Simulating Student Interactions for Virtual Pretesting with In-Context Learning

Arthur Thuy[1,2], Luca Benedetto[3,4], Ekaterina Loginova[5], Dries F. Benoit[1,2]
[1]Ghent University, [2]CVAMO Core Lab, Flanders Make, [3]ALTA Institute, University of Cambridge,
[4]Télécom SudParis, Institut Polytechnique de Paris, [5]Dedalus Healthcare
{arthur.thuy, dries.benoit}@ugent.be, lb990@cam.ac.uk, ekaterina.d.loginova@gmail.com

Abstract

Recent research has experimented with using Large Language Models (LLMs) for simulating student responses to exam questions. This approach, known as *virtual pretesting*, potentially offers a scalable alternative to traditional pretesting, which is costly and time-intensive, by enabling the creation of datasets of *virtual students' responses*. Prior studies focused on zero-shot role-playing, prompting one LLM to imitate students of different levels, but showed limited alignment with response patterns of real students. This work introduces a framework that improves the alignment of LLM-based student simulations through in-context learning (ICL), leveraging previous question-answer records to provide the model with richer information about students' skills and misconceptions. Our experiments show that not all models can leverage the additional contextual information. However, a multi-model approach, which combines simulations from several models, significantly improves alignment of the simulated responses when provided with relevant context: we observe a reduction of up to 30% in difficulty estimation RMSE with respect to the non-contextual and individual contextual models. Overall, our findings indicate that LLMs can be used with ICL to create synthetic datasets of student responses approximating some patterns of learner behavior, however their ability to align with authentic student performance remains limited.

Keywords: virtual pretesting, dataset generation, student simulation, in-context learning

1. Introduction

Keeping students engaged with questions that are challenging, but not frustratingly difficult, is essential for sustaining motivation and fostering effective learning (Vygotsky and Cole, 1978). Personalized learning systems (Van der Linden and Glas, 2000) address this challenge by tailoring questions to a student's skill level. To do this, they leverage Question Difficulty Estimation (QDE) to ensure that questions of the "right" difficulty are selected. Traditionally, QDE has relied on pretesting (Lane et al., 2016): new questions are embedded within exams without contributing to students' scores and the responses to these questions, combined with data from other test items, are then used to perform QDE with statistical models, such as Item Response Theory (IRT) (Hambleton et al., 1991). Although pretesting ensures reliable QDE, it is time-intensive, costly, and exposes assessment content.

To mitigate these limitations, previous research has explored QDE from text by training machine learning models on large collections of calibrated questions to predict their difficulty from their textual content (Benedetto, 2023; Benedetto et al., 2023; AlKhuzaey et al., 2021). The current state of the art involves fine-tuning pre-trained encoder-only Transformer models, such as BERT (Devlin et al., 2018). Although this approach leverages transfer learning to generalize difficulty estimation to new items, it still depends on substantial amounts of calibrated (i.e., pretested) questions for training, which are expensive and difficult to acquire.

Advances in NLP have opened up alternative avenues for QDE by enabling *virtual pretesting*: in this paradigm, LLMs are used to answer exam questions, producing question–answer records that substitute traditional pretesting data (i.e., real students' responses). As such, virtual pretesting can be seen as a technique to generate (or augment) datasets for IRT calibration, and their reliability depends on the alignment between the response patterns of LLMs and those of real students. Two main strategies have been investigated for virtual pretesting. (i) Multi-LLM simulation: exploiting the natural variation in accuracy across different LLMs, ranging from low to high-performing models (Park et al., 2024). (ii) Role-playing with a single LLM: prompting one LLM to generate responses as if it were a student of a specific skill level, ranging from low to high proficiency (Benedetto et al., 2024). In essence, multi-LLM simulation uses the different "skill levels" of various LLMs, while role-playing steers one LLM into answering at different skill levels.

Prior work merely adopted zero-shot prompting, thereby avoiding reliance on real question–answer records that may be unavailable in practice. While the multi-LLM approach is competitive to supervised models on some datasets, its dependence on tens of LLMs makes it computationally expensive and unsuitable for most applications. Conversely, zero-shot role-playing with a single LLM is more efficient but exhibits very limited alignment with real students, reducing its reliability for practical use.

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 10376–10389
11–16 May 2026.

To address these challenges, we propose a role-playing framework that leverages *in-context learning* (ICL) to enhance the alignment of data generated for virtual pretesting. Rather than relying solely on zero-shot prompting, our approach uses authentic question–answer records to provide the LLM with rich information about relevant student skills and misconceptions, thus enabling a more accurate replication of realistic response patterns. Compared to supervised fine-tuning, the amount of authentic question–answer records required is much smaller, making QDE tools more accessible. The repository with code and prompts is available on GitHub[1].

2. Related Work

Natural language processing (NLP) has been widely applied to QDE to reduce reliance on manual calibration (Attali et al., 2014) and pretesting (Lane et al., 2016), both of which are costly and time-consuming. The predominant approach is to train supervised models that predict item difficulty directly from question text (AlKhuzaey et al., 2021; Benedetto et al., 2023). Earlier methods rely on traditional machine learning algorithms and handcrafted features, including linguistic indicators (Beinborn et al., 2015), word embeddings (Hsu et al., 2018; Yaneva et al., 2019, 2020), and TF–IDF representations (Benedetto et al., 2020). More recent work has established fine-tuning of pre-trained encoder-only Transformer models as the state of the art (Benedetto, 2023; Thuy et al., 2025; Feng et al., 2025). Once trained, these models allow for rapid calibration of new items, substantially reducing the need for manual calibration and pretesting.

Despite these advances, supervised fine-tuning requires large labeled datasets containing thousands of calibrated items, which are rarely available in practice. Several approaches attempt to alleviate this data bottleneck. For example, Loginova et al. (2021) propose an unsupervised method based on additional pre-training combined with pairwise difficulty estimation, while Thuy et al. (2024) employ active learning to approach supervised-level performance with substantially fewer labeled examples.

More recently, researchers have begun to explore virtual pretesting, in which *simulated* student responses are used to estimate item difficulty. This has been pursued either by prompting generative LLMs (Benedetto et al., 2024; Park et al., 2024) or by fine-tuning encoder-based LLMs on existing student response logs (Maeda, 2025; Uto et al., 2025). Within the generative setting, two strategies dominate: Multi-LLM simulation, which leverages the diverse performance levels of different LLMs (Park et al., 2024), and Role-playing with a single LLM, where one model is prompted to imitate students across a range of proficiency levels (Benedetto et al., 2024; Liu et al., 2025; Säuberli et al., 2025).

Studies focusing on role-playing consistently adopt zero-shot prompting and generally express skepticism regarding its feasibility for high-stakes applications. They emphasize that LLMs, in the absence of contextual information, fail to capture the cognitive mechanisms underlying student responses, and therefore should not be relied upon for piloting educational assessments. Collectively, this line of work highlights the need to move beyond zero-shot prompting and to enrich LLM simulations with student-specific contextual information.

3. Methodology

Our framework leverages ICL to improve the alignment between responses simulated with LLMs and real students' responses. It consists of two steps: i) previous responses of students are used to determine their skills and misconceptions (§3.1); ii) the responses to new questions of a prototypical student with a certain proficiency level are simulated with role-playing (§3.2), providing the skills and misconceptions as context. Notably, although the second step generates responses to MCQs, the first step can be performed on either MCQs or open-ended responses, depending on the data available. Moreover, if the experimental dataset which has to be augmented for virtual pretesting provides a log of individual student responses, our framework can evaluate student behavior replication performance (details in §3.3). Both student role-playing and replication leverage ICL, supported by a variety of example selection strategies (§3.4), to condition the model's predictions with information on skills and misconceptions.

Figure 1 provides an overview of the framework, which is model-agnostic and does not require access to LLMs' internal parameters. First, we compile a database of skills and misconceptions associated with each student-question interaction (Step 1). Then, this information is provided to the LLM to improve the alignment of its simulation (Step 2). Importantly, questions are split into train/validation/test, with corresponding splits applied to student interactions. Thus, the same student ID may appear in both training and evaluation sets, allowing us to assess how well the LLMs can simulate individual students in a setting analogous to knowledge tracing (KT) (Shen et al., 2024).

3.1. Collecting skills and misconceptions

The first step involves determining the skills and misconceptions corresponding to each student-

[1] https://github.com/arthur-thuy/LLM-virtual-pretesting

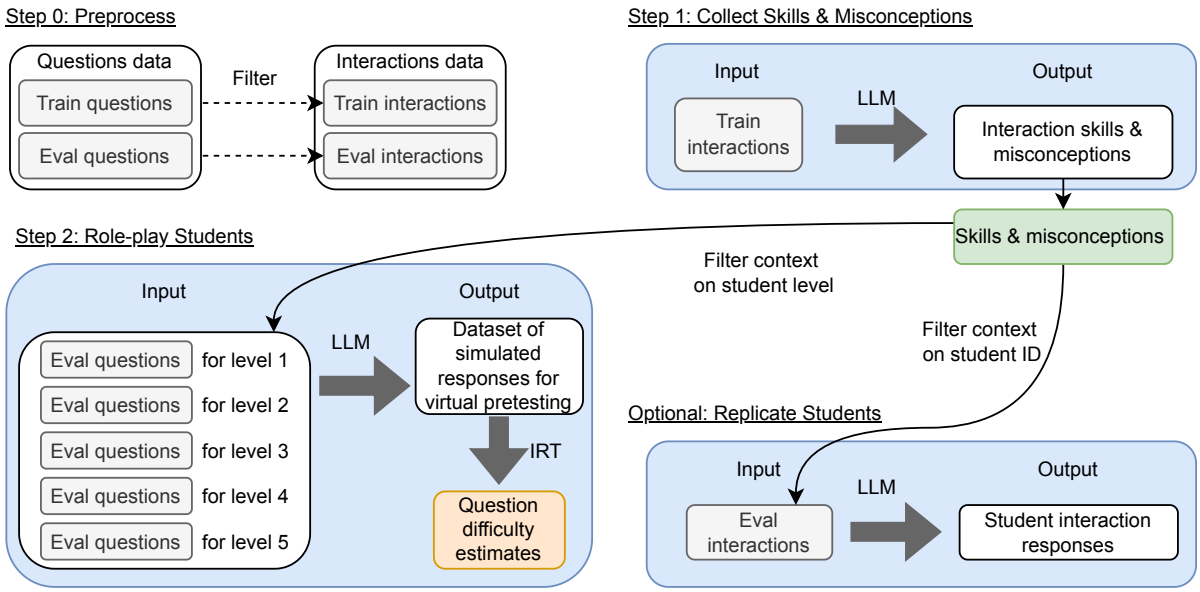


Figure 1: Overview of the methodology.

question interaction; in other words, this consists in observing each interaction and answering the question "which are the skills and/or misconceptions exhibited by the student in this response?". The way this is done, in practice, depends on whether the tasks are MCQs or open-ended.

For MCQs, we prompt a reasoning LLM to identify the skills required to select the correct answer and the possible misconceptions that could lead a student to choose each distractor. We then apply this to all the responses in the dataset, to obtain a dictionary mapping each student response to either a skill (for correct answers) or a misconception (for wrong answers). For open-ended tasks, the LLM is prompted to create a list of (at most) three skills and three misconceptions for each student answer. The result is a dictionary mapping each interaction ID to a list of skills and misconceptions.

We use GPT-o4-mini ("o4-mini-2025-04-16") for this first step, with a default temperature of 1.0, but the framework is model agnostic. The prompt template is provided in Appendix A.1.

3.2. Student Behavior Role-playing

In the role-playing stage, the LLM is prompted to predict how a prototypical student will respond to a new question, provided a certain student level, previous interactions, and a selection of skills and misconceptions as an indication of their current understanding. Notably, when building the conversation, the list of skills and misconceptions are not obtained live from the LLM at hand but inserted from the mapping built in Step 1 (§3.1). The objective of role-playing is to generate a set of synthetic student–question interactions, that reflect realistic performance patterns across different ability levels and can then serve as input to downstream psychometric models, such as IRT, for QDE.

In role-playing, the evaluation set of questions is combined with a predefined number of proficiency levels (five in our experiments, though this choice is flexible). Each question is answered once per proficiency level. For example, with five levels and 20 questions, the model produces 100 simulated interactions. Depending on data availability, the previous interactions shown can be either MCQs or open-ended responses, and this affects how student proficiency levels are obtained (see §4.1).

Within role-playing, the LLM adopts one of two personas. i) *Student Persona*: the LLM is instructed to act as a student at a given proficiency level, reasoning from their skills and misconceptions to provide an answer to a new question. ii) *Teacher Persona*: the model is instructed to act as a teacher, using knowledge of a student's skills and misconceptions to predict how that student would respond to a new question. Prompt templates for both personas are available in Appendix A.2.

3.3. Student Behavior Replication

Student replication evaluates the LLM's ability to simulate the behavior of *one specific student*, rather than a prototypical student at a certain proficiency level. In this sense, it provides a more direct signal of the student-simulation capability of the model. Since replication requires a set of previous student responses to the questions under virtual pretesting, it cannot be performed on all datasets. Our framework uses it as an optional post hoc evaluation step when such data is available. Experimenting with

replication also addresses a gap from prior work, since previous students' interactions have been ignored in previous research. As in role-playing, the LLM is prompted under both the *Student* and *Teacher* personas.

Replication shares similarities with Knowledge Tracing (KT) (Shen et al., 2024), which predicts students' responses in a sequence of learning interactions, but differs as it treats proficiency as static, without modeling temporal learning dynamics.

3.4. Example selection strategies

A central component of our framework is the dynamic construction of contextual information about skills and misconceptions (the examples for ICL). Each example selection strategy returns a list of interactions, which are mapped (via the dictionary described in §3.1) to skills and misconceptions, and inserted into the LLM's context. We distinguish between group-based selectors, used in role-playing, and individual-based selectors, used in replication.

Group-Based Selectors use the interaction history of a group of students of the same proficiency level. i) *Random*: Randomly sample k interactions from the pool of students at the target level. ii) *Knowledge Concepts (KCs)*: from all questions answered by students at that level, select the k questions whose KCs are most similar to the KCs of the target question.

Individual-Based Selectors use the interaction history of a single student. i) *Random*: Randomly sample k past interactions from the target student. ii) *KCs*: From all questions previously answered by the student, select the k most similar to the target question, returning the corresponding k student interactions. Note that the KC selector can only be used if the dataset provide information on the KC (i.e., the topics/skills) associated with the questions.

We deliberately restrict our selection strategies to those that are feasible for role-play. For example, a Recency selector—commonly used in KT—would exploit temporal ordering, but this does not meaningfully extend to role-play across a group of students, as they are on different learning trajectories. While this might seem a limitation, we focus on role-play because it is easier to control (we model student personas rather than individuals).

Notably, while our methodology fits within the broader paradigm of in-context learning for prompt engineering (Dong et al., 2024), it differs from conventional few-shot prompting. Instead of presenting standard input–output exemplars, the prompts provide contextualized representations of skills and misconceptions relevant to the target interaction.

4. Experimental Setup

4.1. Datasets

We evaluate the framework on two publicly available MCQ datasets: DBE-KT22 and Cam-MCQ. They differ in domain (computer science and language learning) and in type of context (MCQ and short essays).

4.1.1. DBE-KT22

DBE-KT22 (Abdelrahman et al., 2022), collected from a relational databases course at the Australian National University, is one of the only publicly available resources that provides both student–question interaction data and the corresponding question text with answer options. The dataset consists of MCQs in English, for which we compute question difficulty using IRT, regarded as the gold standard for the regression-based QDE task.

To ensure consistency, we restrict the dataset to questions with four answer options and students with at least 30 recorded interactions. For students who answered the same question multiple times, only their first attempt is retained. The dataset is split by question ID into training (92 questions), validation (23), and test (39) sets.

The training interactions (i.e., students' answer to training questions) are used for ICL and to compute the student proficiency levels; they include 65,494 responses from 988 students. The evaluation sets are subsampled and stratified across five proficiency levels: small validation (100 interactions), large validation (500), and test set (500).

4.1.2. Cambridge MCQ and FCE

The Cambridge MCQ reading dataset (Cam-MCQ) (Mullooly et al., 2023) is a dataset of reading comprehension MCQs from English exams. Each question comes with an IRT measurement of its difficulty, as obtained from pretesting with real learners. This can be compared with the difficulty obtained from virtual pretesting with LLMs. Unlike DBE-KT22, it does not provide individual student-question interactions. Thus, we extract student skills and misconceptions for varying student levels from the FCE dataset (Yannakoudakis et al., 2011). FCE contains 314 short essays written by students taking an English certification exam, and provides error annotations and grades for each student response; all responses are from different students. We group student responses into five proficiency levels based on their grading, to use them for role-playing.

Cam-MCQ is split by question ID into validation (25 questions) and test (75) sets, and the whole FCE dataset is used as train set. Also, since

Cam-MCQ does not provide individual student responses, we can only evaluate the LLMs' role-playing performance, not replication performance.

4.2. Evaluation Metrics

For **student role-play** we employ two metrics: we either i) assess the difficulty estimated from virtual pretesting with IRT, or ii) directly assess the correctness of LLMs' responses to the questions.

We primarily evaluate QDE, which is a regression task, comparing the gold standard difficulty and the LLM's estimates from virtual pretesting. We use Root Mean Squared Error (RMSE), which is the most common metric in the literature for continuous difficulty levels (Benedetto et al., 2023).

To directly assess the LLM's interactions simulated with role-playing, we use the Monotonicity metric (M) (Benedetto et al., 2024). This builds on the idea that students of higher (simulated) levels should answer more questions correctly than those of lower levels. Therefore, it evaluates the correlation $\rho_{\mathbf{L},\mathbf{T}}$ between the LLM's response correctness per student level $\mathbf{L} = (a_1, a_2, \ldots, a_5)$ and those observed in the training set $\mathbf{T}$, penalizing non-monotonic behavior in the correctness sequence. The penalty for non-monotonicity (P) is calculated as: $\sum_{i=1}^{4} \sqrt{|a_{i+1} - a_i|} \cdot \mathbb{I}(a_{i+1} < a_i)$, where $\mathbb{I}(\cdot)$ is an indicator function. The metric is the difference between the correlation score and the penalty for non-monotonicity: $M = \rho_{\mathbf{L},\mathbf{T}} - P$.

Student replication is essentially a four-way classification task, as the LLM is prompted to predict which of the four answer options the simulated student will select. Thus, we evaluate it using the balanced accuracy between the LLM's response and the actual student's response.

4.3. Configurations

We experiment with the open-weight model families Qwen3 (0.6B, 1.7B, 4B, 8B, 14B) (Yang et al., 2025) and Llama (3.2:1B, 3.2:3B, 3.1:8B) (Dubey et al., 2024). These differ from the closed-source o4-mini model used in §3.1, where restrictive rate limits made large-scale experimentation across multiple seeds infeasible.

For DBE-KT22, we first conduct Student Role-play on the validation set and find the best configuration for each model. We consider: two prompt personas (Student and Teacher), two example selectors (Random and Knowledge Concepts), three context sizes (1, 3, and 5 examples), and two temperature settings (0.0 and 1.0), for a total of 24 configurations per model. Notably, prior studies on virtual pretesting have fixed the temperature to 0.0, whereas we additionally explore 1.0. From these experiments, we identify the best-performing configuration per model, which is then evaluated on the test set and the replication test set.

For Cam-MCQ, we follow a similar procedure, with two notable differences: i) the number of configurations is smaller, and ii) there is no replication stage. The configurations are: two prompt personas (Student and Teacher), one example selector (Random), one context size (1 example), and two temperature settings (0.0 and 1.0), which yields 4 configurations per model. We restrict the context to one example because open-ended responses are much more dense in information about the student's understanding, and we cannot use the Knowledge Concept example selector as KC information is not available in Cam-MCQ. Similar to DBE-KT22, we find the best-performing configuration for each LLM (prompt persona and temperature) on the role-playing validation set and evaluate and their performance on the role-playing test set.

As a primary baseline, we adopt the zero-shot LLM simulation, reflecting the dominant setup in prior role-playing studies (§2). All reported results are averaged over three independent runs with different random seeds.

5. Results and Analysis

5.1. Student Behavior Role-play

This section evaluates the role-playing performance of contextual LLMs in simulating student behavior and compares it with that of non-contextual models, which are the dominant approach in prior research. We report the best-performing configurations on the test set, including the associated hyperparameters, after model selection on the validation set. Results for DBE-KT22 are shown in Table 1, and results for Cam-MCQ in Table 2 (please note that the datasets have different difficulty ranges).

On DBE-KT22 (Table 1), contextual LLMs generally outperform their non-contextual counterparts in terms of difficulty estimation RMSE. Also, among the five best-performing configurations overall (highlighted in bold, with overlapping standard error bounds), four are contextual. For the Monotonicity metric, however, performance differences are less pronounced due to relatively high standard errors, and several configurations achieve comparable results. This suggests that while contextualization tends to improve predictive accuracy (RMSE), its impact on monotonic behavior is less consistent—but this might also be a consequence of the specific penalty used by the Monotonicity metric. Regarding hyperparameters, the KC example selector is preferred over the Random selector, meaning that questions on similar topics seemingly provide better contextual information, and a temperature of 1.0 consistently yields better results than

Model	Size	Type	RMSE ↓	Monotonicity ↑	Persona	Selector	Size	Temp.
qwen3	0.6 B	Context	**1.696** ± 0.040	0.510 ± 0.062	Teacher	KC	1	1.0
		No context	2.019 ± 0.024	-0.031 ± 0.681	Student	—	0	1.0
	1.7 B	Context	2.118 ± 0.055	**0.445** ± 0.294	Student	KC	5	1.0
		No context	1.952 ± 0.042	-0.236 ± 0.058	Teacher	—	0	0.0
	4 B	Context	**1.720** ± 0.044	**0.608** ± 0.156	Teacher	KC	5	1.0
		No context	2.265 ± 0.068	0.271 ± 0.355	Student	—	0	1.0
	8 B	Context	2.017 ± 0.140	**0.735** ± 0.123	Teacher	KC	3	1.0
		No context	**1.682** ± 0.073	**0.849** ± 0.142	Student	—	0	1.0
	14 B	Context	**1.700** ± 0.142	**0.841** ± 0.102	Teacher	KC	3	1.0
		No context	1.965 ± 0.046	**0.554** ± 0.212	Student	—	0	1.0
llama3.2	1 B	Context	**1.594** ± 0.083	**0.674** ± 0.168	Student	Random	1	0.0
		No context	1.739 ± 0.044	0.511 ± 0.156	Teacher	—	0	1.0
	3 B	Context	1.797 ± 0.088	0.388 ± 0.098	Teacher	KC	3	1.0
		No context	1.981 ± 0.084	**0.646** ± 0.210	Teacher	—	0	1.0
llama3.1	8 B	Context	1.843 ± 0.116	0.482 ± 0.138	Student	Random	1	1.0
		No context	1.765 ± 0.074	**0.690** ± 0.037	Teacher	—	0	1.0

Table 1: Student role-play for DBE-KT22. Best results indicated in bold; best result per model is underlined.

Model	Size	Type	RMSE ↓	Monotonicity ↑	Persona	Selector	Size	Temp.
qwen3	0.6 B	Context	15.061 ± 0.532	0.610 ± 0.100	Student	Random	1	0.0
		No context	11.175 ± 0.000	—	Student	—	0	0.0
	1.7 B	Context	12.366 ± 0.149	0.272 ± 0.136	Student	Random	1	1.0
		No context	**10.498** ± 0.004	**0.539** ± 0.327	Student	—	0	0.0
	4 B	Context	12.036 ± 0.116	0.173 ± 0.167	Student	Random	1	0.0
		No context	14.162 ± 0.640	0.281 ± 0.218	Student	—	0	1.0
	8 B	Context	15.315 ± 0.427	**0.921** ± 0.060	Teacher	Random	1	1.0
		No context	15.168 ± 0.321	0.748 ± 0.089	Teacher	—	0	1.0
	14 B	Context	14.590 ± 0.845	**0.848** ± 0.111	Student	Random	1	1.0
		No context	14.546 ± 0.134	**0.743** ± 0.150	Student	—	0	1.0
llama3.2	1 B	Context	14.370 ± 0.708	0.148 ± 0.240	Student	Random	1	0.0
		No context	13.735 ± 0.553	-0.882 ± 0.316	Teacher	—	0	1.0
	3 B	Context	14.425 ± 0.365	0.688 ± 0.042	Student	Random	1	0.0
		No context	13.864 ± 0.174	0.403 ± 0.118	Student	—	0	0.0
llama3.1	8 B	Context	13.524 ± 0.435	0.410 ± 0.222	Student	Random	1	1.0
		No context	13.138 ± 0.264	**0.773** ± 0.196	Teacher	—	0	1.0

Table 2: Student role-play for Cam-MCQ. Best results indicated in bold; best result per model is underlined.

0.0. For the configurations with a Random selector, showing one example performs best, possibly because more random examples merely add noise. No clear trends emerge for the prompt persona.

On Cam-MCQ (Table 2) the pattern is different: non-contextual models generally achieve lower RMSE than their contextual variants, while Monotonicity scores are comparable across both settings. Again, strong RMSE performance does not always align with high Monotonicity scores. For instance, the contextual qwen3:8B achieves an excellent Monotonicity score of 0.921 but poor RMSE (15.315), and the non-contextual qwen3:0.6B exhibits a strong RMSE but undefined Monotonicity, as it predicts all responses as correct—an undesirable behavior. In terms of hyperparameter trends, the Student persona is more frequently selected than the Teacher persona, while temperature values of 1.0 and 0.0 perform similarly. Because only a Random selector of size 1 was used, due to the characteristics of the dataset, the example selector does not provide further insight.

In general, the results do not indicate consistent evidence that contextualization per se enhances LLM performance in student role-playing: while contextual models generally outperform non-

contextual counterparts on DBE-KT22 on RMSE, the same trend is not visible on Cam-MCQ. This might suggest that contextual models are most beneficial when some interactions data is already available for questions of the same type as the ones under pretesting. Indeed, contextual information from different types of questions (the short essays from FCE) does not prove particularly helpful in our experiments. This might also be due to the fact that only the Random selector could be implemented for Cam-MCQ, thus it was less likely to provide contextual information really relevant for the questions under pretesting, and that production skills of the students (from the short essays) does not transfer directly to reading comprehension skills.

Model family	Type	RMSE ↓
DBE-KT22		
qwen3	Context	**1.177** ± 0.009
	No context	1.383 ± 0.019
llama3.1-3.2	Context	1.490 ± 0.076
	No context	1.554 ± 0.035
Cam-MCQ		
qwen3	Context	12.876 ± 0.376
	No context	**10.877** ± 0.083
llama3.1-3.2	Context	11.619 ± 0.631
	No context	**11.308** ± 0.474

Table 3: Hybrid multi-LLM role-playing results on DBE-KT22 (top) and Cam-MCQ (bottom).

5.2. Multi-LLM Student Role-play

To examine whether simulating only five virtual students limits the quality of the IRT-based difficulty estimation, we assess the impact of increasing the number of simulated students by performing role-playing in a Multi-LLM setting. We refer to this as *Hybrid Multi-LLM role-playing* because we perform role-play with individual models and aggregate simulations from different models. Specifically, we aggregate all student role-play simulations produced by models within the same family (qwen3 and llama3.1–3.2). In doing so, we create a larger and more diverse pool of virtual learners, while still significantly reducing the number of models compared to Multi-LLM approaches from previous research. The qwen3 family comprises five model sizes, each generating five virtual student levels, for a total of 25 virtual students. The llama3.1–3.2 family includes three models, producing 15 virtual students. The results are shown in Table 3.

On DBE-KT22, combining the contextual qwen3 simulations leads to a substantial improvement in predictive performance: the RMSE decreases from 1.696 (best individual contextual model) to 1.177 after merging (a 30% reduction in RMSE). This score is substantially lower than that of any other merged configuration. The non-contextual qwen3 aggregation also benefits from merging, though its improvement is less pronounced (18%). In contrast, the llama3.1–3.2 family exhibits only modest gains from merging (7% and 11%).

On Cam-MCQ, merging qwen3 simulations—both contextual and non-contextual—produces RMSE values comparable to the best individual models, without significant improvements. For llama3.1–3.2, the effect of merging is more noticeable: combining simulations narrows the performance gap with the qwen3 family and improves over the corresponding single-model results.

The benefit of the multi-LLM role-playing approach depends on the variability among individual models but, overall, aggregating diverse LLM-based student simulations enhances the robustness and accuracy of virtual pretesting. Merged simulations perform at least as well as the best individual configuration, and in some cases significantly better (e.g., qwen3 on DBE-KT22).

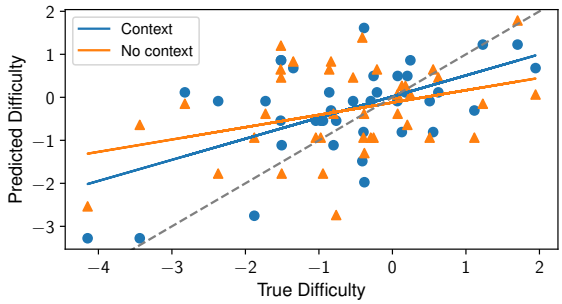
To provide an overview of the trends, Figure 2 shows scatterplots of the predicted (y-axis) and true (x-axis) question difficulty. The solid lines are regression lines, and the dashed line represents the ideal relationship. The differences between the models are immediately visible: on DBE-KT22, the contextual qwen3 family performs fairly well, and outperforms the non-contextual alternative. On the other hand, the llama3.1-3.2 family leads to regression lines that are much flatter, and the non-contextual model performs better.

5.3. Student Behavior Replication

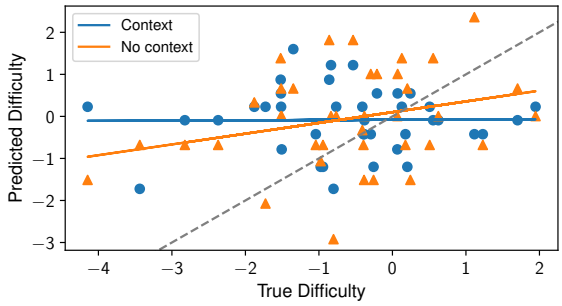
This section evaluates the ability of both contextual and non-contextual LLMs to replicate actual student response behavior on DBE-KT22. As in the student role-play task, non-contextual models serve as baselines. Table 4 reports the test set results after model selection based on student role-play performance, as described in §4.3. The student replication task can only be evaluated on DBE-KT22, since Cam-MCQ does not include individual student interaction data.

The results indicate that only a limited number of configurations achieve acceptable predictive accuracy, with balanced accuracy scores approaching 0.700. Most models perform around the 0.500 level, while llama3.1:8B performs even worse. Overall, replication performance remains modest across both contextual and non-contextual variants.

We find that model family may matter more than model size. Within qwen3, improvements are not strictly monotonic with size, e.g., qwen3:4B and 8B alternate in performance depending on context and dataset. Llama3.2:1B sometimes performs as



(a) DBE-KT22, qwen3.



(b) DBE-KT22, llama3.1-3.2.

Figure 2: Hybrid multi-LLM role-playing results on DBE-KT22. Each point of the scatter plot represents a question, the x-axis is the *true* difficulty from pretesting (with human learners) and y-axis the difficulty from virtual pretesting.

Model	Size	Type	Bal. acc. ↑
qwen3	0.6 B	Context	0.490 ± 0.010
		No context	0.649 ± 0.005
	1.7 B	Context	**0.681** ± 0.004
		No context	**0.683** ± 0.001
	4 B	Context	0.537 ± 0.001
		No context	0.551 ± 0.008
	8 B	Context	0.623 ± 0.002
		No context	0.561 ± 0.008
	14 B	Context	0.530 ± 0.013
		No context	0.508 ± 0.007
llama3.2	1 B	Context	0.672 ± 0.003
		No context	0.509 ± 0.007
	3 B	Context	0.451 ± 0.023
		No context	0.486 ± 0.007
llama3.1	8 B	Context	0.377 ± 0.007
		No context	0.360 ± 0.011

Table 4: Student replication results on DBE-KT22. Best results indicated in bold; best result per model is underlined. Majority baseline is 0.250.

well as or better than much larger models. As such, scaling up model size does not guarantee better simulation or replication ability. Smaller models may be equally capable of capturing student-like response patterns, perhaps due to differences in instruction-tuning or alignment rather than capacity. This is in line with previous research that highlighted the challenges of forcing LLMs to make mistakes (Aher et al., 2023), sometimes referred to as the "curse of hyper-accuracy" (Benedetto et al., 2024).

A key finding is that performance on student role-play does not transfer to student replication. Configurations that achieved strong results in role-play often fail to reproduce the behavior of *individual* students. Among the five best role-play configurations (Table 1), only one—llama3.2:1B contextual—achieves solid replication performance (balanced accuracy 0.672). The others barely exceed baseline performance, with the best of them (qwen3:8B non-contextual) reaching 0.561. Similarly, there is little correspondence between the Monotonicity metric and replication accuracy, suggesting that these evaluation dimensions capture different aspects of model behavior. Interestingly, the reverse pattern is also observed: the best-performing configurations for replication, qwen3:1.7B (both contextual and non-contextual), did not stand out in the role-play results. This discrepancy highlights a misalignment between models that simulate plausible "prototypical" students and those capable of reproducing the variability of real learners.

These findings show that datasets generated with this framework cannot capture *all* the complexity and diversity of individual responses. However, the plausible and consistent responses that LLMs produce with role-playing, especially in Hybrid Multi-LLM Role-playing, could be used (with caution) for a preliminary evaluation or low-stakes settings.

6. Conclusion

This work investigates methods to generate datasets of synthetic student-question interactions for virtual pretesting. We enhance LLM-based role-playing by enriching the context with information on students' previously acquired skills and misconceptions. Our results show that incorporating real student simulation data does not improve the quality of LLM-based student simulations *per se*, and it depends on the data available. Most notably, using students' responses on questions similar to the ones under pretesting leads to the best results, while using different types of questions (e.g., short essays as context for virtual pretesting comprehension MCQs) did not prove effective. While individual LLMs can approximate certain response patterns, their ability to generalize to authentic learner behavior remains limited. However, we find that increasing the number of virtual students through a hybrid multi-LLM role-playing approach consistently yields improved results, suggesting that ensemble-style

aggregation may enhance reliability. As such, role-playing LLMs could be used (with caution) for a preliminary evaluation of exam questions or in low-stakes settings. Still, we highlight the need for more behaviorally grounded evaluation frameworks for use in high-stakes settings.

Future research could evaluate larger open-weight models (e.g., qwen3:32B, llama3.3:70B) and state-of-the-art closed models, both for collecting misconceptions and role-playing, which may provide new insights into generalization, and longer contextual windows through Retrieval-Augmented Generation (RAG) (Lewis et al., 2020). Future work should assess not only accuracy but also cognitive plausibility, for instance by comparing simulated and real error patterns, which would offer a more comprehensive view of how well LLM-based students simulate genuine learning processes.

7. Ethical Considerations and Limitations

Previous studies have used LLMs to simulate the responses of human participants to surveys, raising some concerns about fairness (Harding et al., 2024; Crockett and Messeri, 2023). Indeed, LLMs might reproduce the response patterns of the majority of annotators, without capturing the individual differences between them and without representing less minority demographics. These issues are relevant and important to consider in educational contexts as well, but we argue that they may be less problematic than for general-domain surveys. Indeed, in most cases the learning materials and exam questions are designed to assess factual knowledge and are typically constructed to reduce the influence of wording on student performance (Ha et al., 2019). Still, when using a framework such as the one proposed in this paper for virtual pretesting in real educational settings, it is needed to continuously monitor how questions perform in the exams and whether there are any issues in the provided calibration (but this has to be done for manually generated and calibrated questions, too).

A limitation which is worth noting concerns the skills and misconceptions that are used in the proposed framework. In our experiments, we directly use the skills and misconceptions obtained from the reasoning LLM §3.1. We have qualitatively verified them, to understand whether the model was producing sensible skills and misconceptions, but at this stage we did not edit them nor filter them in any way. A more thorough analysis on the reliability of the LLM output could focus more on this aspect of the framework, possibly employing several domain experts and computing the inter-rater agreement. Indeed, these skills and misconceptions are the core of the contextual information that is provided to the role-playing LLMs, and they might hinder the accuracy of the simulations if they are not representative of previous responses. Finally, we limit the number of simulated student levels to five, following the examples of previous role-playing research (Benedetto et al., 2024), but we could investigate the effect of generating interactions with more levels, to increase the number of interactions for IRT estimation.

8. Acknowledgments

This study was supported by the Research Foundation Flanders (FWO) (grant number 1S97022N). Luca Benedetto started working on this project while funded by Cambridge University Press & Assessment. We thank the anonymous reviewers for their valuable feedback.

9. Bibliographical References

Ghodai Abdelrahman, Sherif Abdelfattah, Qing Wang, and Yu Lin. 2022. Dbe-kt22: A knowledge tracing dataset based on online student evaluation. *arXiv preprint arXiv:2208.12651*.

Gati V Aher, Rosa I Arriaga, and Adam Tauman Kalai. 2023. Using large language models to simulate multiple humans and replicate human subject studies. In *International conference on machine learning*, pages 337–371. PMLR.

Samah AlKhuzaey, Floriana Grasso, Terry R. Payne, and Valentina Tamma. 2021. A Systematic Review of Data-Driven Approaches to Item Difficulty Prediction. In Ido Roll, Danielle McNamara, Sergey Sosnovsky, Rose Luckin, and Vania Dimitrova, editors, *Artificial Intelligence in Education*, volume 12748, pages 29–41. Springer International Publishing, Cham.

Yigal Attali, Luis Saldivia, Carol Jackson, Fred Schuppan, and Wilbur Wanamaker. 2014. Estimating item difficulty with comparative judgments. *ETS Research Report Series*, 2014(2):1–8.

Lisa Beinborn, Torsten Zesch, and Iryna Gurevych. 2015. Candidate evaluation strategies for improved difficulty prediction of language tests. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1–11, Denver, Colorado. Association for Computational Linguistics.

Luca Benedetto. 2023. A quantitative study of nlp approaches to question difficulty estimation. In *International Conference on Artificial Intelligence in Education*, pages 428–434. Springer.

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.

Luca Benedetto, Andrea Cappelli, Roberto Turrin, and Paolo Cremonesi. 2020. R2DE: A NLP approach to estimating IRT parameters of newly generated questions. In *Proceedings of the Tenth International Conference on Learning Analytics & Knowledge*, pages 412–421, Frankfurt Germany. ACM.

Luca Benedetto, Paolo Cremonesi, Andrew Caines, Paula Buttery, Andrea Cappelli, Andrea Giussani, and Roberto Turrin. 2023. A survey on recent approaches to question difficulty estimation from text. *ACM Computing Surveys*, 55(9):1–37.

Molly Crockett and Lisa Messeri. 2023. Should large language models replace human participants?

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024. A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1107–1128, Miami, Florida, USA. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

Wanyong Feng, Peter Tran, Stephen Sireci, and Andrew S. Lan. 2025. Reasoning and Sampling-Augmented MCQ Difficulty Prediction via LLMs. In Alexandra I. Cristea, Erin Walker, Yu Lu, Olga C. Santos, and Seiji Isotani, editors, *Artificial Intelligence in Education*, volume 15880, pages 31–45. Springer Nature Switzerland, Cham.

Le An Ha, Victoria Yaneva, Peter Baldwin, and Janet Mee. 2019. Predicting the difficulty of multiple choice questions in a high-stakes medical exam. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 11–20, Florence, Italy. Association for Computational Linguistics.

Ronald K Hambleton, Hariharan Swaminathan, and H Jane Rogers. 1991. *Fundamentals of item response theory*, volume 2. Sage.

Jacqueline Harding, William D'Alessandro, NG Laskowski, and Robert Long. 2024. Ai language models cannot replace human research participants. *Ai & Society*, 39(5):2603–2605.

Fu-Yuan Hsu, Hahn-Ming Lee, Tao-Hsing Chang, and Yao-Ting Sung. 2018. Automated estimation of item difficulty for multiple-choice tests: An application of word embedding techniques. *Information Processing & Management*, 54(6):969–984.

Suzanne Lane, Mark R Raymond, Thomas M Haladyna, et al. 2016. *Handbook of test development*, volume 2. Routledge New York, NY.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Naiming Liu, Shashank Sonkar, and Richard Baraniuk. 2025. Do llms make mistakes like students? exploring natural alignments between language models and human error patterns. In *International Conference on Artificial Intelligence in Education*, pages 364–377. Springer.

Ekaterina Loginova, Luca Benedetto, Dries Benoit, and Paolo Cremonesi. 2021. Towards the application of calibrated transformers to the unsupervised estimation of question difficulty from text. In *RANLP 2021*, pages 846–855. INCOMA.

Hotaka Maeda. 2025. Field-testing multiple-choice questions with ai examinees: English grammar items. *Educational and Psychological Measurement*, 85(2):221–244.

Andrew Mullooly, Øistein Andersen, Luca Benedetto, Paula Buttery, Andrew Caines, Mark JF Gales, Yasin Karatay, Kate Knill, Adian Liusie, Vatsal Raina, et al. 2023. The cambridge multiple-choice questions reading dataset.

Jae-Woo Park, Seong-Jin Park, Hyun-Sik Won, and Kang-Min Kim. 2024. Large language models are students at various levels: Zero-shot question difficulty estimation. In *Findings of the Association for Computational Linguistics: EMNLP*

2024, pages 8157–8177, Miami, Florida, USA. Association for Computational Linguistics.

Andreas Säuberli, Diego Frassinelli, and Barbara Plank. 2025. Do llms give psychometrically plausible responses in educational assessments? *arXiv preprint arXiv:2506.09796*.

Shuanghong Shen, Qi Liu, Zhenya Huang, Yonghe Zheng, Minghao Yin, Minjuan Wang, and Enhong Chen. 2024. A survey of knowledge tracing: Models, variants, and applications. *IEEE Transactions on Learning Technologies*, 17:1858–1879.

Arthur Thuy, Ekaterina Loginova, and Dries F Benoit. 2024. Active learning to guide labeling efforts for question difficulty estimation. *arXiv preprint arXiv:2409.09258*.

Arthur Thuy, Ekaterina Loginova, and Dries F Benoit. 2025. Ordinality in discrete-level question difficulty estimation: Introducing balanced drps and orderedlogitnn. In *Second Workshop on Automated Evaluation of Learning and Assessment Content*, volume Vol. 4006.

Masaki Uto, Yuto Tomikawa, and Ayaka Suzuki. 2025. Question difficulty prediction based on virtual test-takers and item response theory. In *Workshop on Automated Evaluation of Learning and Assessment Content*, volume Vol. 3772.

Wim J Van der Linden and Cees AW Glas. 2000. *Computerized adaptive testing: Theory and practice*. Springer.

Lev Semenovich Vygotsky and Michael Cole. 1978. *Mind in society: Development of higher psychological processes*. Harvard university press.

Victoria Yaneva, Peter Baldwin, Janet Mee, et al. 2019. Predicting the difficulty of multiple choice questions in a high-stakes medical exam. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 11–20.

Victoria Yaneva, Peter Baldwin, Janet Mee, et al. 2020. Predicting item survival for multiple choice questions in a high-stakes medical exam. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 6812–6818.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading esol texts. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 180–189.

A. Prompts

A.1. Collecting skills and misconceptions

Tables 5 and 6 show the prompt template used to collect skills and misconceptions on the DBE-KT22 and FCE datasets.

A.2. Student Behavior Replication & Roleplay

Tables 7 and 8 show the prompt templates used in Replication and Roleplay, for the Student and Teacher personas, for the contextual and non-contextual settings.

Prompt
SYSTEM: You are shown a multiple choice question of an exam on {exam_type}. You have to analyse the question as follows: - Skills: for the correct answer option, list the knowledge concepts that the student should know to correctly select it; - Misconceptions: for each distractor, list the misconceptions that might lead the student to select it; Your answers should be very concise; each field should have a maximum of 10 words. The skill description should start with 'Understands' and the misconception description should start with 'Confuses'. USER: Multiple choice question: {input}

Table 5: Prompt template for collecting skills and misconceptions on the DBE-KT22 dataset. The variable `exam_type` is the string "database systems (Department of Computer Science)"; the variable `input` represents the multiple-choice question.

Prompt
SYSTEM: You are shown a short essay in English that a student has written. Inspect the essay and list the skills and misconceptions that the student has. Your answers should be very concise; each field should have a maximum of 10 words. The skill description should start with 'Understands' and the misconception description should start with 'Confuses'. Specifically, focus on skills and misconceptions that would transfer from writing to reading comprehension. List up to 3 skills and up to 3 misconceptions. If there are no skills or misconceptions, return 'None'. USER: {input}

Table 6: Prompt template for collecting skills and misconceptions on the FCE dataset. The variable `input` represents the short essay.

Type	Prompt
Student - contextual	SYSTEM: You are a student of level {student_level_group} {student_scale} working on an exam on {exam_type}, containing multiple choice questions. USER: Below is a list of multiple-choice questions you have answered earlier: {interactions} USER: Inspect the multiple-choice questions and list the skills and misconceptions that you have. For each question, list 1 skill or 1 misconception, depending on whether the question was answered correctly. If there are no skills or misconceptions, return 'None'. ASSISTANT: {skills_misconceptions} USER: Inspect the following new multiple-choice question: {input} How would you answer this question as a student of level {student_level_group}? Think about how your student level, skills, and misconceptions relate to the question difficulty and what mistakes you are likely to make. You can answer incorrectly, if that is what you are likely to do for this question.
Student - no context	SYSTEM: You are a student of level {student_level_group} {student_scale} working on an exam on {exam_type}, containing multiple choice questions. USER: Inspect the following new multiple-choice question: {input} How would you answer this question as a student of level {student_level_group}? Think about how your student level relates to the question difficulty and what mistakes you are likely to make. You can answer incorrectly, if that is what you are likely to do for this question.

Table 7: Prompt template for student personas. The variable `exam_type` is the string "database systems (Department of Computer Science)" or "English reading comprehension"; the variable `student_scale` is the string "(of levels 1. Fundamental Awareness, 2. Novice, 3. Intermediate, 4. Advanced, 5. Expert)" for all configurations. The variable `input` represents the multiple-choice question or short essay; `skills_misconceptions` represents the skills and misconceptions gathered earlier and are inserted in the message conversation as an AI message.

Type	Prompt
Teacher - contextual	SYSTEM: You are an expert teacher preparing a set of multiple choice exam questions on {exam_type}. USER: You have a student in your class of level {student_level_group} {student_scale}. Below is a list of multiple-choice questions they have answered earlier: {interactions} USER: Inspect the multiple-choice questions and list the skills and misconceptions that the student has. For each question, list 1 skill or 1 misconception, depending on whether the question was answered correctly. If there are no skills or misconceptions, return 'None'. ASSISTANT: {skills_misconceptions} USER: Inspect the following new multiple-choice question: {input} How would the student of level {student_level_group} answer this question? Think about how the student level, skills, and misconceptions relate to the question difficulty and what mistakes the student is likely to make. You can answer incorrectly, if that is what the student is likely to do for this question.
Teacher - no context	SYSTEM: You are an expert teacher preparing a set of multiple choice exam questions on {exam_type}. USER: Inspect the following new multiple-choice question: {input} You have a student in your class of level {student_level_group} {student_scale}. How would the student of level {student_level_group} answer this question? Think about how the student level relates to the question difficulty and what mistakes the student is likely to make. You can answer incorrectly, if that is what the student is likely to do for this question.

Table 8: Prompt template for teacher personas. The input variables are identical to Table 7. The variable `exam_type` is the string "database systems (Department of Computer Science)" or "English reading comprehension"; the variable `student_scale` is the string "(of levels 1. Fundamental Awareness, 2. Novice, 3. Intermediate, 4. Advanced, 5. Expert)" for all configurations. The variable `input` represents the multiple-choice question or short essay; `skills_misconceptions` represents the skills and misconceptions gathered earlier and are inserted in the message conversation as an AI message.