We implement our KT model using the LLMKT implementation from (Scarlatos et al., 2025a). Additionally, we condition the KT model on estimated OCEAN personas, as detailed in Section 3.2, which increases AUC significantly from 0.5940 to 0.6557. We note that this modification does not leak information to the student model at test-time since the KT model is only used for evaluation. We also acknowledge that the performance of the KT model may appear low; however, it is a difficult task since correctness is being predicted without observing the question the tutor asks. Furthermore, the goal of the KT model is to learn a correlation between utterances and future correctness, forming a complement to the correctness metric that compares correctness for the current turn.

B.3 Software

We use the Azure API for prompting GPT-4.1 for data annotation and the OpenAI API for prompting GPT-4.1 and GPT-5-mini for all other purposes[3]. We use the sentence transformers library (Reimers and Gurevych, 2019) for computing sentence embeddings. We load Llama models and perform SFT using the Huggingface Transformers library (Wolf et al., 2020), perform DPO using the trl library (von Werra et al., 2020), and perform LoRA using the peft library (Mangrulkar et al., 2022). We perform local inference using vLLM (Kwon et al., 2023). We perform all other standard machine learning operations using PyTorch (Paszke et al., 2019) and numpy (Harris et al., 2020), perform data loading and transformation using pandas (Wes McKinney, 2010), compute statistics using SciPy (Virtanen et al., 2020) and scikit-learn (Pedregosa et al., 2011), and compute ROUGE-L using rouge-score[4]. Our work complies with the terms of use for all software we use.

C Additional Dataset Details

We now provide additional details on the Question-Anchored Tutoring Dialogues 2k dataset. The dataset is completely anonymized (Zent et al., 2025), and as a result there is no demographic information available on individual students or tutors. All students are from the United Kingdom. The dataset is licensed using cc-by-nc-sa 4.0, and we release our annotations under the same license.

Dialogues are 23.42 turns long on average across student and tutor turns, with a minimum length of 10 and a maximum of 109. Dialogues can be initiated by tutors or students, with 82.63% of dialogues initiated by tutors. Student/tutor turns are 4.11/14.84 words long on average, respectively, with both containing a mixture of English, numbers, mathematical symbols, and emojis. We show the distribution of act labels in Figure 2 and the distribution of correctness labels in Figure 3.

Since solutions are not provided in the dataset, we use GPT-4.1 to annotate each question with the correct answer, a textual solution, and an explanation for each multiple-choice option. We also ask the model to identify if the question is solvable, since several questions are associated with images that were not properly translated to text or contain other data processing issues. We exclude such unsolvable questions from the dataset, resulting in 60 dialogues (3% of the total) being removed.

[3] https://platform.openai.com/docs/libraries

[4] https://github.com/google-research/google-research/tree/master/rouge

Model	Epochs	Val. Loss	Val. Perf.	Train Time	Test Time
Student Models					
Student SFT (3B)	3	1.9765	–	12	1
Student DPO (3B)	1	0.6458	–	56	1
Student SFT (8B)	3	1.8761	–	19	1
Student DPO (8B)	1	0.6426	–	81	1
Metric Support Models					
Acts (8B)	2	0.0703	0.9219	150	1
Correctness (8B)	1	0.1165	0.9164	76	1
Knowledge (8B)	3	0.6338	0.6557	128	5
Tutor (8B)	3	1.4802	–	20	2

Table 5: Training statistics for all fine-tuned models. Train and test time are reported in minutes. The validation performance (Val. Perf.) metric is accuracy for Acts and Correctness and AUC for Knowledge.

Dialogue Act	Description
Math Answer	When the tutor asks a math content-related question, the student attempts to answer that question.
Not Understanding	The student simply indicates that they do not know the answer to a question or do not understand a concept.
Seek Information	The student seeks more information regarding the math problem or topic, for example, by asking a clarifying or conceptual question.
Off-Topic	The student utterance is unrelated to the problem or math topic, including greetings, goodbyes, and other casual conversation.
Acknowledge	The student simply acknowledges what the tutor said in the previous turn.

Table 6: Dialogue acts that students can make at any turn in a dialogue.

Each dialogue is associated with a set of "subjects", which we use to form the set of knowledge components (KCs) C for our knowledge acquisition metric. We only use the most granular level of subject definitions (level 3 in the dataset). We also include an additional "Default" KC for each dialogue that the LLM can assign to turns that may not fall into the given subjects. Not including Default, there are 157 unique subjects in the dataset, with each dialogue being associated with an average of 4.57 subjects.

In rare cases, GPT-4.1 fails to annotate dialogues with acts or correctness. Act annotation failures occur 2/1/1 times in the train/validation/test splits, respectively, and correctness annotation failures occur 1/2/2 times in the train/validation/test splits, respectively. In these cases, we simply do not include turns in failed dialogues when computing metrics that rely on the ground-truth labels, at both train and test time.

D Additional Results

D.1 Additional Qualitative Analysis

We continue our discussion of qualitative findings to identify areas for improvement in fine-tuned methods. In addition to generating overly short responses, fine-tuned methods also significantly underestimate the Seek Information act, and are less likely than real students to ask questions, identified as common behavioral inconsistencies in LLMs (Martynova et al., 2025). We hypothesize that these differences may be due to using greedy decoding for generation; since a majority of real student responses are three or less words long (62%), the *most likely* response at any point in time will likely be very short. The under-representation of Seek Information further indicates that the models have not learned the nuances of when to use rare linguistic features or generate longer responses that are associated with the act. While random sampling can produce longer and more diverse responses, we found that it leads to significantly reduced performance in preliminary experiments. These findings indicate that more advanced training or decoding techniques, such as overgenerate-and-rank or MCTS, may be required to obtain longer and more complex responses while maintaining accuracy on realism metrics.

D.2 Label Distributions

We examine the distribution of both acts and correctness to better understand the behavior of different student models. Using the full test set, for each student model, we compute the portion of

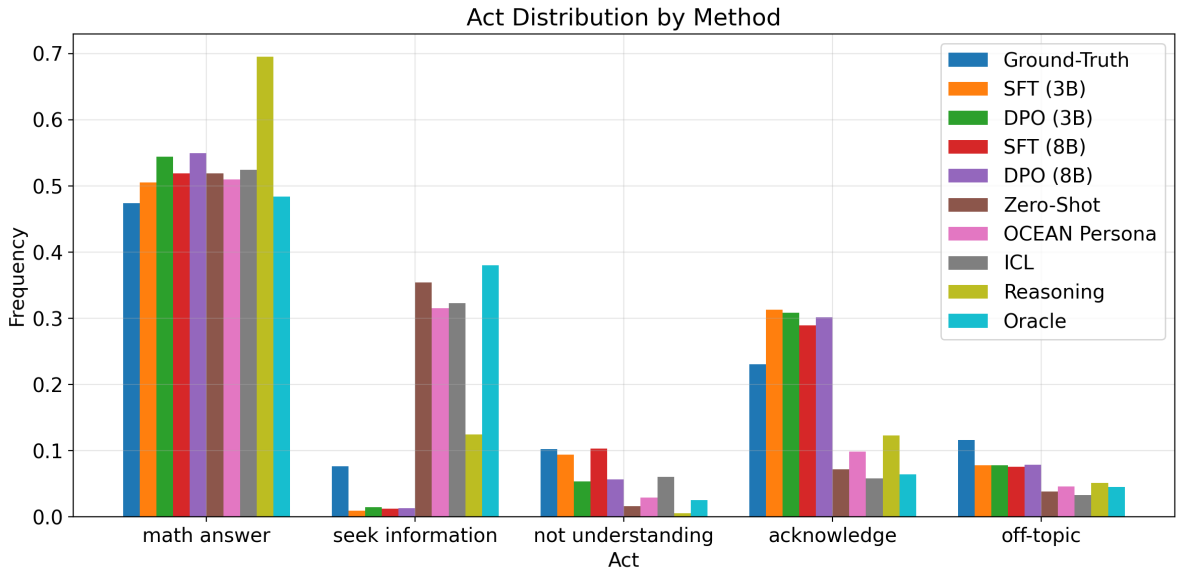

Figure 2: Distribution of act labels for real students and simulated student methods.

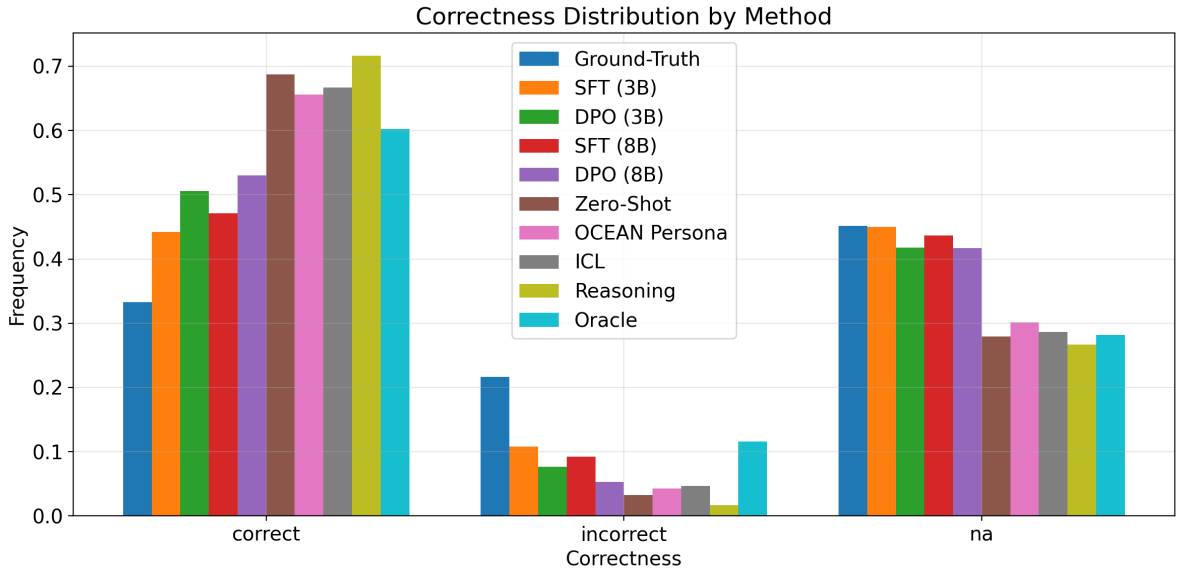

Figure 3: Distribution of correctness labels for real students and simulated student methods.

turns that are classified for each possible act and correctness label, using fine-tuned act and correctness classifiers, respectively. We compare these to the distribution of acts and correctness for the ground-truth, real student turns on the test set.

We show the distribution of acts across methods in Figure 2. We observe that while most methods have similar frequencies of Math Answer, the distributions for other acts vary greatly. Compared to prompting-based methods, the fine-tuned methods have rates that are much more similar to the ground-truth for Not Understanding, Acknowledge, and Off-Topic. However, fine-tuned methods rarely use the Seek Information act. A possible reason for this is that the fine-tuned responses tend to be very short, following common linguistic patterns in the data, while Seek Information turns tend to be longer since they involve asking specific questions. On the other hand, prompting-based methods overly use Seek Information, with lower rates for Not Understanding, Acknowledge, and Off-topic. This shows that prompting-based methods are reluctant to generate more conversational turns that are common in student dialogues, while overestimating how often students ask questions. The Reasoning model stands out because it explicitly reasons about what acts to take, but still is less accurate than the fine-tuned models.

We show the distribution of correctness across methods in Figure 3. We observe that fine-tuned methods are much closer to the ground-truth distribution than prompting methods, with prompting methods more frequently giving correct answers and less frequently giving answers that do not have a particular correctness (na). This result explains how prompting-based methods perform better on the correctness metric, because they are more likely to predict the majority class. However, both fine-tuned and prompting-based methods overestimate how often students are correct, demonstrating a need for methods to better anticipate when students will give incorrect answers.

D.3 Results by Turn

To investigate how challenging different aspects of student simulation are at different points in dialogues, we examine how well methods perform on each of our metrics at the turn level. We plot the average at each turn pair index across the test set. To exclude high variance results, we truncate all dialogues after 15 turn pairs; 83% of dialogues are within this length.

We show the results of this investigation in Figure 4. We observe that all metrics show variability across the turns of a dialogue. Many metrics, such as Acts, Cosine Similarity, ROUGE-L, and Tutor Response, are very easy for fine-tuned methods on the first turn due to most dialogues starting with a simple greeting. Prompting methods, on the other hand, perform poorly on the first turn since they have no prior turns in context. Correctness and Errors vary greatly on the first turn since most first turns have na as their correctness; in the rare cases where the first turn is correct or incorrect, some methods like Oracle correctly predict this, while others are not able to without context. After the first turn, the metrics are mostly stable, with a few exceptions: Acts are more challenging early on, since it is hard to predict the student's behavior with limited context. On the contrary, Knowledge Acquisition, becomes more difficult between turns 0 and 6, since the problem solving process begins in this range and it is difficult to estimate student knowledge with limited context; afterward, as context increases, so does Knowledge Acquisition performance. Fine-tuned methods tend to dominate prompting methods across turns, although the difference is usually greater early in the dialogues, since the prompting methods perform much better when there is more context. We also observe that the Reasoning method dominates other prompting methods later in the dialogues on Acts, Knowledge Acquisition, Cosine Similarity, and ROUGE-L, showing that it benefits from the added context more than other methods.

E Human Evaluation Details

In this section, we detail the instructions, recruitment details, and the annotation interface for our human evaluation. The study was approved by the Institutional Review Board (IRB).

Table 7 presents the job posting used to recruit participants on Upwork. We hired four annotators and excluded the data from one due to consistently low-quality submissions. All annotators were paid $60 USD for 2 hours of work, which is significantly higher than the minimum wage in the United States. Instructions were provided in the form of slides, with separate slides corresponding to each annotation table, including Table 6, Table 9, and Table 10, as well as screenshots of the annotation environment shown in Figure 5 and Figure 6. Table 8 contains the consent form used in the study.

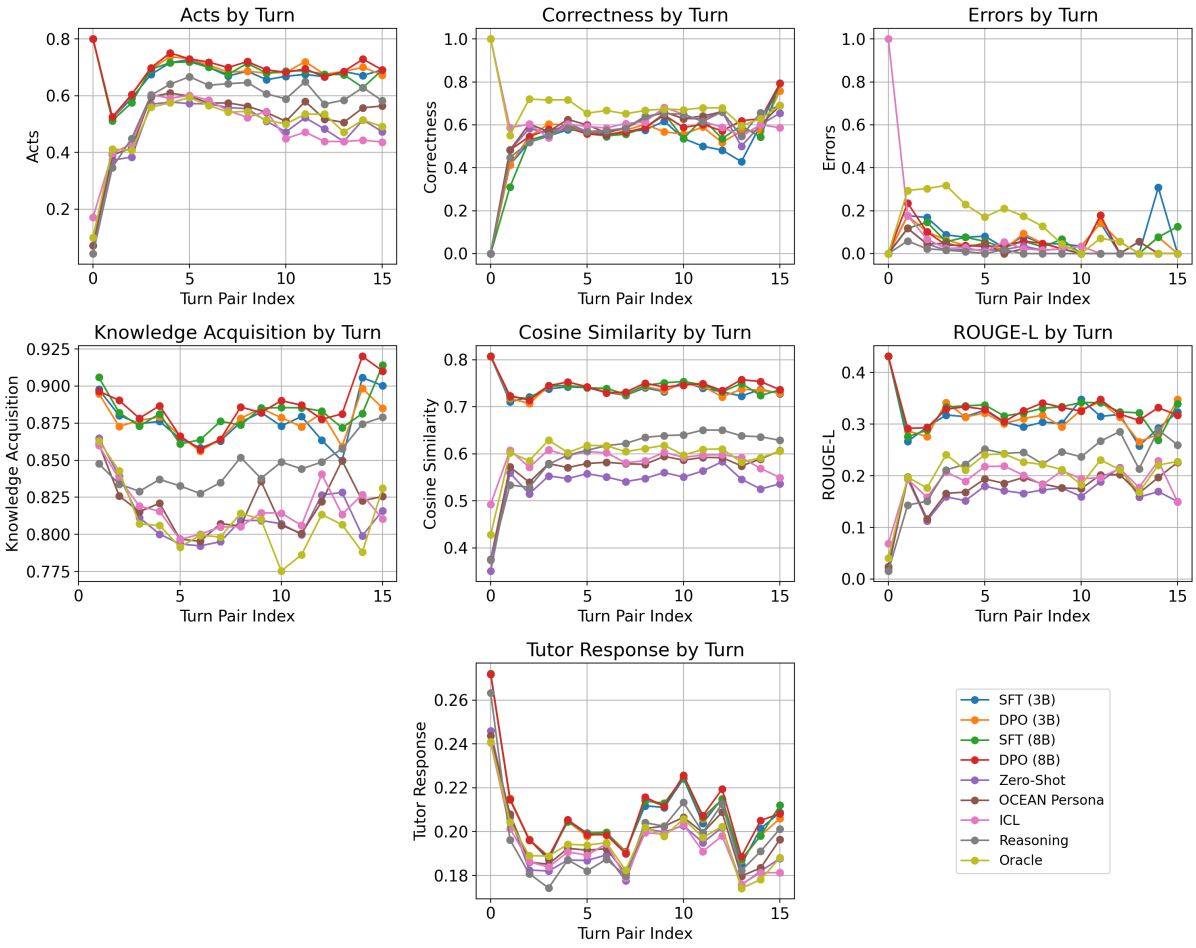

Figure 4: Results across metrics and methods broken down for each turn pair index.

We are seeking a qualified math teacher or tutor to evaluate AI-generated dialogues centered on middle school math problems. The ideal candidate will have experience in mathematics education and the ability to assess both the realism of student dialog turns. Responsibilities include reviewing dialogue content and providing ratings based on specific evaluation metrics. If you are passionate about education and interested in shaping how AI can support learning, we'd love to hear from you! For detail, please read the following post:

—

Help Shape the Future of AI Tutoring — $60 for ~2 Hours (Remote, Flexible)
Are you a math teacher or tutor in the US or UK? Your expertise can make AI tutoring actually useful for real learners. Lend your judgment, get paid, and help set the standard for quality in educational AI.

What you'll do
* Review short or full student–tutor dialogues tied to middle-school math problems
* Rate how realistic and helpful the AI's responses are
* No prep, no grading, no lesson plans—just your professional judgment

Why join
* Real impact: Your ratings directly influence how AI supports students
* Simple + flexible: Complete online on your schedule in one sitting
* Fair pay: $60 for a single ~2-hour session (paid upon full completion)

Who we're looking for
* Adults (18+) who read and understand English
* Prior math teaching or tutoring experience in the United States or United Kingdom (required)

How it works
* You'll be doing a turn-level evaluation
* You'll complete multiple problems within that single task
* Clear instructions provided; straightforward rating interface

Confidentiality
* Contact/consent info (name, email) collected only for study admin and deleted after analysis
* Data are anonymized and securely stored; any quotes used will not include identifying details

Interested? Apply here on Upwork

Table 7: Recruitment posting for math teachers and tutors for human evaluation.

Study Invitation
You are invited to participate in a research study evaluating an AI-generated student dialogue for math problems. This study is conducted by a research group from **[University Name]**.

Why are we doing this research study?
The purpose of this research study is to better understand how people evaluate AI-generated tutoring dialogues. We aim to identify which types of AI responses seem most similar to what real students would say. By studying how participants judge individual dialogue turns, we can develop clearer criteria for evaluating educational AI systems and improve their reliability and usefulness for learners.

Who can participate in this research study?
Adults aged 18 years or older who can read and understand English and who have prior mathematics teaching or tutoring experience in the United States or the United Kingdom are eligible to participate in this study.

What will I be asked to do and how much time will it take?
Participants will read several middle-school–level math problems and the accompanying dialogue materials. Participants will evaluate individual AI-generated responses within a dialogue produced by different AI systems. For each item, participants will provide ratings describing how appropriate or realistic the AI responses are. All math problems are at the middle-school level, such as ratio-chaining items (for example, "a:b = 3:2 and b:c = 3:4; what is a:c?").

- Participants will complete multiple problems within their assigned task. The total time required for the study is approximately two hours. Participants may work at their own pace but must complete their assigned task in full to receive compensation.
- Participants will first read a description of the study and then provide informed consent before beginning any study activities.
- Participants will complete several math problems during the study. For each problem, they will begin by reading and understanding the problem before proceeding to the corresponding dialogue materials.
- Participants will read a short dialogue associated with each math problem. After reading the context, they will review several AI-generated candidate responses representing possible student turns at specific moments in the dialogue. For each candidate, participants will rate how similar the response is to what a student would plausibly say at that point. This sequence will repeat for each assigned problem.

Will being in this research study help me in any way?
Being in this research study is not expected to provide any direct personal benefit to you. However, your participation may help researchers better understand how people evaluate AI-generated tutoring dialogues and may contribute to the improvement of future educational technologies.

What are my risks of being in this research study?
There is always a risk of breach of confidentiality, but this has been mitigated as described below. The risks involved in participating in this study are minimal, similar to what you might experience during a typical teaching activity. You are free to take breaks or stop at any time if you wish.

How will my personal information be protected?
Your participation will be kept confidential. We will collect your name, phone number, and email for contact and consent purposes, but this information will be deleted after the experiment analysis is complete. All personal information will be anonymized, and any quotes presented in a paper will not include identifying details. The analyzed data will be securely stored in anonymized form, with access restricted to the research team.

Will I be given any money or other compensation for being in this research study?
Yes. Participants will receive monetary compensation for completing the study. Compensation is provided at a rate of $30 per hour, for a total of $60 for completing the two-hour session. Participants must complete their assigned task in full to receive compensation.

What happens if I say yes, but I change my mind later?
Participation in this study is completely voluntary. You may withdraw at any time without any penalty or loss of benefits. If you decide to withdraw, any data collected up to that point will be used anonymously unless you request otherwise.

Who can I talk to if I have questions?
If you have questions about this project or decide to drop out, you may contact the research team at **[Researcher Email]**. If you have questions concerning your rights as a research subject, you may contact the **[University Human Research Protection Office]** at **[HRPO Phone Number]** or **[HRPO Email]**.

Consent
Please sign (or e-sign) below to indicate your consent, then scan or photograph this page and submit it as instructed by the study team.

Printed Name: _____

Date: _____

Please print or download a copy of this page for your records.

Table 8: Informed consent form for our human evaluation.

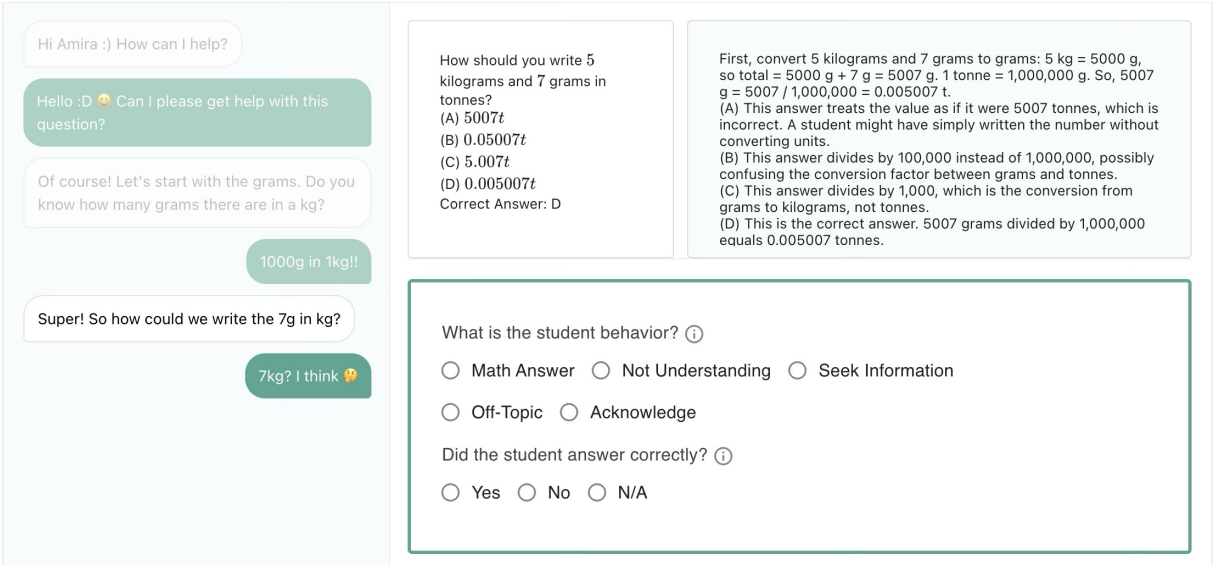

Figure 5: Human evaluation interface for evaluating ground-truth turns.

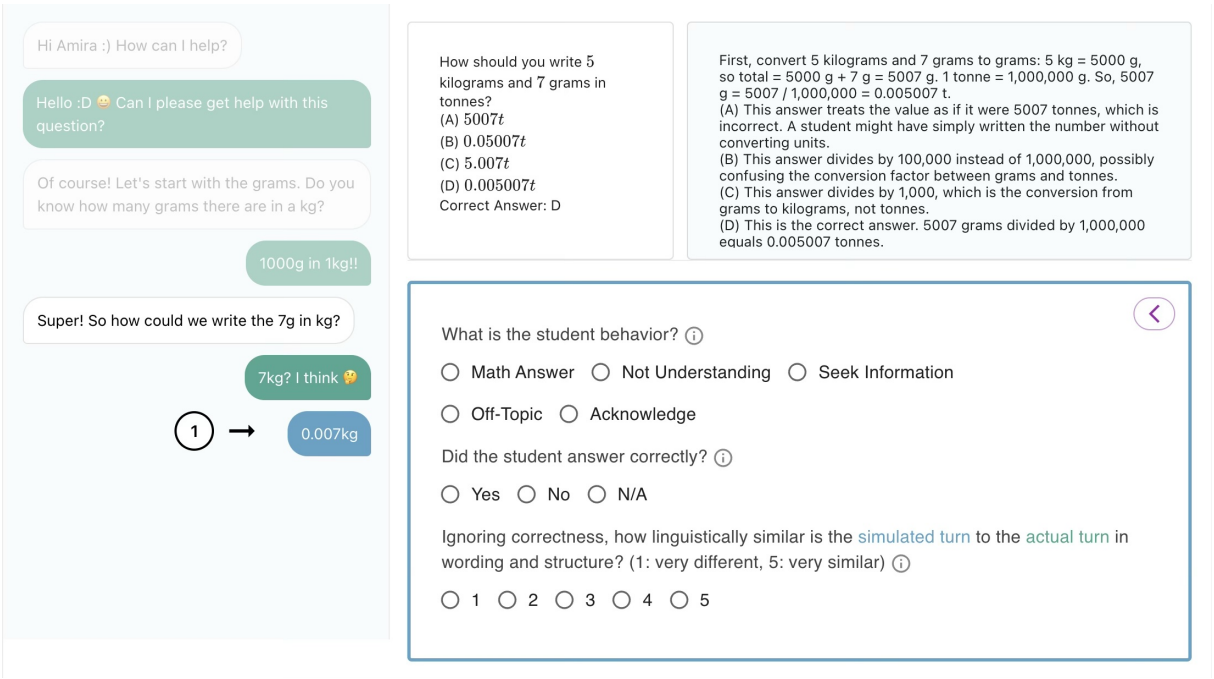

Figure 6: Human evaluation interface for evaluating simulated turns.

Correctness	Description
Yes	Student correctly responds to the previous tutor turn.
No	Student incorrectly responds to the previous tutor turn, or indicates they do not know the answer.
N/A	All other cases, such as when the tutor does not ask a question or only asks a conversational question, or if the student response is purely conversational. A turn is conversational when it does not address a mathematical task posed by the tutor.

Table 9: Correctness, *Did the student answer correctly?*

Rating	Similarity	Description
1	Not linguistically similar	Completely different wording, structure, or response type.
2	Slightly linguistically similar	Very limited overlap (e.g., both contain a number or keyword), but overall expression is different.
3	Moderately linguistically similar	Same general type of response, but noticeable differences in wording, structure, or detail.
4	Highly linguistically similar	Very similar wording and structure, with minor differences.
5	Nearly identical linguistically	Almost the same wording, structure, and level of detail.

Table 10: Linguistic Similarity, *Ignoring correctness, how linguistically similar is the simulated turn to the actual turn in wording and structure?*

F Examples

Question
Arrange the numbers in descending order. $-5, -8, 7, -6$ A: $-8, 7, -6, -5$ B: $7, -8, -6, -5$ C: $-8, -6, -5, 7$ D: $7, -5, -6, -8$ Correct Answer: D
Question Solution (Generated by GPT-4.1)
Solution: To arrange the numbers -5, -8, 7, -6 in descending order, we start with the largest and go to the smallest. 7 is the largest, then -5, then -6, then -8. So the correct order is 7, -5, -6, -8. A Explanation: This option lists -8 first, which is the smallest number, not the largest. This is the reverse of the correct order (ascending order for the negatives, but with 7 misplaced). A student might mistakenly think that more negative numbers are larger. B Explanation: This option starts with 7, which is correct, but then puts -8 before -6 and -5. This is incorrect because -8 is the smallest (most negative) number. A student might have placed 7 first but then listed the negatives in the wrong order. C Explanation: This option lists all the negative numbers first in increasing order, then 7 at the end. This is ascending order, not descending. A student might have confused ascending and descending order. D Explanation: This is the correct answer: 7, -5, -6, -8. 7 is the largest, followed by -5, then -6, then -8.
Dialogue
Tutor: Hi again! Firstly, do you know what descending order is? Student: YEARH! Tutor: Go on, what does that mean? Student: Biggest to smallest Tutor: Spot on!! From those number which number would be the biggest? Student: ✓. Wich numbers Tutor: We have -5 -8 7 -6 Student: Ohh -5 Tutor: 7 must be the biggest number as it is the only positive number. Try and imagine or even draw out a number line, 7 is the only number greater than 0. So from our answer options we can narrow it down do B or D as they both start with 7. Does that make sense so far? Student: ✓ Tutor: Excellent Student: ✓✓✓✓✓✓✓✓✓✓✓✓✓✓✓✓✓✓✓✓✓✓✓✓✓ Tutor: Now we have these number left -5 -8 -6 Student: ✓ Tutor: From these number which one is the biggest? So try and think of a number line, which number is closest to 0? Student: -5 Tutor: Well done Mariana, that points us to answer D! 7, -5, -6, -8 Student: D✓ Tutor: Bye for now! Student: Bye thanks for your help!
OCEAN Persona (Generated by GPT-4.1)
Reasoning: The student demonstrates understanding of mathematical concepts when prompted and is able to follow the tutor's guidance. Their responses are brief and sometimes enthusiastic (e.g., 'YEARH!', multiple checkmarks), but do not show much elaboration or curiosity beyond the immediate task, suggesting neutral openness. The student is responsive and follows instructions, indicating a reasonable level of conscientiousness, though there is no evidence of independent organization or planning, so this is also neutral. The student is not very talkative and gives short answers, which suggests low extraversion. The student is polite, cooperative, and expresses gratitude at the end, indicating high agreeableness. There is no evidence of anxiety, frustration, or negative emotions, so neuroticism appears low. Openness: neutral Conscientiousness: neutral Extraversion: low Agreeableness: high Neuroticism: low
Oracle Summary (Generated by GPT-4.1)

The student demonstrates a basic understanding of mathematical concepts when prompted, correctly identifying "descending order" as "biggest to smallest" and, with guidance, selecting the correct answer. Initially, the student makes a conceptual error by choosing -5 as the largest number, indicating some confusion about the relative size of negative and positive numbers. However, the student is receptive to feedback, quickly adjusts their thinking when the tutor explains, and successfully applies the number line strategy when prompted. Behaviorally, the student is highly responsive, frequently uses affirmative emojis (✓) and enthusiastic language, and often provides brief, sometimes minimal, answers, suggesting a preference for quick confirmation over detailed explanation. The student is agreeable and polite, expressing gratitude at the end, and shows openness to learning by engaging with the tutor's questions. Linguistically, the student uses informal, enthusiastic responses, sometimes with spelling errors ("YEARH!," "Wich numbers," "Ohh -5"), and tends to favor emoji-heavy, concise replies over elaboration. Overall, the student is eager, agreeable, and responsive, with a tendency toward impulsive answers but a willingness to learn and correct mistakes when guided.

Retrieved ICL Example
Student: Hello! Tutor: Hiya! How can I help? Student: May you help me please? Am very stuck! Tutor: of course! Student: Thank you Tutor: are you stuck on writing numbers in desending order? Student: yes Tutor: do you know what descending means? Student: Is it largest to smallest? I don't really know? Tutor: that's correct! Student: Oh... Tutor: so which one is the largest? Student: 7 Tutor: good, now which is the next biggest? Student: -5 Tutor: good! now what? Student: -6 Tutor: good! and finally? Student: -8 Tutor: good! Student: oh that was easy! hehe Tutor: would you like any more help? Student: i want to say a big thank you for helping me! Tutor: hahah no problem! You did very well! Have a nice evening!

Table 13: Example dialogue, question, solution, and associated context provided to OCEAN, Oracle, and ICL prompting methods. Emojis except checkmark are removed.

Question
Alex has two fraction cards. $\frac{5}{8}$ ☐ The product of the cards is $\frac{1}{6}$ What is the other card? A: $\frac{19}{24}$ B: $\frac{5}{48}$ C: $\frac{15}{4}$ D: $\frac{4}{15}$ Correct Answer: D

Dialogue Context
Tutor: Hello again Leo! How can I help? Student: I'm don't get the fact about missing number questions. Tutor: Well, we know that 5/8 x ? = 1/6. So we need to rearrange that to get ? = Student: Would we do the inverse. Tutor: What would that be?
Student: 5/8 divided by 1/6?
SFT 3B: 8/5
DPO 3B: 8/5
SFT 8B: 5/8
DPO 8B: 5/8 divided by 1/6
Zero-Shot: So, would I divide 1/6 by 5/8?
OCEAN: Um, so would I divide 1/6 by 5/8? I think that's the inverse, right?
ICL: So like, 1/6 divided by 5/8?
Reasoning: Yes — multiply by the reciprocal. x = (1/6) ÷ (5/8) = (1/6)*(8/5) = 8/30 = 4/15
Oracle: So, like, the opposite of multiplying? So we would divide 1/6 by 5/8? Is that right?
Tutor: Nearly! We need to divide the left by 5/8 to get ? on its own, so we must do the same to the right hand side

	Next Student Turn Scores						
Source	Acts (Label)	Corr. (Label)	Errors	Knowledge (Quant.)	Cos. Sim.	ROUGE-L	Tutor Resp.
Ground-Truth	– (MA)	– (incorrect)	–	– (4.00)	–	–	–
SFT 3B	1 (MA)	0 (correct)	0	0.9167 (3.67)	0.5388	0.2500	0.1893
DPO 3B	1 (MA)	0 (correct)	0	0.9167 (3.67)	0.5388	0.2500	0.1893
SFT 8B	1 (MA)	1 (incorrect)	0	0.9167 (3.67)	0.6600	0.5000	0.1818
DPO 8B	1 (MA)	1 (incorrect)	1	0.9167 (3.67)	0.9567	1.0000	0.1818
Zero-Shot	0 (SI)	0 (correct)	0	0.7500 (3.00)	0.8714	0.2667	0.1818
OCEAN	0 (SI)	0 (correct)	0	0.9167 (3.67)	0.7874	0.1739	0.1798
ICL	1 (MA)	0 (correct)	0	0.8333 (3.33)	0.9039	0.2857	0.1720
Reasoning	1 (MA)	0 (correct)	0	1.0000 (4.00)	0.7960	0.3333	0.1531
Oracle	0 (SI)	0 (correct)	0	0.7500 (3.00)	0.7498	0.1667	0.1778

Table 11: Qualitative example for a single turn, showing the ground-truth student response (bordered in green), the simulated responses for that turn from all methods (bordered in yellow), and corresponding scores across all metrics. The ground-truth student response is an incorrect math answer (MA) to the previous tutor turn, where the correct answer would divide in the other direction. Most simulated responses are classified as MA, while some are classified as Seek Information (SI) due to having a more questioning tone. All simulated responses except for SFT 8B and DPO 8B are correct, with SFT 3B and DPO 3B inverting 5/8, most prompting methods inverting the full equation, and Reasoning giving a complete solution to the problem. While SFT 8B and DPO 8B are both incorrect, only DPO 8B has the same error as the ground-truth student response. For knowledge, we show the score and the average delta quantile across KCs. In this example, knowledge is highly correlated with task difficulty, reflecting findings in (Scarlatos et al., 2025a). Specifically, incorrect or partially correct answers are likely to lead to easier followup tutor questions due to scaffolding, and are therefore more likely to be answered correctly by the student. On the other hand, correct answers are likely to be followed by the tutor posing a new task, which is likely to be harder than a scaffolded question, leading to a lower likelihood of the student answering correctly. The knowledge increases for Reasoning because the question is solved, so no new tasks will be posed. Cos. Sim. and ROUGE-L are highly correlated with textual overlap. Finally, Tutor Resp. is lower for prompting methods since they give full solutions, which would be unlikely to lead to the ground-truth tutor response.

Question			
$\frac{A}{10} = \frac{9}{15}$ What is the value of A ? A: 3 B: 4 C: 6 D: 9 Correct Answer: C			
Knowledge Components (KCs)			
Fractions of an Amount Mental Addition and Subtraction Equivalent Fractions Dividing Fractions Naming Co-ordinates in 2D Default			
Dialogue Context			
Tutor: Have you tried simplifying 9/15? Student: yes. 3/5 Tutor: Excellent! Good work!!!! Now how would you get that to a denominator of 10? Student: what do you mean. 3/5 Tutor: so in 3/5 you have 5 as the denominator Student: oh Tutor: this question needs 10 Student: oh you times the numerator and denominator by 2 Tutor: Exactly!!!!! So what do you think it will become?			
		Next Student Turn	
Source	Utterance	∇Z	$\text{quant}(\nabla Z)$
Ground-Truth	c	[0.0117, 0.0117, -0.0156, 0.0039, 0.0078, -0.0078]	[2, 2, 1, 2, 2, 1]
Simulated	6/10	[0.0391, 0.0352, 0.0391, 0.0352, 0.0156, 0.0352]	[3, 3, 3, 3, 3, 3]
Delta Quantile Upper Bounds: [-0.0234, 0.0000, 0.0156, 0.0430, ∞]			
Knowledge Acquisition Similarity			
$1 - \frac{\lvert 2-3\rvert+\lvert 2-3\rvert+\lvert 1-3\rvert+\lvert 2-3\rvert+\lvert 2-3\rvert+\lvert 1-3\rvert}{4\times 6} = 0.6666$			

Table 12: Example decomposition of Knowledge Acquisition similarity for a single turn. The simulated turn is from DPO 8B. Early turns in the dialogue with greetings are excluded for brevity. While both answers are correct, the ground-truth simply gives the final answer option, while the simulated turn gives the full fraction associated with the correct answer. The knowledge tracing model estimates that the student's knowledge delta for the ground-truth turn is roughly average for all KCs (2 or 1 between 0 and 4), while the knowledge delta for the simulated turn is above average for all KCs (all 3). This difference may be due to the possibility that answering "c" is simply a lucky guess, whereas answering "6/10" conveys an understanding of the solution, leading to a higher knowledge estimate. This example demonstrates the need for a knowledge-based metric, since two turns can have the same correctness but give different indications of student mastery on concepts.

G Prompts

G.1 Annotations

You are a math education expert. Your job is to label the **dialogue acts** for student turns in a given dialogue.

These are the available dialogue act labels:
- Math Answer: When the tutor asks a math content-related question, the student attempts to answer to that question
- Seek Information: The student seeks more information regarding the math problem or topic, for example, by asking a clarifying or conceptual question
- Not Understanding: The student simply indicates that they do not know the answer to a question or do not understand a concept
- Acknowledge: The student simply acknowledges what the tutor said in the previous turn
- Off-Topic: The student utterance is unrelated to the problem or math topic, including greetings, goodbyes, and other casual conversation

For each **student turn** in the dialogue, choose the dialogue act that best describes the turn. Pick exactly one act for each turn from the list above, and write the dialogue act name exactly as it appears. Before writing the acts for a turn, provide reasoning about what the best act should be.

Please provide your answer as a JSON object with the following format:

```
{
   "turn n": {
      "reasoning": "...",
      "act": "..."
   },
   "turn n+2": {
      "reasoning": "...",
      "act": "..."
   },
   ...
}
```

Table 14: System prompt for annotating dialogue acts.

You are an experienced math teacher and education expert. You are given a dialogue between a student and tutor where the student is trying to solve a math problem. Your job is to identify when the student responds correctly to the tutor. Please follow these instructions carefully when making your prediction:
- For each student turn, identify the correctness of the student's response to the previous tutor turn.
- Correctness can be true, false, or null. It is true when the student correctly responds to the previous tutor turn. It is false if the student incorrectly responds to the previous tutor turn, or indicates they do not know the answer. It is null in all other cases, such as when the tutor does not ask a question or only asks a conversational question, or if the student response is purely conversational. A turn is conversational when it does not address a mathematical task posed by the tutor.
- Before making each correctness prediction, write a short summary of each student turn in the dialogue. The summary should include the task previously posed by the tutor, and explain why the student's response is correct, incorrect, or conversational.
- Your final prediction should be a JSON object using the template: {"turn n": {"summary": ..., "correct": true/-false/null}, "turn n+2": ...}.
- Use the turn index from the conversation history as the key in your result. There should be exactly one entry for each student turn in the dialogue.

Table 15: System prompt for annotating correctness.

You are an experienced math teacher and education expert. You are given a dialogue between a student and tutor where the student is trying to solve a math problem. Your job is to list the knowledge components (KCs) that can be used to classify the learning objectives at each turn in this dialogue. Please follow these instructions carefully when making your prediction:
- Tutor turns are often phrased as questions or tasks. In these cases, choose KCs that the student will need in order to respond correctly to the tutor's question. If the tutor turn does not pose a question or task, then you do not need to assign KCs to it.
- You will be given a list of KCs to choose from. When choosing them, write them exactly as they appear.
- If the tutor posed a task but none of the given KCs apply, assign "Default".
- Write a short summary of each tutor turn in the dialogue, including the intended learning objectives.
- Along with each summary, list ALL candidate KCs that can be used to describe each tutor turn in the dialogue.
- Your final response should be a JSON object using the template: {"turn n": {"summary": "...", "kcs": ["kc 1 id", "kc 2 id", ...]}, "turn n+2": ...}
- Use the turn index from the conversation history as the key in your result. There should be exactly one entry for each tutor turn in the dialogue.

Table 16: System prompt for annotating knowledge components.

You are a math education expert. Your task is to analyze the options of math multiple choice questions. Follow these instructions carefully:
- First attempt to solve the problem. If it is not possible to solve the problem because it is poorly defined, then say the problem is not solvable.
- Then write an explanation for each option. If the option is the correct answer, write the correct solution to reach that answer. If the option is an incorrect answer, explain the error a student might make to reach that answer.
- Give your final response as a JSON object with the following template:

```
{
  "solution": ...,
  "solvable": true/false,
  "correct_option": 1-4,
  "option_1_explanation": ...,
  "option_2_explanation": ...,
  "option_3_explanation": ...,
  "option_4_explanation": ...
}
```

Table 17: System prompt for annotating question solutions.

You are analyzing a dialogue between a student and a math tutor. Your task is to assess the student's personality based on the OCEAN model, also known as the Big Five Traits.
OCEAN Traits Description:
- **Openness to Experience:** Reflects the student's curiosity, creativity, willingness to try new things, and openness to new ideas and experiences.
- **Conscientiousness:** Indicates the student's level of organization, diligence, responsibility, and reliability in approaching tasks.
- **Extraversion:** Represents how outgoing, energetic, and socially confident the student appears.
- **Agreeableness:** Measures the student's friendliness, cooperativeness, compassion, and willingness to collaborate.
- **Neuroticism:** Assesses the student's emotional stability, tendency to experience negative emotions such as anxiety, moodiness, or vulnerability to stress.
First provide reasoning about the student's behavior with respect to the OCEAN model. Then, determine if the student's expression of each trait is **high**, **neutral**, or **low**. Base your reasoning only on the dialogue provided. In your final answer, output your results as a JSON object with the following template:

```
{
 "reasoning": "...",
 "Openness": "low/neutral/high",
 "Conscientiousness": "low/neutral/high",
 "Extraversion": "low/neutral/high",
 "Agreeableness": "low/neutral/high",
 "Neuroticism": "low/neutral/high"
}
```

Table 18: System prompt for annotating OCEAN personas.

You are analyzing a dialogue between a student and a math tutor. Your task is to summarize the student's persona based on their interactions in the dialogue. Focus on the following aspects:
- How well the student acquires knowledge during the dialogue.
- The types of mathematical errors the student makes.
- Any notable behavioral patterns, such as frequent question asking, immediately jumping to the answer, distracting from the task at hand, etc.
- The student's personality traits, such as openness, conscientiousness, extraversion, agreeableness, and neuroticism.
- Notable linguistic patterns in the student's responses.

Your response should be a single paragraph summarizing the student's persona.

Table 19: System prompt for creating Oracle summaries.

G.2 Evaluation Models

Your task is to classify the dialogue acts for the last turn in the given dialogue.

Table 20: System prompt for fine-tuned act classifier model.

Your task is to classify whether the last student turn in the given dialogue is one of: "correct", "incorrect", or "na".

Table 21: System prompt for fine-tuned correctness classifier model.

You are a tutor guiding a student through a math problem.

Table 22: System prompt for fine-tuned tutor model.

You are a math education expert. You will observe a tutoring dialogue where a student is attempting to solve a math problem. You will see two versions of the next student turn: a ground-truth turn and a candidate turn. Your job is to evaluate the correctness and errors of the candidate turn.
- The correctness of the ground-truth turn is given. You must evaluate the correctness of the candidate turn.
- Correctness can be "correct", "incorrect", or "na". It is "correct" if the student correctly responds to the previous tutor turn. It is "incorrect" if the student incorrectly responds to the previous tutor turn, or indicates they do not know the answer. It is "na" in all other cases, such as when the tutor does not ask a question or only asks a conversational question, or if the student response is purely conversational. A turn is conversational when it does not address a mathematical task posed by the tutor.
- If both the ground-truth AND candidate turns are "incorrect", evaluate if they have the same error. They have the same error if the two turns are mathematically EQUIVALENT. If they are mathematically inequivalent, they do NOT have the same error.

After reasoning, please return the correctness of the **candidate** turn as "correct", "incorrect", or "na". If both the ground-truth and candidate turns are incorrect, add "same error" or "different error" to your response (ex: "incorrect, same error"). Do not include any other text in your response.

Table 23: System prompt for correctness and error prediction.

You are an experienced math teacher. You are given a dialogue between a student and teacher where a student is trying to solve a math problem. Your job is to predict if the student has a particular knowledge component (KC) at the current point in the dialogue. Please follow these instructions carefully when making your prediction:
- The student will need to possess this KC in order to respond correctly to the teacher's most recent question.
- Use previous information in the dialogue to determine if the student has this KC or not.
- Only respond with a single word, "True" or "False".

Table 24: System prompt for knowledge tracing model.

G.3 Student Models

You are a student attempting to solve a math problem, seeking help from a tutor.

Table 25: System prompt for fine-tuned student model.

You will act as a student in a conversation with a teacher in training. You will need to act as much like a student as possible. If possible do not respond with overly long messages. The conversation with the teacher will be about the following math problem. You may or may not know how to solve it already, let the teacher guide you to the correct understanding. You will be tested at the end and scored thus it is best if you collaborate with the teacher as it has more experience in math than you. If you believe you have figured out the problem and don't need any more help, put <end_of_dialogue> after your response.

Table 26: System prompt for Zero-Shot student model.

You will act as a student in a conversation with a teacher in training. You will need to act as much like a student as possible. If possible do not respond with overly long messages. The conversation with the teacher will be about the following math problem. You may or may not know how to solve it already, let the teacher guide you to the correct understanding. You will be tested at the end and scored thus it is best if you collaborate with the teacher as it has more experience in math than you. If you believe you have figured out the problem and don't need any more help, put <end_of_dialogue> after your response.

You will be given a Big Five persona that describes how you should act in the dialogue. Follow this persona as closely as possible.

Table 27: System prompt for OCEAN student model.

You will act as a student in a conversation with a teacher in training. You will need to act as much like a student as possible. If possible do not respond with overly long messages. The conversation with the teacher will be about the following math problem. You may or may not know how to solve it already, let the teacher guide you to the correct understanding. You will be tested at the end and scored thus it is best if you collaborate with the teacher as it has more experience in math than you. If you believe you have figured out the problem and don't need any more help, put <end_of_dialogue> after your response.

You will be given a persona that describes how you should act in the dialogue. Follow this persona as closely as possible.

Table 28: System prompt for Oracle student model.

You will act as a student in a conversation with a teacher in training. You will need to act as much like a student as possible. If possible do not respond with overly long messages. The conversation with the teacher will be about the following math problem. You may or may not know how to solve it already, let the teacher guide you to the correct understanding. You will be tested at the end and scored thus it is best if you collaborate with the teacher as it has more experience in math than you. If you believe you have figured out the problem and don't need any more help, put <end_of_dialogue> after your response.

You will also be given an example of a previous dialogue. Your responses should be similar to the ones in this example.

Table 29: System prompt for ICL student model.

You will act as a student in a conversation with a teacher in training. You will need to act as much like a student as possible. If possible do not respond with overly long messages. The conversation with the teacher will be about the following math problem. You may or may not know how to solve it already, let the teacher guide you to the correct understanding. You will be tested at the end and scored thus it is best if you collaborate with the teacher as it has more experience in math than you. If you believe you have figured out the problem and don't need any more help, put <end_of_dialogue> after your response.

Your response will be judged on how well it matches what the actual student said next in the dialogue (unseen). The following criteria will be used to evaluate your response:
- Acts: Does your response make the same dialogue act as the real student response
- Correctness: Does your response have the same correctness as the real student response
- Errors: If your response is an incorrect math answer, does it have the same underlying error as the real student response
- Knowledge: Does your response represent the same mastery of knowledge concepts as the real student response
- Linguistic: Does your response have the same linguistic features as the real student response

These are the available dialogue acts:
- Math Answer: When the tutor asks a math content-related question, the student attempts to answer to that question
- Seek Information: The student seeks more information regarding the math problem or topic, for example, by asking a clarifying or conceptual question
- Not Understanding: The student simply indicates that they do not know the answer to a question or do not understand a concept
- Acknowledge: The student simply acknowledges what the tutor said in the previous turn
- Off-Topic: The student utterance is unrelated to the problem or math topic, including greetings, goodbyes, and other casual conversation

These are the available correctness states:
- Correct: The student correctly responds to the mathematical task posed in the previous tutor turn
- Incorrect: The student incorrectly responds to the mathematical task posed in the previous tutor turn or indicates they don't know the answer
- NA: The tutor doesn't pose a task that has a clear correct/incorrect answer OR the student doesn't indicate correctness in their response

Reason about how to respond in order to maximize the evaluation criteria. Your final response should only contain the predicted student utterance.

Table 30: System prompt for Reasoning student model.