EvalConvoLearn: An Open-Source Framework for Evaluating Grounded Learner Simulations in Tutoring Conversations

Baptiste Moreau-Pernet BAPTISTE@LEVI.DIGITALHARBOR.ORG

Abstract

Conversational learner simulations are valuable tools for testing learning theories, evaluating instructional materials and automated tutors, or powering teachable agents. Recently, large language models (LLM) have enabled richer, more naturalistic interactions with simulated learners; however, no open framework exists for evaluating whether such simulations faithfully reproduce real learner behavior. We introduce **EvalConvoLearn**, an open-source framework that assesses learner simulations along two axes: *learning behavior* (skill-conditioned mastery outcomes) and *conversational quality* (talk moves, error type distributions, question rate, turn length). EvalConvoLearn measures how closely a simulated learner approximates answer distributions observed in data by grounding metrics in authentic tutoring conversation datasets, and anchoring generated tutor responses in existing tutor utterances. The framework is demonstrated on a dataset of tutoring dialogues, including results for two LLM-based learner simulations, and the published GitHub[1] code.

Keywords: learner simulation, evaluation framework, tutoring dialogues

1. Introduction

Simulated learners serve diverse purposes in education (Koedinger et al., 2015): testing theories of learning and instructional approaches, automating authoring of tutoring tools, or acting as teachable agents through which students learn by teaching. Early work used cognitive models—e.g., Betty's Brain (Leelawong and Biswas, 2008) and SimStudent (Matsuda et al., 2007)—to simulate learner interactions in structured tutoring environments. More recently, LLM-based tutors interact with students through naturalistic conversations (e.g., Khan Academy's Khanmigo,[2] Google LearnLM[3]). Evaluating such tutors at scale requires correspondingly capable *conversational* learner simulations.

However, ensuring that such simulations are valid and useful for the various goals above is an open problem: simulating realistic student learning and conversational characteristics is complex because LLMs are trained as helpful assistants (Martynova et al., 2025). Previous work has assessed simulators on their ability to match real-world learner skill mastery over time (Maclellan et al., 2016) or reproduce learner conversational characteristics (Perczel et al., 2025). Scarlatos et al. (2025) introduce a consolidated set of metrics by matching a simulated learner response with the real student message given the conversation history. However, they only evaluate simulations on one turn and not at the dialogue level, and their work is closed-source, making reproduction and extensions to other datasets difficult.

1. https://github.com/RenaissancePhilanthropy/EvalConvoLearn
2. https://www.khanmigo.ai/
3. https://blog.google/outreach-initiatives/education/google-learnlm-gemini-generative-ai/

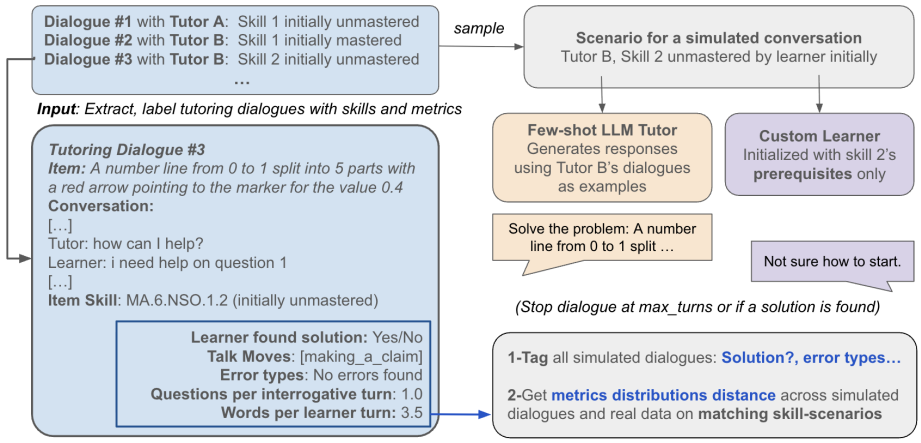

Figure 1: EvalConvoLearn learner evaluation steps.

We present **EvalConvoLearn**, an open-source framework evaluating the quality of conversational learner simulations. It can serve as a foundation to benchmark learner simulations across educational contexts and design goals for learners, including error types, learning trajectories, or social and interaction metrics (Yuan et al., 2026). EvalConvoLearn scores simulations' ability to produce realistic student behavior *distributions* across metrics extracted from tutoring conversation datasets.

2. The EvalConvoLearn Framework

EvalConvoLearn evaluates a learner simulation along two axes: (i) appropriate skill acquisition (Weitekamp et al., 2025) and (ii) conversational realism. The metrics are intentionally simple and target learner goal-aligned fidelity rather than surface realism, following Yuan et al. (2026). Given a tutoring conversation dataset with practice items and tagged skills (or items automatically tagged), EvalConvoLearn extracts *learning scenarios* defined as the product of a target skill and the learner's prior mastery of it (Figure 1). For each scenario, the framework runs simulated problem-solving conversations between the custom learner and EvalConvoLearn's tutor, then compares simulated and real metric distributions. The tutor is grounded at the individual tutor level with few-shot dialogue examples from other conversations in the dataset to simulate more realistic responses.

2.1. Learner Simulation Interface

EvalConvoLearn defines a minimal interface of four functions that any custom learner simulation must implement to be evaluated: The learner **initializes** its knowledge state according to a skill profile (mastered/unmastered skills or prerequisites). Then, it **generates a response** given a conversation history and its knowledge state, and is able to **update** its knowledge state from a conversation. Finally, it should be able to return whether it currently **masters** a given skill (used to match learners with mastery scenarios). If a custom learner cannot look up specific skills in its knowledge state, a default initialization loop runs

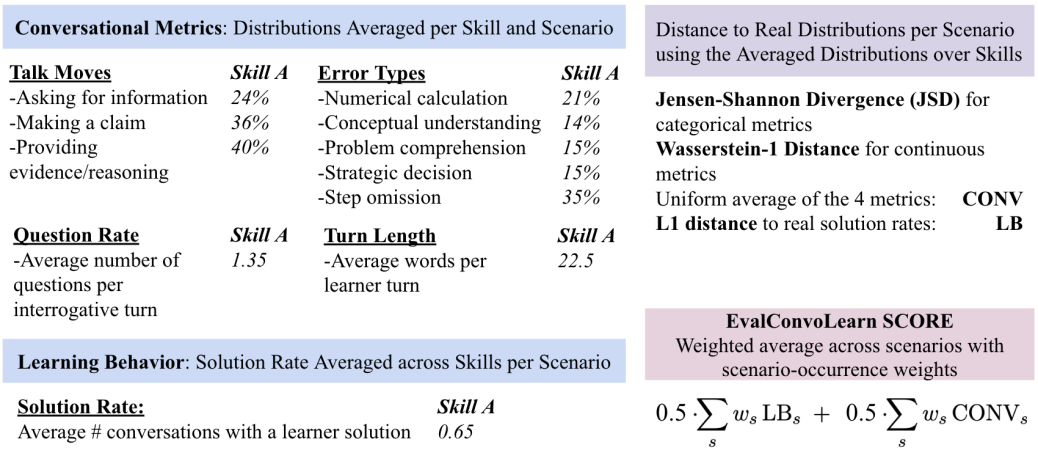

Figure 2: EvalConvoLearn metrics aggregation process

problem-solving conversations along the skill's prerequisites tree, while assessing progress at each step via skill-aligned items using the learner's *response generation* functionality.

2.2. Scenario-Based Metrics Evaluation

Following Scarlatos et al. (2025) applying Knowledge Tracing models (Corbett and Anderson, 1994) to tutoring dialogues at the turn level, we treat each learner turn as a skill practice attempt. We cap conversations at 7 turns to allow time for learning while preventing unrealistic lengths, following a heuristic of seven attempts to mastery (Koedinger et al., 2023). We strategically select a *skill pool* with sufficient aligned dialogues in the dataset that support all mastery scenarios. The learning-behavior score LB is the L1 distance between simulated and real outcome distributions for each scenario and skill, macro-averaged across the skill pool (Figure 2).

We evaluate conversational realism with four metrics: The presence of three talk moves (Suresh et al., 2022) and five error types (Qi et al., 2025) (LLM-labeled), number of questions in interrogative learner turns (Perczel et al., 2025), and turn length (Perczel et al., 2025). We manually evaluated LLM labels on 50 conversations and got a per-label Cohen's κ between .73 and 1.0 for talk moves (exact match .96) and from .79 to 1.0 for error types (exact match .88). We score each metric by computing the distance between its distribution in the simulated and real data within each scenario, using Jensen–Shannon divergence (JSD) or Wasserstein-1 distance (Figure 2). The resulting conversational score CONV_s averages the four distance metrics with equal weights. Users may add other conversational metrics and weigh them differently for their own context and evaluation goals.

The final EvalConvoLearn score (ECL) averages CONV_s and LB_s across scenarios s, accounting for scenario distributions from their occurrence in real dialogues (Figure 2). Both scores may be weighted according to their relative importance in one's learning context.

3. Evaluation on the Eedi Dataset

We apply EvalConvoLearn to the largest publicly available dataset of student–TA tutoring dialogues, from the Eedi learning platform (Zent et al., 2025). Since students proactively seek help on Eedi, we assume they have *not* yet mastered the target item skill but *have* mastered its prerequisites given Eedi's progressive curriculum. This assumption could be relaxed with random item assignments to learners, e.g. in simulated datasets. We retain conversations with $\geq$50% student utterances, conducted by tutors with $\geq$5 conversations (to enable few-shot prompting), yielding 66 conversations. We tag each practice item with a single skill from a subset of the B.E.S.T.[4] curriculum using GPT-4.1-mini for its cost/accuracy trade-off, and match simulations by keeping the 7 first conversation turns only.

We simulate conversations between a tutor using 3-shot prompting and two LLM-based learners using GPT-4.1-mini with custom knowledge implementations: The **Conversation Summaries** learner stores summaries of past conversations and uses them as context when generating a new response. The **Binary Skills** learner uses LLM-as-a-judge to tag items with skills and stores binary mastery outcomes after conversations. Table 1 reports Conversational (CONV), Learner Behavior (LB), and ECL scores for each configuration, aggregated across 3 skills with 8 conversations per skill. We used Claude's Sonnet-4.6 for tutor responses and metric evaluations.

Table 1: EvalConvoLearn results across 3 runs (mean ± standard error). Lower is closer to real distributions. The first row reports the CONV breakdown distances; the second row reports the aggregated scores (CONV, LB, ECL).

Setting	Talk Moves CONV	Errors LB	Questions ECL	Turn Length
Binary Skills	0.207 ± 0.000 **0.433 ± 0.011**	**0.385 ± 0.042** **0.556 ± 0.014**	0.415 ± 0.021 **0.494 ± 0.009**	0.724 ± 0.022
Conv. History	**0.206 ± 0.006** 0.450 ± 0.004	0.708 ± 0.025 0.556 ± 0.014	**0.226 ± 0.006** 0.503 ± 0.008	**0.658 ± 0.011**

4. Discussion and Future Work

The learners have similar *LB* scores as they both solve around 90% of items, which is 56 points more than real students, but differ slightly in conversational metrics while keeping high within-learner consistency. These results provide a signal to iterate on each learner's design by targeting their conversational or learning abilities, and suggest a large room for improvement for the models to produce realistic learner behavior. Future improvements should also tune the tutor model (See our ablation study of tutor models A), moving towards a dual learner-tutor optimization task.

4. https://www.fldoe.org/academics/standards/subject-areas/math-science/mathematics/

EvalConvoLearn is a work in progress. Critically, we are looking to expand the suite of scenarios and further validate the metrics with human ratings. When released, users can test the framework with new datasets, advanced learner and tutor simulations (e.g. fine-tuned or RL-trained models (Scarlatos et al., 2025)), and iterate on the learner quality signal to develop learner simulation benchmarks for various educational contexts.

Acknowledgments

The author thanks the Learning Engineering Virtual Institute, a program of Renaissance Philanthropy, for supporting the development of this work, as well as AJ Strauman-Scott for her contribution to the project.

References

Albert T. Corbett and John R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278, 1994.

Kenneth R. Koedinger, Noboru Matsuda, Christopher J. MacLellan, and Elizabeth A. McLaughlin. Methods for evaluating simulated learners: Examples from simstudent. In *AIED Workshops*, 2015.

Kenneth R. Koedinger, Paulo F. Carvalho, Ran Liu, and Elizabeth A. McLaughlin. An astonishing regularity in student learning rate. *Proceedings of the National Academy of Sciences*, 120(13):e2221311120, 2023.

Krittaya Leelawong and Gautam Biswas. Designing learning by teaching agents: The betty's brain system. *International journal of artificial intelligence in education*, 18(3):181–208, 2008.

Christopher J. Maclellan, Erik Harpstead, Rony Patel, and Kenneth R. Koedinger. The apprentice learner architecture: Closing the loop between learning theory and educational data. *International Educational Data Mining Society*, 2016.

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

Noboru Matsuda, William W. Cohen, Jonathan Sewall, Gustavo Lacerda, and Kenneth R. Koedinger. Predicting students' performance with simstudent: Learning cognitive skills from observation. *Frontiers in Artificial Intelligence and Applications*, 158:467, 2007.

Sania Nayab, Giulio Rossolini, Marco Simoni, Andrea Saracino, Giorgio Buttazzo, Nicolamaria Manes, and Fabrizio Giacomelli. Concise thoughts: Impact of output length on llm reasoning and cost. *arXiv preprint arXiv:2407.19825*, 2024.

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

Changyong Qi, Longwei Zheng, Anna He, Haoxin Xu, Linzhao Jia, Yuang Wei, Bingqian Jiang, and Xiaoqing Gu. Simulating student learning behaviors with llm-based role-playing agents: A data-driven and cognitively inspired framework. *Expert Systems with Applications*, page 130753, 2025.

Alexander Scarlatos, Ryan S. Baker, and Andrew Lan. Exploring knowledge tracing in tutor-student dialogues using llms. In *Proceedings of the 15th international learning analytics and knowledge conference*, pages 249–259, 2025.

Abhijit Suresh, Jennifer Jacobs, Charis Harty, Margaret Perkoff, James H. Martin, and Tamara Sumner. The talkmoves dataset: K-12 mathematics lesson transcripts annotated for teacher and student discursive moves. In *Proceedings of the thirteenth language resources and evaluation conference*, pages 4654–4662, 2022.

Daniel Weitekamp, Momin N. Siddiqui, and Christopher J. MacLellan. Tutorgym: A testbed for evaluating ai agents as tutors and students. In *International Conference on Artificial Intelligence in Education*, pages 361–376. Springer, 2025.

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

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

Appendix A. Tutor Context Ablation study

Because learner behavior depends on tutor messages, ECL scores can only be compared across simulations with similar tutor behavior distributions. We attempt to reduce the variance on the tutor's behavior and make it more realistic using few-shot prompting with individual tutor responses extracted from the dataset. To measure the impact of different tutors on the EvalConvoLearn scores, we conducted an ablation study by removing sample tutor responses from tutor prompts, with results in Table 2.

We observe that adding few-shot prompting improves final ECL scores for both learners, mainly through consistently better learner behavior. However, conversational scores see no change or get worse (in the case of the Binary Skills learner) when adding few-shot prompting. Adding human tutor examples seems to reduce the propensity of learners to solve problems in 7 conversation turns, which may be explained by shorter, more conversational, or less information-rich tutor responses. Simple analysis (N=450) corroborates this hypothesis by showing longer average tutor responses in the 0-shot setting (51 words) compared to the 3-shot setting (46 words), which is expected as few-shot prompting tends to correct the LLM propensity to be too verbose Nayab et al. (2024).

Additionally, the observed difference between Binary Skills learners' *CONV* scores comes specifically from more realistic error types in the 0-shot setting than the 3-shot

Table 2: Simulated learner scores with and without tutor prompting contexts, for Binary Skills (BS) and Conversation Summaries (CS) learners

Learner	**Eval**	**Tutor**	**CONV**	**LB**	**ECL**
BS, GPT-4.1-mini	Sonnet 4.6	3-shot	0.433 ± 0.011	**0.556 ± 0.014**	**0.494 ± 0.009**
BS, GPT-4.1-mini	Sonnet 4.6	0-shot	**0.406 ± 0.004**	0.611 ± 0.014	0.509 ± 0.007
CS, GPT-4.1-mini	Sonnet 4.6	3-shot	0.450 ± 0.004	**0.556 ± 0.014**	0.503 ± 0.008
CS, GPT-4.1-mini	Sonnet 4.6	0-shot	0.453 ± 0.035	0.653 ± 0.014	0.553 ± 0.025

setting (JSD of .29 versus .39, not reported here). Further work is needed to analyze the differences of tutor response features across few-shot settings, and refine the "error types" metric in particular by understanding how different tutor-learner conversational interactions can shift the score's distribution.

Appendix B. LLM Prompts

All code and prompts are available on the public GitHub code.