Hybrid Evaluation of Tutor Dialogues

Ulrike Padó[0009−0000−0664−7487]

Hochschule für Technik Stuttgart, Schellingstr. 24, 70174 Stuttgart, Germany
ulrike.pado@hft-stuttgart.de

Abstract. Tutorial dialogue, e.g., with LLM-based chatbots, offers interactive, flexible learning support. However, during development and for quality control it is imperative to closely monitor the tutor's factual correctness and didactic alignment. To save human effort, an LLM judge can be used. We propose to back off from the cutting edge of research to a low cost, low development effort LLM judge. This raises concerns about (a) evaluation quality and (b) black-box effects where the educator has no insight into the tutorial dialogues. We address both points by applying a hybrid human-LLM judge evaluation process. We show that it can deliver reliable evaluation results using a low-cost judge model, at reduced human effort that preserves human review.

Keywords: LLM tutor · LLM-as-a-judge · evaluation

1 Introduction

Tutorial dialogue is an effective tool in teaching, and chatbots are becoming increasingly popular and comparatively easily accessible tutor implementations [12]. However, educators need to be sure that chatbot tutors give factually accurate answers as well as showing correct didactic alignment (for example, guiding students instead of giving exercise answers away). A straightforward and informative, but prohibitively time-consuming, evaluation strategy for both these performance dimensions is manual analysis of the conversation logs.

Therefore, machine learning models (most often, Large Language Models – LLMs) have been proposed to judge the tutors' behavior and answer quality, adding a second layer of automation. For instance, the Shared Task at the 2025 edition of the Building Educational Applications (BEA) workshop [1] addressed exactly this task [4]. A large number of participants submitted models, many of which show very promising results.

However, access to these research prototypes is of course unrealistic for the majority of educators. Therefore, the focus of this paper is how to carry over the insights from the "cutting edge" of current research to the breadth of the field of educators who wish to evaluate chatbot tutors. We experiment with simplified "blunt-edge" judge LLMs: We choose open source models for their low cost and replicability, do no fine-tuning or task-specific training and deploy standardized prompts derived from the literature.

[1] https://sig-edu.org/bea/2025

Our research question is how well such simplified, accessible models perform in comparison to "cutting-edge" models and how they can be used to effectively support human evaluation.

Since we find that performance of the blunt-edge models is much lower than for the optimized cutting-edge models, we adapt a hybrid human-machine process from the literature that was originally developed for supporting manual grading of free-text answers to exam questions [9]: Using the strengths of the automated system helps focusing human attention where the judge LLM is weak. This increases the reliability of evaluation and reduces human effort while still allowing educators an intuitive insight into their chatbots' behaviors and errors through hands-on review (see Section 3). The hybrid process consists of five steps, which we implement in this paper using publicly available data sets, open-source LLMs and prompts from the literature. We demonstrate that it generates high-quality output at vastly reduced human effort, despite the judge model's performance gap to the cutting edge of research.

2 Related Work

We focus on chatbots as implementations of tutor dialogue systems, because for several years now, chatbot use in education is rapidly gaining popularity[2, 7, 12]: LLM-based chatbots can flexibly take the part of a teacher or other human conversation partner in many situations without substantial changes in their architecture, allowing students to profit from individual, naturalistic interactions. One such area is preparation for oral exams (cf. [11]), another is language learning (cf. [16]), or the deployment of a tutor for class content (cf., e.g., [6, 10])).

Development and evaluation of such educational chatbots involves both the content quality of their outputs as well as the didactic alignment of their behavior. Both is to a large part controlled through the provided prompt, but actual performance can be hard to predict. Analyzing chatbot behavior requires extensive review of chatbot interaction logs, which takes time and effort. Instead, the "LLM as a judge" strategy [15, 5] can be adopted, such that LLMs review the behavior of the LLM-based chatbots.

For the preparation of correct and well-aligned chatbots as well as reliable judge LLMs, standardized benchmark data plays an important role. For the domain of tutoring dialogues in Mathematics, the MRBench data set has been developed [8]. It contains dialogues between students and tutors that are completed by several different human and LLM tutors. The quality of the completions is annotated along four didactically motivated dimensions: Mistake Identification (did the tutor spot the student's mistake?), Mistake Location (did the tutor correctly identify where the mistake happened?), Guidance (does the tutor provide appropriate guidance?) and Actionability (is it clear what the student should do next?). A recent competition on the basis of this data invited the development of cutting-edge judge LLMs to evaluate the tutors in MRBench V3 [4].

However, despite impressive results from this competition, there is no analogous end-user offering, fully automated processes remain error-prone, and over-

reliance on automated systems may hide from the educator how their chatbot behaves in practice. Instead of full automation, a hybrid approach that pairs some human review with automated processing combines the advantages of manual and automated review [1, 9, 14]. Hybrid approaches also break the potential circularity of evaluating LLM output with another LLM.

3 Hybrid Human-Machine Evaluation

Padó et al. in [9] propose a human-machine hybrid evaluation process which has the advantage of being applicable and easily parametrizable for any classification task, judge model and data set. It places educators' informed decisions about model performance on their own data at the center, such that the available human work time and the expected quality of the outcome can be balanced based on a reliable estimate. Translated to the judge LLM evaluation task, the hybrid evaluation process is as follows:

1. **Define** the maximum affordable disagreement between judge LLM and human annotation of tutor quality
2. **Collect** a data set, i.e., a set of tutor dialogues, with human annotation of the desired characteristic
3. **Choose** a judge LLM
4. **Analyze** the judge LLM's performance for all decision classes
5. **Decide** on the basis of Step 4 which decisions of the judge LLM to accept and which to manually revise for the final tutor evaluation

We will now go on to implement this process using low-cost blunt-edge LLM judge models on the recently published MRBench V3 benchmark data, in order to evaluate the applicability of the process for tutor LLM evaluation. In real life, this initial investment of time and effort pays off in the long run by substantially reduced review effort.

4 Step 1: Define Acceptable Quality

The first step is to define acceptable performance of the hybrid process. In our case, this is the remaining error rate of the LLM judgments after targeted human revision. This threshold helps decide whether acceptable performance can plausibly be reached in practice using the judge LLM in combination with human review.

In the Mistake Identification setting, for example, the task for the judge LLM is to find the cases where one of the eight tutors correctly identified the student's error, and the cases where they did not. While it might be natural to expect perfect performance by the judge LLM, humans do not agree perfectly on this task: For the MRBench data, substantial agreement for the annotation is reported both for the test set ($\kappa = 0.71$, [8]) as well as the development set ($\kappa = 0.65$,[4]), which indicates that the annotation is trustworthy. However,

perfect agreement ($\kappa = 1$) among several human judges is nearly impossible to reach in any annotation task.

Since we cannot expect any human or LLM judge to generate the exact gold judgment in every case, we set the bar for acceptable judge LLM performance on any tutor data set at 95% Accuracy (Accuracy measures the percentage of correct decisions made by the LLM). This level is somewhat arbitrary since sufficient data is not available to estimate typical human agreement, but it is very conservative: Note that the bar for the (unrelated) short-answer grading task is set to 85% Accuracy in accordance with observations of human agreement for that task by Padó et al. in [9].

5 Step 2: Collect a Data Set

The MRBench V3 data set [8] can be used directly. It contains 300 Mathematics tutoring dialogues in a development section and 191 more dialogues in a test section [4]. These dialogues are completed by seven tutoring LLMs and one human expert (we do not use the small data set for a novice human tutor offered with the development data). The tutor interactions are annotated according to the four dimensions of (correct) Mistake Identification, Location, Guidance and Actionability. Annotation can be "Yes", "No", or "To some Extent"; the majority of dialogues is annotated as "Yes" for all four tasks. In the following, we report the lenient two-class evaluation, counting "To Some Extent" as "Yes".

6 Step 3: Choose a Judge LLM

We describe the selection of the best blunt-edge judge LLM on the MRBench development set and its performance on the test data. For compatibility with the Shared Task results, we report the macro F_1 score, which weights the two classes "Yes" and "No" equally. This helps counteract the large class imbalance towards the "Yes" class, which would otherwise dominate the evaluation.

Blunt-Edge Model Selection To construct our blunt-edge model, we look for the best combination of three different open-source LLMs to be used as judges, and four different prompts. After initial experiments, we settled on Mistral[2], Gemma[3] and GPT-OSS[4]. The models are hosted for research purposes by Gesellschaft für wissenschaftliche Datenverarbeitung Göttingen (GWDG). Other models available there didn't reliably return the expected output format or had long processing time at the time of experimentation (more than 120 seconds per model call). We queried the models with the default parameters and a temperature of 0.01 (defining very low creativity and high adherence to high-probability output).

[2] https://huggingface.co/mistralai/Mistral-Large-Instruct-2407
[3] https://huggingface.co/google/gemma-3-27b-it
[4] https://huggingface.co/openai/gpt-oss-120b

Table 1. Blunt-edge model selection: Macro F_1 (lenient) on the development set for three models and four prompt variants (Baseline, Fan et al., Role+Example, Wang et al.) for the Mistake Identification, Location, Guidance and Actionability tasks. The best result for each task is in boldface, the second best result is underlined.

	Mistral				**Gemma**				**GPT-OSS**			
Task	Bsl	Fan	R+Ex	Wang	Bsl	Fan	R+Ex	Wang	Bsl	Fan	R+Ex	Wang
Id	54.45	**77.70**	74.02	71.07	51.76	<u>77.46</u>	74.70	70.94	68.06	65.42	68.79	60.83
Loc	47.49	64.77	62.97	61.71	51.06	67.92	63.18	61.32	64.40	<u>72.66</u>	**73.74**	69.52
Guid	61.92	64.35	62.24	64.71	61.87	<u>67.28</u>	66.46	62.04	60.50	63.60	**71.07**	66.83
Act	47.66	59.35	65.35	61.00	55.35	67.32	66.07	61.15	65.61	<u>69.14</u>	**70.16**	68.32

We evaluated four different prompts which represent prompting strategies of increasing sophistication (all prompts can be found in the Appendix). The **baseline prompt** just instructs the judge LLM to return 1 if the task in question is completed successfully by the AI in the given dialog, 0 otherwise. Next, a prompt template from **Fan et al.** [3] describes the context more clearly (an AI reacts to a student's mistake) and adds a more verbose task description. The third prompt template, **Role+Example**, assigns a role to the judge LLM ("you are an experienced Maths tutor") and gives examples how the tutor may succeed or fail in the task. The final template from **Wang et al.** [13] is the longest and most sophisticated. It sketches an approach where the LLM simulates a conversation among several experts, each of whom solves part of the task. We shortened the literature prompt to one example conversation of experts and added the instruction to return 0 or 1 according to the judgment decision. We used the RAGAS[5] libraries to implement tutor quality prediction and evaluation.

Table 1 shows the results for all four tasks in MRBench. GPT-OSS is the overall best **model**, but on the primary Identification task, Mistral clearly outperforms it. These two models have a similar amount of parameters (Mistral has 123 billion, GPT-OSS 120 billion). In comparison, Gemma is small at 27 billion parameters, but still achieves second-place results for two tasks and usually outperforms Mistral. This result is encouraging for anyone wishing to host models locally, as smaller models use fewer computing resources.

As expected, the terse Baseline **prompt** returns low results throughout. However, the very complex Wang et al. prompt also never performs best. All three models do best on the Fan et al. or Role+Ex. prompts, which describe the task context at limited length: The optimal combination for Location, Guidance and Action is GPT-OSS with the Role+Ex. prompt, but all models regularly do well with the even shorter Fan et al. prompt. We continue our evaluation with the best model for each task: Mistral/Fan et al. for the Identification task and GPT-OSS/Role+Ex. for the other tasks.

Blunt-Edge vs Cutting-Edge Table 2 compares the performance of the best blunt-edge models from the previous step to both the majority baseline (the hypothet-

[5] https://docs.ragas.io/en/latest/

Table 2. Blunt-edge vs. cutting-edge: Macro F_1 (lenient) on the test set for the majority baseline, the best blunt-edge models (with hypothetical rank in the Shared Task leaderboard) and the best Shared Task participants (cutting-edge, [4])

Task	Baseline	Blunt-Edge (Rank)	Cutting-Edge
Identification	45.22	77.78 (42/45)	91.85
Location	39.74	75.64 (12/32)	84.04
Guidance	39.95	77.48 (3/36)	78.60
Actionability	40.95	71.01 (27/30)	86.59

ical performance of a model that always predicts the more frequent class, "Yes" in our case) and the best-performing cutting-edge models from the BEA 2025 Shared Task, on the test set.

Not surprisingly, the blunt-edge models perform worse than the cutting-edge models. For the Identification and Actionability tasks, the performance difference is largest. However, for the Location task, the blunt-edge model would have placed 12th in the Shared Task competition by lenient Macro F_1. For the Guidance task, the blunt-edge model even is within 1.5 points F-Score of the Shared Task winner and would have placed third. The blunt-edge models also clearly outperform the majority baselines for all four tasks.

In sum, while investing time and effort in model and prompt optimization clearly places the cutting-edge models ahead, the simplified blunt-edge models can still hold their own and are even competitive for some tasks. We therefore investigate them further as realistic tool choices.

7 Step 4: Analyze Blunt-Edge Judge LLM Performance

We now turn to analyzing the chosen judge LLM's performance for each individual tutor in order to decide how to parametrize the hybrid evaluation process. We use the development data since the target annotations are not public for the test set. We look at seven LLM tutors and one human expert, ignoring the small number of human novice interactions in the MRBench development set. Note that we now focus on the Mistake Identification task only, as it forms the basis for the other three tasks covered by MRBench.

Table 3 shows the Accuracy of Mistral with the Fan et al. prompt for each of the eight tutors. From the Accuracy measures, it is clear that the judge LLM has an easier time evaluating some of the tutors than others. For example, judge LLM performance for the tutor models Mistral, Llama31405b and GPT4 is already above our performance threshold of 95% Accuracy from Step 1: The judge LLM disagrees with the human evaluation of these tutor LLMs at most ten times across the 300 tutorial dialogues, and on this basis, no human review is deemed necessary in the logic of the hybrid evaluation process. On the other end of the spectrum, the same judge LLM incurs about 20% errors for the Human Expert and the Phi3 model. Interestingly, in Mistake Identification, the Human Expert is top-ranked with the three easy-to-judge models and the Phi3 model does worst.

Table 3. Blunt-edge judge LLM evaluation quality for eight LLM tutors: Overall Accuracy (lenient) and class-wise Accuracy for classes Yes and No; development set.

Tutor	Acc	Acc_{Yes}	Acc_{No}
GPT4	96.3	96.3	**1.0**
Llama31405b	95.7	**97.0**	0.0
Mistral	95.3	**97.2**	40.0
Sonnet	91.3	**94.7**	42.1
Gemini	89.3	**91.1**	33.3
Llama318b	86.3	**88.9**	30.8
Phi3	82.0	63.8	**93.5**
Human Expert	79.0	**95.8**	13.1

The hybrid process suggests that we accept the predictions of the judge LLM for whichever class it predicts at higher class-wise Accuracy and review the predictions of the other class, correcting where necessary. Class-wise Accuracy shows which percentage of the model's predictions of that class are reliable. For example, when evaluating the Sonnet tutor, the judge's Accuracy for class "Yes" in Table 3 is $Acc_{Yes} = 94.7$, the Accuracy for "No" is $Acc_{No} = 42.1$. This indicates that about 95% of all predictions of "Yes" (i.e., the judge model believes that the tutor model has correctly identified the student's mistake) are correct, while only about 42% of all "No" votes are correct. Therefore, human reviewing effort should focus on this worse-performing class [9]. Except for the GPT4 and Phi3 tutors, the LLM judge is always most reliable for class "Yes" and usually does much worse for class "No".

8 Step 5: Simulate Hybrid Evaluation

We now turn to simulating the performance of the hybrid evaluation process and the human effort saved for the Mistake Identification task. We found that we need no extra human review (saving 100% of effort) for the Mistral, Llama31405b and GPT4 tutor LLMs (Table 4). Review is of course still possible if the educator wishes to look at a sample of dialogues to form an intuition about the conversations.

Human review is necessary for the remaining five models. For our simulation, we assume that the reviewer checks all of the judge LLM's predictions for the class it predicts less reliably (e.g., "No" when evaluating the Sonnet tutor).

The right-hand portion of Table 4 shows the evaluation quality achieved with human review in the hybrid evaluation process: For six out of the eight tutor models, results are now above our Accuracy threshold, at substantial effort savings up to 100% (when no human review is needed). Reviewing the judge LLM's predictions for the Human Expert and for the Phi3 model saves least human work compared to manual revision of the whole data set, but savings are still noticeable, since at most 40% of the dialogues need to be read.

Table 4. Simulated results of the hybrid evaluation process (on the development data): Accuracy for the judge LLM alone and for the hybrid process as well as effort saved compared to manual review of the full data set. Hybrid process results ≥ 95 in bold face.

Tutor	Acc_{judge}	Acc_{hybrid}	Effort saved
GPT4	96.3	**96.3**	100%
Llama31405b	95.7	**95.7**	100%
Mistral	95.3	**95.3**	100%
Sonnet	91.3	**95.0**	94%
Gemini	89.3	91.3	97%
Llama318b	86.3	89.3	96%
Phi3	82.0	**96.0**	60%
Human Expert	79.0	**96.7**	80%

For Gemini and Llama318b, the judge LLM makes so few predictions of class "No" that manual revision does not return enough corrected predictions to achieve an Accuracy ≥ 0.95. In these cases, the educator can continue to revise random samples of the predictions for class "Yes" until enough errors are found to reach the quality threshold. For Llama318b, 11% of all predictions of class "Yes" are in fact erroneous, and about 160 additional dialogues classed as "Yes" would need to be reviewed in order to eliminate enough of these errors to reach the Accuracy threshold. Effort saved would be reduced to 42%. For Gemini, 9% of all predictions of class "Yes" are wrong, and in order to reach the Accuracy threshold, about 125 additional dialogues would have to be reviewed, reducing the effort saved to 55%.

For both these models, the problem is predictable from the class-wise evaluation by low Recall measures for class "No" (which indicates which percentage of instances of this class are found by the judge LLM, not shown). For both models, Recall is only about 10%, which means that many actual tutor LLM errors are classed as correct and can only be found by revision of the large number of "Yes" dialogues. This means that an educator would have warning about the small number of unreliable-class predictions and could factor in the expectation that additional revision of the reliable-class predictions will be necessary for this judge LLM.

In sum, it is possible to reach the predefined Accuracy goal for all eight tutors. For the worst-performing combinations of judge LLM and tutor, more manual effort has to be invested, but the educator still saves substantial amounts of work compared to full manual review.

Given these simulations, an educator working with data similar to MRBench can now choose a promising combination of tutor and judge LLM depending on availability as well as their individual Accuracy goals and desired amount of human effort to be invested. Adjusting one of these thresholds may affect the other: Less human effort spent may cause lower Accuracy, and vice versa.

9 Discussion and Conclusions

With the growing popularity of chatbots in education (for example as tutors, conversation partners for language practice or mock examiners), verifying the correctness and appropriateness of chatbot answers becomes an important task for educators. Manual review requires high effort – therefore, we recommend to use the "LLM-as-a-judge" strategy.

We show that the judge LLM does not have to perform at the cutting edge of research in order to be useful for evaluation; in fact, a blunt-edge model, the freely available open-source Mistral combined with a relatively simple prompt (cf. [3]) is sufficient. This is due to our adaptation of the hybrid short-answer grading process from [9] to the tutor evaluation task. The process identifies the strengths of the judge LLM and allows the human reviewer to focus where the judge LLM is unreliable. In this way, human effort is saved and the educator has the opportunity to form an intuition about the tutor's dialogue behavior.

While the quality of the LLM judgments varies with the tutor to be evaluated, acceptable Accuracy of the final judgments can be reached for each model combination that we investigated. In two cases, this comes at the price of higher manual effort, but we also find that three combinations of tutor and judge LLM already fulfill our requirements without human review.

In Machine Learning, it is common for model performance to vary across different data sets. Therefore, we stress the recommendation made in [9] that educators should collect a small sample of their own annotated data to simulate the evaluation process for their specific data set and models in order to get a reliable estimate of the expected prediction quality and manual effort needed. This is especially true if their usage context differs from the MRBench scenario.

In sum, we have demonstrated that the relevant problem of chatbot tutor evaluation can be solved in a cost- and effort-effective way by integrating human and (reliable) machine judgments in a way that easily carries over to other types of tutor implementations.

Acknowledgments. The author gratefully acknowledges the LLM hosting services granted by the KISSKI project of Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen (GWDG).

Disclosure of Interests. The author has no competing interests to declare that are relevant to the content of this article.

Appendix

Prompts

Mistake Identification Variable parts of the prompts that were adapted to the specific tasks are *italicized*.

Baseline "Return 1 if *the AI identifies a mistake made by the human*. Else, return 0."

Fan et al. "The student's last utterance contains a mistake. The AI tutor responds to this mistake. *Your task is to assess whether the tutor's response successfully identifies the mistake made by the student. Your task is to evaluate the final tutor response and determine whether it successfully identifies the error in the student's reasoning.*"

Role + Example "You are an experienced Maths tutor who has taught many students by discussing their approach to their practice problems with them. You know all the common errors and misconceptions that can come up in such a setting. In the tutoring conversation you are given, the human makes a mistake in their calculations. Analyse the human's contribution, and identify the error. Then analyse the AI's final utterance. *Return 1 if the AI clearly identifies the mistake made by the human, for example by pointing out the mistake, by re-stating the original question, providing the correct solution or by asking a clarifying question in a way that is relevant to the conversation. Else return 0, for example if the AI just repeats the student's last statement, makes an incorrect calculation or if it refers to a new or different task.*"

Wang et al. "When faced with a task, begin by identifying the participants who will contribute to solving the task. Then, initiate a multi-round collaboration process until a final solution is reached. The participants will give critical comments and detailed suggestions whenever necessary. Here are some examples: — Example Task 1: Use numbers and basic arithmetic operations (+ - * /) to obtain 24. You need to use all numbers, and each number can only be used once. Input: 6 12 1 1 Participants: AI Assistant (you); Math Expert Start collaboration! Math Expert: Let's analyze the task in detail. You need to make sure that you meet the requirement, that you need to use exactly the four numbers (6 12 1 1) to construct 24. To reach 24, you can think of the common divisors of 24 such as 4, 6, 8, 3 and try to construct these first. Also you need to think of potential additions that can reach 24, such as 12 + 12. AI Assistant (you): Thanks for the hints! Here's one initial solution: (12 / (1 + 1)) * 6 = 24 Math Expert: Let's check the answer step by step. (1+1) = 2, (12 / 2) = 6, 6 * 6 = 36 which is not 24! The answer is not correct. Can you fix this by considering other combinations? Please do not make similar mistakes. AI Assistant (you): Thanks for pointing out the mistake. Here is a revised solution considering 24 can also be reached by 3 * 8: (6 + 1 + 1) * (12 / 4) = 24. Math Expert: Let's first check if the calculation is correct. (6 + 1 + 1) = 8, 12 / 4 = 3, 8 * 3 = 24. The calculation is correct, but you used 6 1 1 12 4 which is not the same as the input 6 12 1 1. Can you avoid using a number that is not part of the input? AI Assistant (you): You are right, here is a revised solution considering 24 can be reached by 12 + 12 and without using any additional numbers: 6 * (1 - 1) + 12 = 24. Math Expert: Let's check the answer again. 1 - 1 = 0, 6 * 0 = 0, 0 + 12 = 12. I believe you are very close, here is a hint: try to change the "1 - 1" to "1 + 1". AI Assistant (you): Sure, here is the corrected answer: 6 * (1+1) + 12 = 24 Math Expert: Let's verify the solution. 1 + 1 = 2, 6 * 2 = 12, 12 + 12 = 12. You used 1 1 6 12 which is identical to the input 6 12 1 1. Everything looks

good! Finish collaboration! Final answer: 6 * (1 + 1) + 12 = 24 — Now, identify the participants and collaboratively solve the following task step by step.

In the conversation you are given, the human makes a mistake in their calculations. Identify it and analyse the AI's final contribution. Return 1 if the AI identifies the student's error. Else, return 0."

Location The task descriptions in the Mistake Identification prompts are adapted to:

Baseline "Return 1 if the AI identifies the location of the student's error. Else, return 0."

Fan et al. "Your task is to assess whether the tutor's response successfully locates the mistake made by the student. Your task is to evaluate the final tutor response and determine whether it successfully identifies the error in the student's reasoning and its location."

Role + Example "Return 1 if the AI identifies the location of the student's error, for example by pointing out the error or by asking a question that points to the error. Else, return 0, for example if the AI accepts the student's incorrect answer or points the student towards a wrong part of their calculations that was correct."

Wang et al. "Return 1 if the AI identifies the location of the student's error. Else, return 0."

Guidance The task descriptions in the Mistake Identification prompts are adapted to:

Baseline "Return 1 if the AI provides guidance to the human how to fix their error. Else, return 0."

Fan et al. "Your task is to assess whether the tutor's response successfully provides guidance on how to fix the mistake made by the student. Your task is to evaluate the final tutor response and determine whether it successfully provides guidance on how to fix the error in the student's reasoning. If the AI is successful, return 1. Else, return 0."

Role + Example "Return 1 if the AI provides guidance to the student how to fix the error, for example by hinting at the solution, providing an explanation or asking a supporting question. Else, return 0, for example if the AI accepts the student's incorrect answer or points the student towards a wrong part of their calculations that was correct."

Wang et al. "Return 1 if the AI provides guidance on how to fix the student's error. Else, return 0."

Actionability The task descriptions in the Mistake Identification prompts are adapted to:

Baseline "Return 1 if the AI makes it clear to the student what they are supposed to do next. Else, return 0."

Fan et al. "Your task is to assess whether the tutor's response successfully makes it clear to the student what they are supposed to do next. Your task is to evaluate the final tutor response and determine whether it successfully makes it clear to the student what they are supposed to do next. If the AI is successful, return 1. Else, return 0."

Role + Example "Return 1 if the AI make it clear to the student what they are supposed to do next, for example by asking a supporting question, providing part of the solution or providing the path to the solution. Else, return 0, for example if the AI accepts the student's incorrect answer, is vague, unclear or a conversation stopper."

Wang et al. "Return 1 if the AI makes it clear to the student what they are supposed to do next. Else, return 0."

References

1. Condor, A.: Exploring automatic short answer grading as a tool to assist in human rating. In: Bittencourt, I., Cukurova, M., Muldner, K., Luckin, R., Millán, E. (eds.) Proceedings of the International Conference on Artificial Intelligence in Education, AIED 2020. pp. 74 – 79 (2020). https://doi.org/10.1007/978-3-030-52240-7_14
2. Debets, T., Banihashem, S.K., Joosten-Ten Brinke, D., Vos, T.E., Maillette de Buy Wenniger, G., Camp, G.: Chatbots in education: A systematic review of objectives, underlying technology and theory, evaluation criteria, and impacts. Computers & Education **234**, 105323 (2025). https://doi.org/10.1016/j.compedu.2025.105323
3. Fan, Y., Tan, C., Song, W.: BJTU at BEA 2025 shared task: Task-aware prompt tuning and data augmentation for evaluating AI math tutors. In: Kochmar, E., Alhafni, B., Bexte, M., Burstein, J., Horbach, A., Laarmann-Quante, R., Tack, A., Yaneva, V., Yuan, Z. (eds.) Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025). pp. 1073–1077. Association for Computational Linguistics, Vienna, Austria (Jul 2025). https://doi.org/10.18653/v1/2025.bea-1.82
4. Kochmar, E., Maurya, K., Petukhova, K., Srivatsa, K.A., Tack, A., Vasselli, J.: Findings of the BEA 2025 shared task on pedagogical ability assessment of AI-powered tutors. In: Kochmar, E., Alhafni, B., Bexte, M., Burstein, J., Horbach, A., Laarmann-Quante, R., Tack, A., Yaneva, V., Yuan, Z. (eds.) Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025). pp. 1011–1033. Association for Computational Linguistics, Vienna, Austria (Jul 2025). https://doi.org/10.18653/v1/2025.bea-1.77

5. Li, D., Jiang, B., Huang, L., Beigi, A., Zhao, C., Tan, Z., Bhattacharjee, A., Jiang, Y., Chen, C., Wu, T., Shu, K., Cheng, L., Liu, H.: From generation to judgment: Opportunities and challenges of LLM-as-a-judge. In: Christodoulopoulos, C., Chakraborty, T., Rose, C., Peng, V. (eds.) Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing. pp. 2757–2791. Association for Computational Linguistics, Suzhou, China (Nov 2025). https://doi.org/10.18653/v1/2025.emnlp-main.138
6. Liu, R., Zenke, C., Liu, C., Holmes, A., Thornton, P., Malan, D.J.: Teaching CS50 with AI: leveraging generative artificial intelligence in computer science education. In: Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1. pp. 750–756 (2024), https://dl.acm.org/doi/10.1145/3626253.3635427
7. Ma, W., Ma, W., Hu, Y., Bi, X.: The who, why, and how of AI-based chatbots for learning and teaching in higher education: A systematic review. Education and Information Technologies **30**, 7781–7805 (2025). https://doi.org/10.1007/s10639-024-13128-6
8. Maurya, K.K., Srivatsa, K.A., Petukhova, K., Kochmar, E.: Unifying AI tutor evaluation: An evaluation taxonomy for pedagogical ability assessment of LLM-powered AI tutors. In: Chiruzzo, L., Ritter, A., Wang, L. (eds.) 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). pp. 1234–1251. Association for Computational Linguistics, Albuquerque, New Mexico (Apr 2025). https://doi.org/10.18653/v1/2025.naacl-long.57
9. Padó, U., Eryilmaz, Y., Kirschner, L.: Short-answer grading for German: Addressing the challenges. International Journal of Artificial Intelligence in Education **34**(4) (2024). https://doi.org/10.1007/s40593-023-00383-w
10. Pampel, B., Martin, S., Padó, U.: From capable to trustworthy: Empowering educators to build reliable ai chatbots with retrieval-augmented generation. In: 17th International Conference (CSEDU 2025), Revised Selected Papers. Springer CCIS (to appear)
11. Silvestri, C., Roshal, J., Shah, M., Widmann, W., Townsend, C., Brian, R., L'Huillier, J., Navarro, S., Lund, S., Sathe, T.: Evaluation of a novel large language model (LLM)-powered chatbot for oral boards scenarios. Global Surgical Education **3**(112) (2024). https://doi.org/10.1007/s44186-024-00303-z
12. Swacha, J., Gracel, M.: Retrieval-augmented generation (RAG) chatbots for education: A survey of applications. Applied Sciences **15**(8), 4234 (2025). https://doi.org/10.3390/app15084234
13. Wang, Z., Mao, S., Wu, W., Ge, T., Wei, F., Ji, H.: Unleashing the emergent cognitive synergy in large language models: A task-solving agent through multi-persona self-collaboration. In: Duh, K., Gomez, H., Bethard, S. (eds.) Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers). pp. 257–279. Association for Computational Linguistics, Mexico City, Mexico (Jun 2024). https://doi.org/10.18653/v1/2024.naacl-long.15
14. Win Myint, P.Y., Lo, S.L., Zhang, Y.: Harnessing the power of AI-instructor collaborative grading approach: Topic-based effective grading for semi open-ended multipart questions. Computers and Education: Artificial Intelligence **7**, 100339 (2024). https://doi.org/10.1016/j.caeai.2024.100339
15. Zheng, L., Chiang, W.L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E.P., Zhang, H., Gonzalez, J.E., Stoica, I.: Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In: Proceedings of the

37th Conference on Neural Information Processing Systems (NeurIPS) (2023), https://dl.acm.org/doi/10.5555/3666122.3668142
16. Zheng, Y.B., Zhou, Y.X., Chen, X.D., Ye, X.D.: The influence of large language models as collaborative dialogue partners on EFL English oral proficiency and foreign language anxiety. Computer Assisted Language Learning pp. 1–27 (2025). https://doi.org/10.1080/09588221.2025.2453191