ADAPTIVE AGENTIC TUTORING WITH EVOLVING LEARNER PROFILES

Anonymous authors
Paper under double-blind review

ABSTRACT

Tutoring a learner through a question and choosing what they should practice next require different operations, but both depend on the same course material and evidence of that learner's difficulties. We present DeepTutor, which separates problem tutoring from practice generation while sharing course retrieval and an evolving learner profile revised from interaction traces. The tutoring pipeline investigates, solves with tools, and writes an evidence-backed explanation; the practice pipeline selects a learning target before generating and validating question–answer–explanation items. We introduce TutorBench to evaluate both outputs in multi-turn dialogue: 270 tasks across five disciplines pair source-grounded knowledge gaps with 90 benchmark learner specifications, and a first-person student simulator elicits tutoring and follow-up practice. Experiments on TutorBench show that DeepTutor attains the highest average scores for both tutoring and generated practice. Ablations show that course grounding helps explanations and practice stay faithful to source material, while the evolving learner profile helps tailor them to learners' difficulties. In a blind comparison with an agentic tutoring baseline, human raters and the LLM judge both favor DeepTutor across all ten evaluation dimensions. These results support coordinating distinct tutoring and practice pipelines through shared course evidence and an evolving learner profile.

1 INTRODUCTION

An adaptive tutor must respond to a learner's particular confusion while remaining faithful to the material being taught. Two students asking about the same derivative may need different help: one has not learned the chain rule, while the other knows it but misapplies it to a composite function. The explanation should address the difference, and the next practice item should test the point each student still finds difficult. This requires the tutor to connect what happens during an explanation with its later choice of practice (Létourneau et al., 2025; Chu et al., 2025).

Problem tutoring and practice generation pose different computational demands. An explanation must investigate the current question and make its reasoning accessible; practice generation must identify what to test and verify that a new question has a sound answer. Prior work has advanced guided feedback (Daheim et al., 2024; Macina et al., 2025; Dinucu-Jianu et al., 2025), question generation (Li & Zhang, 2024; Cheng et al., 2025; Yao et al., 2025), and learner modeling across interactions (Corbett & Anderson, 1995; Piech et al., 2015; Park et al., 2024). Bringing these capabilities together raises a further question: how should evidence gathered while tutoring shape a later practice request? A tutor could provide a clear explanation and generate a valid question, yet miss the learner's specific difficulty if the two tasks use disconnected accounts of the learner. Formative assessment calls for using observed difficulties to choose the next learning activity (Black & Wiliam, 1998; Shute, 2008).

DeepTutor addresses this coordination problem by separating task execution while sharing the context that guides decisions. Problem tutoring follows an investigate–solve–write pipeline, which can use recorded difficulties to choose sub-goals and explain the solution at an appropriate depth. Practice generation first selects a learning target, then generates and validates question–answer–explanation items. Both pipelines retrieve evidence from the same course sources and consult an evolving learner profile. After each interaction, a Profile Agent compares the new trace with earlier records and revises its hypotheses about the learner's difficulties (Figure 1). The updated profile can

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

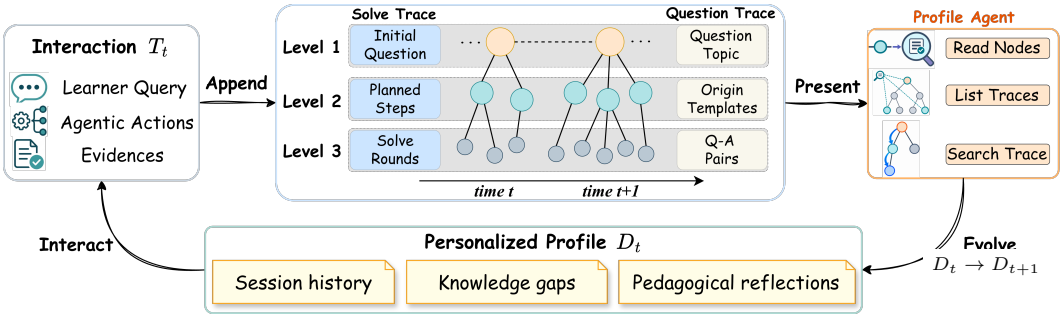


Figure 1: Learner-profile evolution across interactions. The Profile Agent examines each new trace alongside prior evidence to revise session history, knowledge-gap hypotheses, and pedagogical reflections. The resulting profile guides subsequent tutoring and practice in Figure 2, while the Trace Forest preserves the records supporting its updates.

then guide a subsequent tutoring response or practice request. Figure 2 shows how the two pipelines use this shared context.

The same distinction matters for evaluation. A correct explanation and a well-formed question do not establish that either one addresses this learner's difficulty. We therefore introduce **TutorBench**, which pairs interactive tasks with source-grounded learner specifications and knowledge gaps. A first-person student simulator elicits tutoring dialogue and follow-up practice; separate rubrics assess how well each output uses course evidence and responds to the specified learner. The evaluation measures produced explanations and practice items under simulated interactions rather than subsequent learning gains.

Our contributions are as follows:

- We propose separate pipelines for evidence-backed problem tutoring and learner-targeted practice generation, coordinated by shared course evidence and learner context.
- We maintain an evolving learner profile with a Profile Agent that revisits searchable interaction traces and supplies updated context to both pipelines.
- We introduce TutorBench to evaluate tutoring and practice against source-grounded learner difficulties, and assess the components with ablations and a blind human comparison.

2 Related Work

Personalized Agents. Agents can adapt to a user by summarizing dialogue or maintaining a user model (Tan et al., 2025; Li et al., 2025a; Cho et al., 2022), retrieving relevant material (Li et al., 2025b; Guo et al., 2025b), and organizing prior interactions at multiple levels of detail (Wang et al., 2025b; Packer et al., 2023; Zhang et al., 2026; 2025). In education, knowledge tracing estimates changing mastery from observed performance (Corbett & Anderson, 1995; Piech et al., 2015; Zhang et al., 2017; Ghosh et al., 2020). DeepTutor instead uses a revisable profile of observed difficulties and pedagogical reflections. A Profile Agent interprets each tutoring or practice interaction in light of earlier traces, then supplies the updated profile to both task pipelines. The traces let it revisit the evidence behind a learner hypothesis when a later interaction calls for a different explanation or practice target.

AI for Education. Educational agents have combined conversational tutoring with student models (Park et al., 2024; Wang et al., 2025a), diagnosis (Yao et al., 2026), recommendation (Li et al., 2025c), and pedagogical optimization (Dinucu-Jianu et al., 2025); open platforms expose these capabilities to learners (Hajji et al., 2026). MathTutorBench assesses open-ended tutoring behavior (Macina et al., 2025). DeepTutor focuses on how explanations and practice questions, despite their different execution steps, can draw on the same course evidence and evolving learner profile. TutorBench evaluates both outputs against source-grounded learner specifications. Appendix A gives additional context on personalized agents and educational systems.

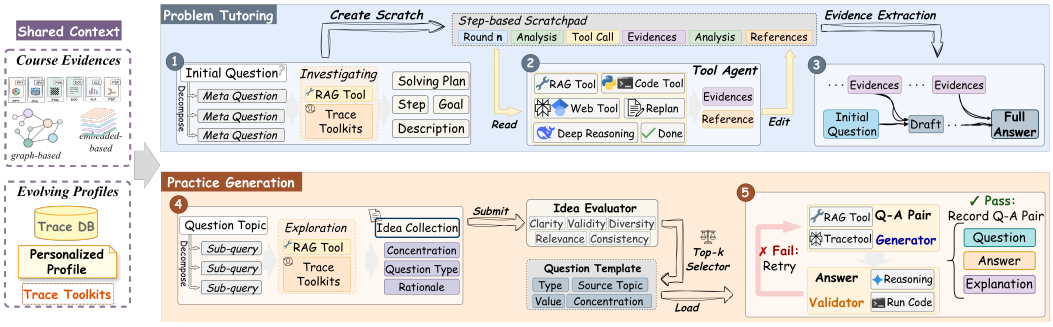


Figure 2: Two task-specific pipelines share course evidence and the current learner profile. Problem tutoring investigates, solves with tools, and writes an evidence-backed response; practice generation explores a learner gap, selects a template, and generates and validates a question–answer–explanation item. Both leave traces for the profile update in Figure 1.

3 METHODS

DeepTutor coordinates problem tutoring and practice generation through course knowledge $\mathcal{K}$ and an evolving learner profile $\mathcal{D} = (\mathcal{D}_s, \mathcal{D}_w, \mathcal{D}_r)$. Given a student question q, tutoring investigates, solves, and writes an explanation a (§3.2); given a topic τ, practice generation selects targets, generates questions, and validates them (§3.3). Figure 2 shows both pipelines. The Profile Agent revises $\mathcal{D}$ from interaction evidence, carrying the tutor's updated understanding of the learner into subsequent tasks (Figure 1; Appendix Algorithm 1).

3.1 HYBRID PERSONALIZATION ENGINE

Both pipelines use course evidence and an evolving account of the learner. Static Knowledge Grounding (SKG) retrieves course material for factual support. Under Evolving Learner Profiles (ELP), the Profile Agent revises the learner profile from interaction traces and supplies it to subsequent tutoring and practice requests.

3.1.1 STATIC KNOWLEDGE GROUNDING (SKG)

SKG is the course-evidence channel used by both task pipelines. Source documents are decomposed into content units, retaining modality-aware structure when available (Wang et al., 2024a). A knowledge graph $\mathcal{G}$ indexes structural and contextual relations among these units, while an embedding index $\mathcal{B}$ supports semantic retrieval (Guo et al., 2025b). Together they form a dual-index substrate over $\mathcal{K}$.

For a tutoring question or practice topic, graph traversal supplies related units and dense search supplies semantically similar ones. Reciprocal rank fusion (Cormack et al., 2009) combines these candidates; deduplication and a context budget yield the retrieved course context $\mathcal{C}_{\text{rag}}$ used for explanations and question construction.

3.1.2 EVOLVING LEARNER PROFILES (ELP)

The learner profile evolves as the tutor interprets evidence from successive interactions. Figure 1 illustrates this cycle: a completed interaction contributes a trace, the Profile Agent examines it alongside earlier records, and the revised profile guides subsequent tutoring and practice. The profile captures three complementary views: $\mathcal{D}_s$ summarizes session history, topics, and solving paths; $\mathcal{D}_w$ records trace-supported hypotheses about confusions and knowledge gaps; and $\mathcal{D}_r$ stores pedagogical reflections for future planning. Together, these views express the tutor's current interpretation of the learner's interactions.

The evidence supporting profile evolution is organized in a Trace Forest $\mathcal{F}$, with one searchable tree per completed problem-tutoring or practice-generation interaction. Each tree preserves three levels of detail: Level 1 stores the initial question or practice topic, Level 2 stores planned solving steps or question templates, and Level 3 records solving rounds or generated question–answer pairs,

including relevant evidence and validation outcomes. Dense embeddings support retrieval across the forest (Packer et al., 2023; Wang et al., 2025b; Zhang et al., 2026; 2025), while the hierarchy connects detailed records to the session and intermediate steps that produced them.

Evidence-based Profile Updates. When a new trace becomes available, the Profile Agent uses TraceToolkit to examine it in the context of prior interactions. SEARCHTRACE retrieves related records, LISTTRACES locates sessions by time, task type, or topic, and READNODES opens selected nodes with their ancestor paths. Comparing new and earlier evidence informs revisions to the session summary, gap hypotheses, and pedagogical reflections. For example, a follow-up question about an intermediate step can be interpreted alongside the earlier explanation and solving path. Revisiting that context helps the agent formulate a more specific gap hypothesis for future tutoring and practice. Generated practice items document what was offered to the learner; assessing performance on those items requires subsequent learner interaction. The profile thus remains an evidence-based, revisable interpretation rather than a direct measurement of mastery.

The updated $\mathcal{D}$ supplies learner context $\mathcal{C}_{\text{prof}}$ directly to both pipelines alongside course context $\mathcal{C}_{\text{rag}}$. It guides tutoring sub-goals and explanatory depth, as well as practice targets and difficulty. Task agents can also inspect supporting traces during investigation or idea exploration, connecting their decisions to the interaction evidence behind the current profile.

3.2 PERSONALIZED PROBLEM TUTORING

Building on tool-augmented reasoning (Yao et al., 2023; Wu et al., 2025; Li et al., 2025d; Chen et al., 2026), DeepTutor separates evidence acquisition, stepwise solving, and response writing. The learner profile guides both the solving plan and the explanation, following the principle of scaffolding from a learner's current understanding (Wood et al., 1976). Figure 2 and Appendix Algorithm 1 mark these stages as ①–③.

Stage ①: Personalized Investigation. The planner first runs a lightweight investigation pass before committing to a plan: it decomposes q into meta-questions and gathers evidence from $\mathcal{K}$ and relevant prior traces when needed, while using the latest learner profile to focus the investigation. Conditioning jointly on the resulting domain context $\mathcal{C}_{\text{rag}}$ and learner context $\mathcal{C}_{\text{prof}}$, it produces a tutoring plan $\mathcal{P} = \langle s_1, \ldots, s_K \rangle$ of concrete, annotated sub-goals. This investigation uses the current profile to choose targeted sub-goals (e.g., "review chain rule in trigonometry" rather than "review calculus").

Stage ②: Tool-Assisted Stepwise Solving. For each sub-goal s_k, the tool agent analyzes the target, chooses among retrieval, web search, code execution, deeper reasoning, replanning, and completion, then records notes (Yao et al., 2023), conditioned on both $\mathcal{C}_{\text{rag}}$ and $\mathcal{C}_{\text{prof}}$. Two mechanisms manage the growing reasoning context: self-notes distill each step's outcome into a concise takeaway that later steps reference in lieu of verbose intermediates, and hierarchical compression progressively summarizes completed sub-goals into compact digests, freeing context capacity for deeper reasoning on later ones. When the current plan proves inadequate, adaptive replanning revises remaining sub-goals while preserving completed work.

Stage ③: Evidence-based Iterative Writing. Rather than concatenating raw tool outputs, the writer extracts structured evidence from the step-based scratchpad, drafts the answer, and revises it through successive passes while reconciling potentially conflicting findings. The learner profile, supplied as $\mathcal{C}_{\text{prof}}$, guides explanatory depth and tone: the writer can expand intermediate steps around recorded difficulties and foreground key insights for a proficient learner (Vygotsky, 1978). The writer is instructed to cite retrieved support for externally grounded factual claims, drawing primarily on $\mathcal{K}$ and, when enabled, auxiliary web sources.

3.3 PERSONALIZED QUESTION GENERATION

Practice generation has two decisions: which learning target to probe and how to turn that target into a valid question. DeepTutor handles them in separate stages, marked ④–⑤ in Figure 2 and Appendix Algorithm 1. The first selects ideas using course and learner context; the second generates and checks question–answer–explanation items.

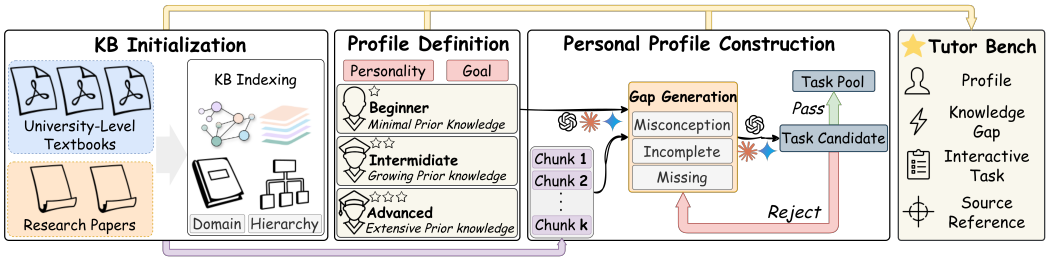


Figure 3: Construction pipeline of TutorBench. Four stages produce entries containing a benchmark learner specification, grounded knowledge gaps, an interactive task, and source references.

Stage ④: Personalized Idea Generation. The idea agent decomposes τ into sub-queries and examines course evidence and relevant prior traces. The current profile $\mathcal{C}_{\text{prof}}$ supplies recorded difficulties and pedagogical reflections, which guide target selection alongside $\mathcal{C}_{\text{rag}}$. Each candidate idea specifies a source topic, target concentration, question type, and personalized rationale. The idea evaluator checks clarity, validity, diversity, relevance, and consistency; a top-k selector turns the strongest ideas into templates $\{\mathcal{T}_i\}$ recording type, source topic, value, and concentration. The TutorBench evaluation instantiates five multiple-choice questions per session.

Stage ⑤: Critic-Guided Question–Answer–Explanation Generation. For each template $\mathcal{T}_i$, a **generator** produces a question, answer, and explanation triple (q_i, a_i, e_i), drawing on course evidence, relevant prior traces, and $\mathcal{C}_{\text{prof}}$ for grounding and learner calibration. A separate **validator** checks template alignment, factual correctness, and pedagogical soundness, using sandboxed code execution for computational questions. It evaluates the item without the generator's intermediate reasoning, making the check less dependent on the generator's own justification (Spiliopoulou et al., 2025). Items that fail the check receive diagnostic feedback and are regenerated for another validation pass.

3.4 Profile-Guided Coordination Across Tasks

Both pipelines can be invoked independently and draw on the latest learner profile. A chain-rule difficulty identified during tutoring can therefore guide a later practice request toward composite-function differentiation. The generated items then enter the session history, providing context for future planning. This coordination carries interaction evidence across tasks through the profile updates in Figure 1. Appendix B gives the execution algorithm and solver actions.

4 TutorBench

TutorBench comprises 270 tasks from 90 learner specifications and 30 knowledge bases across five disciplines. Each entry combines an interactive request, three page-grounded gaps, and source references (Figure 3). The benchmark specification initializes the simulated student; DeepTutor's evolving profile $\mathcal{D}$ is built separately from interaction traces.

Construction Pipeline. Textbooks and research papers are indexed into the dual-representation knowledge base (§3.1.1), with three learner specifications defined for each KB. Sampled pages supply misconceptions, incomplete understandings, and missing knowledge, each with an observable manifestation and correct understanding (Smith III et al., 1994; Shute, 2008). Rejection sampling retains three tasks per specification that align with the gaps and read naturally (Appendix Algorithm 2); human reviewers check factual correctness, learner plausibility, and gap–task consistency. Tasks span concept understanding, problem solving, application, and comparison. The specification and gap manifestations initialize the simulator, while correct understandings are hidden from evaluated systems and used only for judging. Appendix C gives the full procedure.

	Tutoring Quality						Practice Quality						
System	SF	PER	APP	VID	LD	*Avg*	FIT	GND	DIV	ANS	CC	*Avg*	**OQ**
Backbone-Matched Baselines													
Naive[a]	3.22	4.24	4.44	3.89	3.99	3.96	3.20	2.46	2.83	3.87	3.12	3.10	3.53
CoT[b]	3.34	4.22	4.47	3.82	4.01	3.97	3.18	2.50	2.79	3.81	3.04	3.06	3.52
Self-Refine[c]	3.28	4.28	**4.61**	3.94	4.14	4.05	3.17	2.49	2.88	3.88	2.97	3.08	3.57
ReAct[d]	3.35	4.37	4.40	3.70	3.96	3.96	3.16	2.47	2.75	3.94	3.07	3.08	3.52
Independent Agent Systems													
Claude Code[e]	2.84	4.54	4.56	4.46	4.51	4.18	**3.36**	2.84	**3.49**	**4.10**	3.12	3.38	3.78
Codex[f]	2.92	4.36	4.24	4.05	3.99	3.91	3.31	2.83	2.95	3.77	2.63	3.10	3.51
GenMentor[g]	2.86	4.56	4.46	4.14	4.22	4.05	3.28	2.61	2.90	3.70	2.50	3.00	3.52
DEEPTUTOR (ours)	**3.36**	**4.59**	4.56	**4.81**	**4.61**	**4.39**	3.35	**2.96**	3.44	3.98	**3.38**	**3.42**	**3.91**

Table 1: Interactive evaluation scores (1–5; higher is better). The two *Avg* columns summarize five tutoring and five practice dimensions; OQ is the unweighted mean of all ten dimensions. The highlighted DeepTutor row shares the backbone and task interface with the backbone-matched baselines; independent agent systems are shown as separate comparisons. **Bold** denotes the highest displayed value and underline the second highest, including displayed ties. Method and system references: [a]Brown et al. (2020) for direct prompting, [b]Wei et al. (2022), [c]Madaan et al. (2023), [d]Yao et al. (2023), [e]Anthropic (2025), [f]OpenAI (2025), [g]Wang et al. (2025a).

5 EXPERIMENTS

We evaluate tutoring and practice under a shared model and task interface, isolate SKG and ELP through ablations, and test sensitivity to rubric composition. A blind comparison with GenMentor examines agreement between human and LLM preferences, while a cross-domain analysis tests consistency across subjects. All TutorBench analyses use all 270 tasks. Appendix D.3.4 studies the solver scaffold without learner context.

5.1 TUTORING AND PRACTICE QUALITY

The adaptive diagnosis and guided remediation that unfold over multiple turns are often missed by traditional single-turn metrics. Building on rubric-based LLM evaluation in open-ended settings (Zheng et al., 2023; Macina et al., 2025) and simulated-student paradigms for tutoring assessment and training (Dinucu-Jianu et al., 2025; Scarlatos et al., 2026; Pan et al., 2025), we introduce a TutorBench protocol in which an LLM-based student simulator engages each tutoring system, and the resulting transcripts are scored against personalized rubrics.

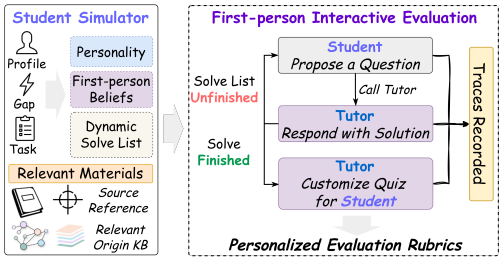


Figure 4: First-person TutorBench evaluation. A simulated student interacts with a tutor; personalized rubrics score the dialogue and generated practice.

By pre-transforming knowledge gaps into first-person belief statements, the simulator acts as the learner rather than narrating errors from an external perspective (Scarlatos et al., 2026).

Measures and protocol. TutorBench scores explanations and generated practice on ten 1–5 dimensions grounded in formative feedback, learner modeling, source-grounded generation, and educational assessment (Black & Wiliam, 1998; Shute, 2008; Corbett & Anderson, 1995; Anderson et al., 1995; Kurdi et al., 2020). The five tutoring-side dimensions are Source Faithfulness (SF), which checks attribution and consistency with course evidence (Maynez et al., 2020; Rashkin et al., 2023); Personalization (PER), which checks alignment with the learner's confusion (Vygotsky, 1978; Corbett & Anderson, 1995); Applicability (APP), which checks actionable guidance (Hattie & Timperley, 2007; Shute, 2008); Vividness (VID), which checks useful multiple representations (Ainsworth, 2006;

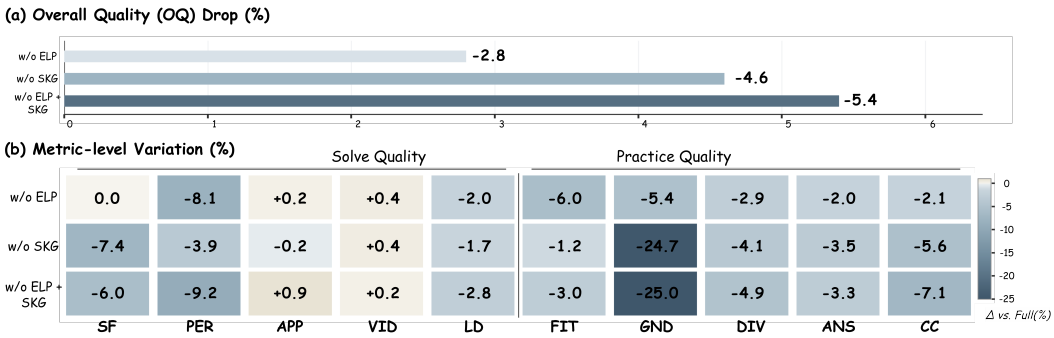


Figure 5: Ablation of Static Knowledge Grounding (SKG) and Evolving Learner Profiles (ELP). Panel (a) shows the overall quality (OQ) drop from removing each component, while panel (b) breaks down the per-metric change relative to the full system.

Mayer, 2003); and Logical Depth (LD), which checks explicit reasoning (Chi et al., 1989; Koedinger et al., 2012).

The five practice-side dimensions are Fitness (FIT) to diagnosed gaps and difficulty (Wood et al., 1976; Lord, 1980); Groundedness (GND) in course sources (Lewis et al., 2020; Gao et al., 2023); Diversity (DIV) of angle and cognitive demand (Anderson et al., 2001; Kurdi et al., 2020); Answer Quality (ANS) (Haladyna et al., 2002); and Cross Concept integration (CC) (Chi et al., 1981; Barnett & Ceci, 2002).

The protocol has three stages: offline entry generation, multi-turn student simulation, and independent judging of saved transcripts. Each transcript is scored three times by a fixed judge and the scores are averaged. The backbone-matched comparison evaluates the full 270-task set from 90 benchmark learner specifications and 30 knowledge bases: *Gemini-3-Flash* powers the student simulator, DeepTutor, and four prompting controls; *Claude Sonnet 4.6* judges their outputs at temperature zero. Figure 4 illustrates the student–tutor exchange and personalized scoring, while Appendix D.1 details the simulation and judging procedures.

Baselines. Table 1 compares DeepTutor with two groups of baselines. The backbone-matched baselines use the same source KB, Gemini-3-Flash backbone, and task interface as DeepTutor: Naive uses direct prompting (Brown et al., 2020), CoT adds chain-of-thought (Wei et al., 2022), Self-Refine adds a pedagogical review pass (Madaan et al., 2023), and ReAct uses a staged-prompting implementation inspired by the original method (Yao et al., 2023). The independent agent systems retain their own workflows: Claude Code (Anthropic, 2025), Codex (OpenAI, 2025), and the multi-agent tutoring system GenMentor (Wang et al., 2025a).

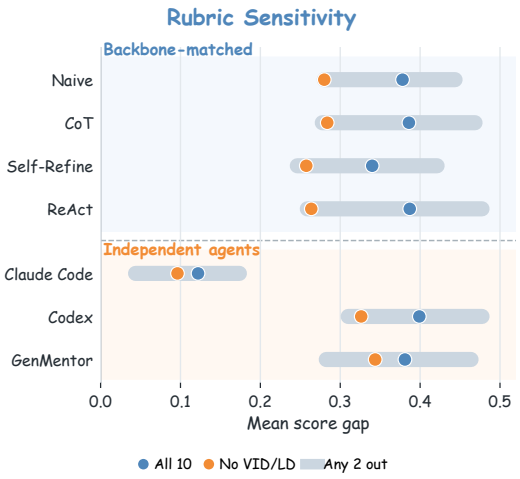


Figure 6: Rubric sensitivity of DeepTutor's score gaps. Bars span the 45 pairs of omitted metrics; dots show all ten metrics and the omission of VID and LD.

Results. DeepTutor reaches OQ 3.91, compared with 3.53 for direct prompting and 3.57 for Self-Refine. Against direct prompting, tutoring and practice averages rise from 3.96 to 4.39 and from 3.10 to 3.42. Personalization (4.59 versus 4.24), practice Groundedness (2.96 versus 2.46), and Cross Concept integration (3.38 versus 3.12) improve. Self-Refine leads on Applicability; Claude Code leads on Fitness, Diversity, and Answer Quality. Practice Groundedness remains low in absolute terms.

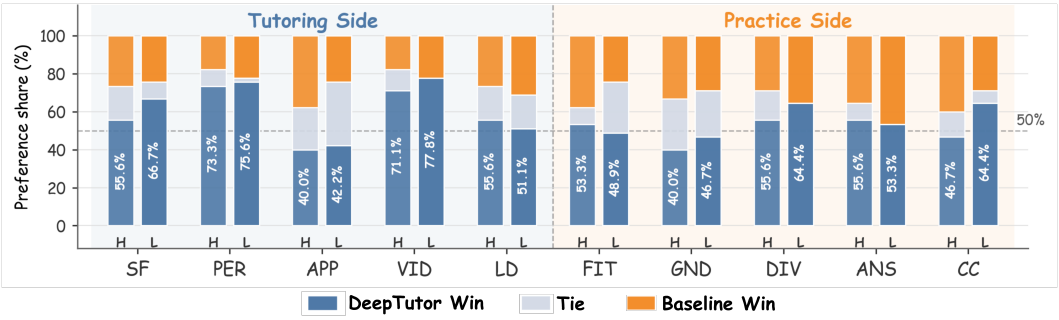


Figure 7: Human and LLM preferences between DeepTutor and GenMentor (Wang et al., 2025a) across ten rubric metrics on all 270 TutorBench tasks. H/L denote aggregated human and LLM-judge preferences; blue, gray, and orange show DeepTutor wins, ties, and GenMentor wins. Human disagreements are counted as ties.

5.2 Component Ablation and Rubric Sensitivity

We remove Static Knowledge Grounding (SKG), Evolving Learner Profiles (ELP), or both from the full system. The ELP ablation disables the tutor-maintained profile, its updates, and access to prior traces. Figure 5 shows different metric-level signatures. Without SKG, practice Groundedness drops most sharply, followed by Source Faithfulness and Cross Concept integration, consistent with the role of course evidence in both output types. Without ELP, the largest declines are in Personalization and Fitness, the two measures most closely tied to learner targeting. Removing both yields the largest OQ decline. Some metrics change little or even rise slightly under a single ablation, so the result is a pattern of differentiated contributions rather than uniform improvement from every component on every measure.

Rubric-composition sensitivity. Overall quality averages ten rubric dimensions, so a gain concentrated in a few dimensions could dominate the aggregate. We recompute each system gap after omitting Vividness and Logical Depth, then repeat the calculation for all 45 pairs of omitted dimensions. Vividness and Logical Depth account for 39–46% of DeepTutor's ten-metric score gap against the four backbone-matched baselines, yet omitting both leaves a 0.26–0.28 point advantage (Appendix Figure 9). Every eight-metric gap remains positive: the minimum is 0.25 points versus Self-Refine and 0.04 points versus Claude Code (Figure 6). Gaps to independent systems are descriptive because their workflows and backbones differ. This calculation tests rubric composition, not response length.

5.3 Human Preference Alignment

To examine whether automated preference trends agree with human judgments, we conduct a blind pairwise study on all 270 TutorBench tasks across the five domains. For each task, human raters compare anonymized outputs from DeepTutor and the tutoring baseline GenMentor (Wang et al., 2025a) generated under the same benchmark learner specification, source material, student-simulator protocol, and practice-generation request. The comparison packet includes the benchmark learner specification, task description, source material, tutoring transcript, and generated practice questions, allowing raters to judge both tutoring behavior and follow-up practice quality.

Two annotators compare outputs per session and metric; disagreements count as ties. Human and LLM preferences correlate across ten metric-level DeepTutor win rates (Pearson $r = 0.82$, $p = 0.0038$; Spearman $\rho = 0.83$, $p = 0.0027$). Both favor DeepTutor on every metric, most strongly for Personalization, Vividness, and Diversity; Applicability, Groundedness, and Cross Concept are closer. Figure 7 breaks down these outcomes by metric; Appendix D.2 details annotation and aggregation. These correlations measure alignment across dimensions rather than individual comparisons.

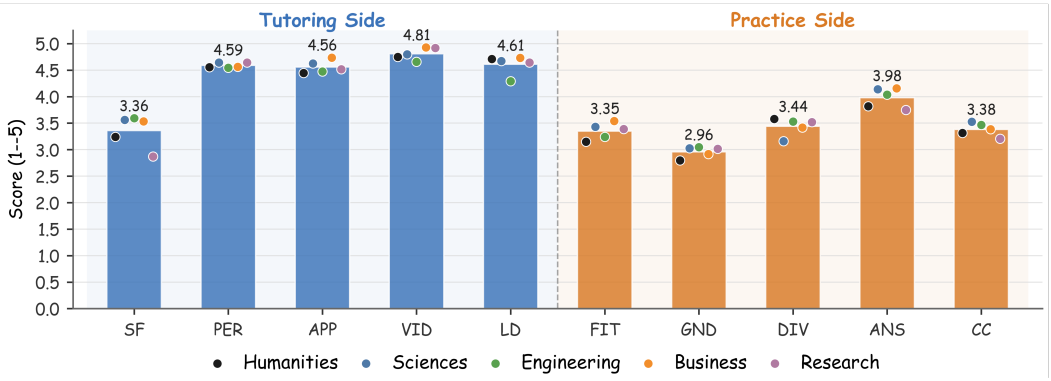


Figure 8: Cross-domain metric profiles for the first-person interactive simulation. Bars show the average score across five domains, while colored dots show domain-specific scores.

5.4 Cross-Domain Breakdown

Because the two pipelines draw on course sources from different subjects, we examine whether their quality is consistent across humanities, sciences, engineering, business, and frontier research. Figure 8 plots each metric by domain. DeepTutor's OQ spans 0.16 points, while the mean cross-domain span of an individual metric is 0.36 points. Source Faithfulness varies most among the tutoring metrics; for practice, Answer Quality remains comparatively high, but Groundedness is the lowest-scoring dimension in every domain. These profiles show that the combined score is relatively stable while source fidelity, especially for generated practice, remains sensitive to the domain and is a persistent weakness of the shared-evidence pipeline.

Additional experiments. Appendix D.3.4 evaluates the investigate–solve–write pipeline on general problem-solving benchmarks with SKG and ELP disabled. This supplementary experiment concerns the solver scaffold, separate from the learner-targeted evaluation above.

6 Conclusion

We presented DeepTutor, which coordinates separate tutoring and practice pipelines through shared course evidence and an evolving learner profile. TutorBench evaluates both outputs against source-grounded learner specifications in interactive simulation. On all 270 tasks, DeepTutor achieves higher rubric-assessed quality than the backbone-matched prompting baselines; ablations link SKG to source fidelity and ELP to personalization and practice fit. A blind comparison with GenMentor finds human preferences aligned with LLM preferences across rubric dimensions. These findings show how a revised profile can guide distinct tutoring tasks. Limitations and broader impacts appear in Appendix E.

AI Use Statement

As part of the methods reported in this paper, LLMs generated synthetic TutorBench tasks, powered the student simulator and tutoring systems, and scored their outputs using the stated rubrics; benchmark entries also underwent human review. Generative AI tools additionally assisted with methodological feedback, interpretation and presentation of results, creation and revision of schematic figures, and the structure, wording, and LaTeX preparation of this manuscript. The authors report no other AI uses in the disclosure categories of the ICLR 2027 policy and take responsibility for the final content, including AI-assisted text and figures.

Reproducibility Statement

The DeepTutor system and the TutorBench evaluation suite are available at https://anonymous.4open.science/r/DeepTutorAnonymous-77A6/. Appendix B outlines the execution algorithm and solver actions; Appendix C documents benchmark construction; and Appendix D describes the student simulator, scoring rubrics, human preference study, and general problem-solving setup.

References

Shaaron Ainsworth. DeFT: A conceptual framework for considering learning with multiple representations. *Learning and Instruction*, 16(3):183–198, 2006. doi: 10.1016/j.learninstruc.2006.03.001. URL https://www.sciencedirect.com/science/article/pii/S0959475206000259.

John R. Anderson, Albert T. Corbett, Kenneth R. Koedinger, and Ray Pelletier. Cognitive tutors: Lessons learned. *The Journal of the Learning Sciences*, 4(2):167–207, 1995. doi: 10.1207/s15327809jls0402_2. URL https://doi.org/10.1207/s15327809jls0402_2.

Lorin W. Anderson, David R. Krathwohl, Peter W. Airasian, Kathleen A. Cruikshank, Richard E. Mayer, Paul R. Pintrich, James Raths, and Merlin C. Wittrock. *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. Longman, 2001. URL https://www.loc.gov/item/00063423/.

Anthropic. Claude code documentation. https://docs.claude.com/en/docs/claude-code/overview, 2025. Accessed 2026-09-25.

Susan M. Barnett and Stephen J. Ceci. When and where do we apply what we learn? a taxonomy for far transfer. *Psychological Bulletin*, 128(4):612–637, 2002. doi: 10.1037/0033-2909.128.4.612. URL https://pubmed.ncbi.nlm.nih.gov/12081085/.

Paul Black and Dylan Wiliam. Assessment and classroom learning. *Assessment in Education: Principles, Policy & Practice*, 5(1):7–74, 1998. doi: 10.1080/0969595980050102. URL https://doi.org/10.1080/0969595980050102.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://papers.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

Guoxin Chen, Zile Qiao, Xuanzhong Chen, Donglei Yu, Haotian Xu, Xin Zhao, Ruihua Song, Wenbiao Yin, Huifeng Yin, Liwen Zhang, Kuan Li, Minpeng Liao, Yong Jiang, Pengjun Xie, Fei Huang, and Jingren Zhou. Iterresearch: Rethinking long-horizon agents with interaction scaling. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=qQ5MZ5Mx7p.

Qiguang Chen, Libo Qin, Jinhao Liu, Dengyun Peng, Jiannan Guan, Peng Wang, Mengkang Hu, Yuhang Zhou, Te Gao, and Wanxiang Che. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models. *arXiv preprint arXiv:2503.09567*, 2025. URL https://arxiv.org/abs/2503.09567.

Cheng Cheng, Zhenya Huang, Guanhao Zhao, Yuxiang Guo, Xin Lin, Jinze Wu, Xin Li, and Shijin Wang. From objectives to questions: A planning-based framework for educational mathematical question generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12836–12856, 2025. doi: 10.18653/v1/2025.acl-long.628. URL https://aclanthology.org/2025.acl-long.628/.

Michelene T. H. Chi, Paul J. Feltovich, and Robert Glaser. Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5(2):121–152, 1981. doi: 10.1207/s15516709cog0502_2. URL https://www.sciencedirect.com/science/article/pii/S0364021381800298.

Michelene T. H. Chi, Miriam Bassok, Matthew W. Lewis, Peter Reimann, and Robert Glaser. Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science*, 13(2):145–182, 1989. doi: 10.1207/s15516709cog1302_1. URL https://www.sciencedirect.com/science/article/pii/0364021389900025.

Itsugun Cho, Dongyang Wang, Ryota Takahashi, and Hiroaki Saito. A personalized dialogue generator with implicit user persona detection. In *Proceedings of the 29th International Conference on Computational Linguistics*, pp. 367–377, 2022. URL https://aclanthology.org/2022.coling-1.29/.

Zhendong Chu, Shen Wang, Jian Xie, Tinghui Zhu, Yibo Yan, Jingheng Ye, Aoxiao Zhong, Xuming Hu, Jing Liang, Philip S. Yu, and Qingsong Wen. LLM agents for education: Advances and applications. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 13782–13810, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.743. URL https://aclanthology.org/2025.findings-emnlp.743/.

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, 1995. doi: 10.1007/BF01099821. URL https://doi.org/10.1007/BF01099821.

Gordon V. Cormack, Charles L. A. Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 758–759, 2009. doi: 10.1145/1571941.1572114. URL https://doi.org/10.1145/1571941.1572114.

Nico Daheim, Jakub Macina, Manu Kapur, Iryna Gurevych, and Mrinmaya Sachan. Stepwise verification and remediation of student reasoning errors with large language model tutors. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8386–8411, 2024. doi: 10.18653/v1/2024.emnlp-main.478. URL https://aclanthology.org/2024.emnlp-main.478/.

David Dinucu-Jianu, Jakub Macina, Nico Daheim, Ido Hakimi, Iryna Gurevych, and Mrinmaya Sachan. From problem-solving to teaching problem-solving: Aligning LLMs with pedagogy using reinforcement learning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 272–292, 2025. doi: 10.18653/v1/2025.emnlp-main.15. URL https://aclanthology.org/2025.emnlp-main.15/.

Eduard Frankford, Clemens Sauerwein, Patrick Bassner, Stephan Krusche, and Ruth Breu. AI-tutoring in software engineering education. In *Proceedings of the 46th International Conference on Software Engineering: Software Engineering Education and Training*, pp. 309–319, 2024. doi: 10.1145/3639474.3640061. URL https://doi.org/10.1145/3639474.3640061.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6465–6488, 2023. doi: 10.18653/v1/2023.emnlp-main.398. URL https://aclanthology.org/2023.emnlp-main.398/.

Aritra Ghosh, Neil Heffernan, and Andrew S. Lan. Context-aware attentive knowledge tracing. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2330–2339, 2020. doi: 10.1145/3394486.3403282. URL https://doi.org/10.1145/3394486.3403282.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645:633–638, 2025a. doi: 10.1038/s41586-025-09422-z. URL https://doi.org/10.1038/s41586-025-09422-z.

Zirui Guo, Xubin Ren, Lingrui Xu, Jiahao Zhang, and Chao Huang. Rag-anything: All-in-one rag framework, 2025b. URL https://arxiv.org/abs/2510.12323.

Mohamed El Hajji, Tarek Ait Baha, Aicha Dakir, Hammou Fadili, and Youssef Es-Saady. Open tutorAI: An open-source platform for personalized and immersive learning with generative AI. *arXiv preprint arXiv:2602.07176*, 2026. URL https://arxiv.org/abs/2602.07176.

Thomas M. Haladyna, Steven M. Downing, and Michael C. Rodriguez. A review of multiple-choice item-writing guidelines for classroom assessment. *Applied Measurement in Education*, 15(3):309–333, 2002. doi: 10.1207/S15324818AME1503_5. URL https://www.tandfonline.com/doi/abs/10.1207/S15324818AME1503_5.

John Hattie and Helen Timperley. The power of feedback. *Review of Educational Research*, 77(1):81–112, 2007. doi: 10.3102/003465430298487. URL https://journals.sagepub.com/doi/10.3102/003465430298487.

Yuyang Hu, Shichun Liu, Yanwei Yue, Guibin Zhang, Boyang Liu, Fangyi Zhu, Jiahang Lin, Honglin Guo, Shihan Dou, Zhiheng Xi, Senjie Jin, Jiejun Tan, Yanbin Yin, Jiongnan Liu, Zeyu Zhang, Zhongxiang Sun, Yutao Zhu, Hao Sun, Boci Peng, Zhenrong Cheng, Xuanbo Fan, Jiaxin Guo, Xinlei Yu, Zhenhong Zhou, Zewen Hu, Jiahao Huo, Junhao Wang, Yuwei Niu, Yu Wang, Zhenfei Yin, Xiaobin Hu, Yue Liao, Qiankun Li, Kun Wang, Wangchunshu Zhou, Yixin Liu, Dawei Cheng, Qi Zhang, Tao Gui, Shirui Pan, Yan Zhang, Philip Torr, Zhicheng Dou, Ji-Rong Wen, Xuanjing Huang, Yu-Gang Jiang, and Shuicheng Yan. Memory in the age of AI agents. *arXiv preprint arXiv:2512.13564*, 2025. URL https://arxiv.org/abs/2512.13564.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=chfJJYC3iL.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. In *8th Annual Conference on Robot Learning*, 2024. URL https://openreview.net/forum?id=ZMnD6QZAE6.

Kenneth R. Koedinger, Albert T. Corbett, and Charles Perfetti. The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive Science*, 36(5):757–798, 2012. doi: 10.1111/j.1551-6709.2012.01245.x. URL https://pubmed.ncbi.nlm.nih.gov/22486653/.

Ghader Kurdi, Jared Leo, Bijan Parsia, Uli Sattler, and Salam Al-Emari. A systematic review of automatic question generation for educational purposes. *International Journal of Artificial Intelligence in Education*, 30(1):121–204, 2020. doi: 10.1007/s40593-019-00186-y. URL https://doi.org/10.1007/s40593-019-00186-y.

Angélique Létourneau, Marion Deslandes Martineau, Patrick Charland, Alexander-John Karran, Jared Boasen, and Pierre Majorique Léger. A systematic review of AI-driven intelligent tutoring systems (ITS) in K-12 education. *npj Science of Learning*, 10(1):29, 2025. doi: 10.1038/s41539-025-00320-7. URL https://www.nature.com/articles/s41539-025-00320-7.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020. URL https://papers.neurips.cc/paper_files/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Hao Li, Chenghao Yang, An Zhang, Yang Deng, Xiang Wang, and Tat-Seng Chua. Hello again! LLM-powered personalized agent for long-term dialogue. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5259–5276, 2025a. doi: 10.18653/v1/2025.naacl-long.272. URL https://aclanthology.org/2025.naacl-long.272/.

Kunze Li and Yu Zhang. Planning first, question second: An LLM-guided method for controllable question generation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 4715–4729, 2024. doi: 10.18653/v1/2024.findings-acl.280. URL https://aclanthology.org/2024.findings-acl.280/.

Xiaopeng Li, Pengyue Jia, Derong Xu, Yi Wen, Yingyi Zhang, Wenlin Zhang, Wanyu Wang, Yichao Wang, Zhaocheng Du, Xiangyang Li, Yong Liu, Huifeng Guo, Ruiming Tang, and Xiangyu Zhao. A survey of personalization: From RAG to agent. *arXiv preprint arXiv:2504.10147*, 2025b. URL https://arxiv.org/abs/2504.10147.

Zhaoxing Li, Jindi Wang, Wen Gu, Vahid Yazdanpanah, Lei Shi, Alexandra I. Cristea, Sarah Kiden, and Sebastian Stein. TutorLLM: Customizing learning recommendations with knowledge tracing and retrieval-augmented generation. In *Human-Computer Interaction – INTERACT 2025*, Lecture Notes in Computer Science, pp. 137–146. Springer Nature Switzerland, 2025c. doi: 10.1007/978-3-032-05005-2_8. URL https://link.springer.com/chapter/10.1007/978-3-032-05005-2_8. First online: 2025-09-09.

Zhuofeng Li, Haoxiang Zhang, Seungju Han, Sheng Liu, Jianwen Xie, Yu Zhang, Yejin Choi, James Zou, and Pan Lu. In-the-flow agentic system optimization for effective planning and tool use. In *NeurIPS 2025 Workshop on Efficient Reasoning*, 2025d. URL https://openreview.net/forum?id=XHkX4DbZUv.

Frederic M. Lord. *Applications of Item Response Theory to Practical Testing Problems*. Lawrence Erlbaum Associates, 1980. URL https://www.ets.org/research/policy_research_reports/publications/book/1980/jexj.html.

Jakub Macina, Nico Daheim, Ido Hakimi, Manu Kapur, Iryna Gurevych, and Mrinmaya Sachan. MathTutorBench: A benchmark for measuring open-ended pedagogical capabilities of LLM tutors. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 204–221, 2025. doi: 10.18653/v1/2025.emnlp-main.11. URL https://aclanthology.org/2025.emnlp-main.11/.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html.

Richard E. Mayer. The promise of multimedia learning: Using the same instructional design methods across different media. *Learning and Instruction*, 13(2):125–139, 2003. doi: 10.1016/S0959-4752(02)00016-6. URL https://www.sciencedirect.com/science/article/pii/S0959475202000166.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1906–1919, 2020. doi: 10.18653/v1/2020.acl-main.173. URL https://aclanthology.org/2020.acl-main.173/.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: a benchmark for general AI assistants. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=fibxvahvs3.

OpenAI. Introducing codex. https://openai.com/index/introducing-codex/, 2025. Accessed 2026-09-25.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*, 2023. URL https://arxiv.org/abs/2310.08560.

Sitong Pan, Robin Schmucker, Bernardo Garcia Bulle Bueno, Salome Aguilar Llanes, Fernanda Albo Alarcón, Hangxiao Zhu, Adam Teo, and Meng Xia. TutorUp: What if your students were simulated? training tutors to address engagement challenges in online learning. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pp. 1–18, 2025. doi: 10.1145/3706598.3713589. URL https://doi.org/10.1145/3706598.3713589.

Minju Park, Sojung Kim, Seunghyun Lee, Soonwoo Kwon, and Kyuseok Kim. Empowering personalized learning through a conversation-based tutoring system with student modeling. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pp. 1–10, 2024. doi: 10.1145/3613905.3651122. URL https://doi.org/10.1145/3613905.3651122.

Long Phan, Alice Gatti, Nathaniel Li, Adam Khoja, Ryan Kim, Richard Ren, Jason Hausenloy, Oliver Zhang, Mantas Mazeika, Dan Hendrycks, Ziwen Han, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, Michael Choi, Anish Agrawal, Arnav Chopra, Aakaash Nattanmai, Gordon McKellips, Anish Cheraku, Asim Suhail, Ethan Luo, Marvin Deng, Jason Luo, Ashley Zhang, Kavin Jindel, Jay Paek, Kasper Halevy, Allen Baranov, Michael Liu, Advaith Avadhanam, David Zhang, Vincent Cheng, Brad Ma, Evan Fu, Liam Do, Joshua Lass, Hubert Yang, Surya Sunkari, Vishruth Bharath, Violet Ai, James Leung, Rishit Agrawal, Alan Zhou, Kevin Chen, Tejas Kalpathi, Ziqi Xu, Gavin Wang, Tyler Xiao, Erik Maung, Sam Lee, Ryan Yang, Roy Yue, Ben Zhao, Julia Yoon, Xiangwan Sun, Aryan Singh, Clark Peng, Tyler Osbey, Taozhi Wang, Daryl Echeazu, Timothy Wu, Spandan Patel, Vidhi Kulkarni, Vijaykaarti Sundarapandiyan, Andrew Le, Zafir Nasim, Srikar Yalam, Ritesh Kasamsetty, Soham Samal, David Sun, Nihar Shah, Abhijeet Saha, Alex Zhang, Leon Nguyen, Laasya Nagumalli, Kaixin Wang, Aidan Wu, Anwith Telluri, Summer Yue, Alexandr Wang, Dmitry Dodonov, Tung Nguyen, Jaeho Lee, Daron Anderson, Mikhail Doroshenko, Alun Cennyth Stokes, Mobeen Mahmood, Oleksandr Pokutnyi, Oleg Iskra, Jessica P. Wang, John-Clark Levin, Mstyslav Kazakov, Fiona Feng, Steven Y. Feng, Haoran Zhao, Michael Yu, Varun Gangal, Chelsea Zou, Zihan Wang, Serguei Popov, Robert Gerbicz, Geoff Galgon, Johannes Schmitt, Will Yeadon, Yongki Lee, Scott Sauers, Alvaro Sanchez, Fabian Giska, Marc Roth, Søren Riis, Saiteja Utpala, Noah Burns, Gashaw M. Goshu, Mohinder Maheshbhai Naiya, Chidozie Agu, Zachary Giboney, Antrell Cheatom, Francesco Fournier-Facio, Sarah-Jane Crowson, Lennart Finke, Zerui Cheng, Jennifer Zampese, Ryan G. Hoerr, Mark Nandor, Hyunwoo Park, Tim Gehrunger, Jiaqi Cai, Ben McCarty, Alexis C. Garretson, Edwin Taylor, Damien Sileo, Qiuyu Ren, Usman Qazi, Lianghui Li, Jungbae Nam, John B. Wydallis, Pavel

Arkhipov, Jack Wei Lun Shi, Aras Bacho, Chris G. Willcocks, Hangrui Cao, Sumeet Motwani, Emily de Oliveira Santos, Johannes Veith, Edward Vendrow, Doru Cojoc, Kengo Zenitani, Joshua Robinson, Longke Tang, Yuqi Li, Joshua Vendrow, Natanael Wildner Fraga, Vladyslav Kuchkin, Andrey Pupasov Maksimov, Pierre Marion, Denis Efremov, Jayson Lynch, Kaiqu Liang, Aleksandar Mikov, Andrew Gritsevskiy, Julien Guillod, Gözdenur Demir, Dakotah Martinez, Ben Pageler, Kevin Zhou, Saeed Soori, Ori Press, Henry Tang, Paolo Rissone, Sean R. Green, Lina Brüssel, Moon Twayana, Aymeric Dieuleveut, Joseph Marvin Imperial, Ameya Prabhu, Jinzhou Yang, Nick Crispino, Arun Rao, Dimitri Zvonkine, Gabriel Loiseau, Mikhail Kalinin, Marco Lukas, Ciprian Manolescu, Nate Stambaugh, Subrata Mishra, Tad Hogg, Carlo Bosio, Brian P. Coppola, Julian Salazar, Jaehyeok Jin, Rafael Sayous, Stefan Ivanov, Philippe Schwaller, Shaipranesh Senthilkumar, Andres M. Bran, Andres Algaba, Kelsey Van den Houte, Lynn Van Der Sypt, Brecht Verbeken, David Noever, Alexei Kopylov, Benjamin Myklebust, Bikun Li, Lisa Schut, Evgenii Zheltonozhskii, Qiaochu Yuan, Derek Lim, Richard Stanley, Tong Yang, John Maar, Julian Wykowski, Mart Oller, Anmol Sahu, Cesare Giulio Ardito, Yuzheng Hu, Ariel Ghislain Kemogne Kamdoum, Alvin Jin, Tobias Garcia Vilchis, Yuexuan Zu, Martin Lackner, James Koppel, Gongbo Sun, Daniil S. Antonenko, Steffi Chern, Bingchen Zhao, Pierrot Arsene, Joseph M. Cavanagh, Daofeng Li, Jiawei Shen, Donato Crisostomi, Wenjin Zhang, Ali Dehghan, Sergey Ivanov, David Perrella, Nurdin Kaparov, Allen Zang, Ilia Sucholutsky, Arina Kharlamova, Daniil Orel, Vladislav Poritski, Shalev Ben-David, Zachary Berger, Parker Whitfill, Michael Foster, Daniel Munro, Linh Ho, Shankar Sivarajan, Dan Bar Hava, Aleksey Kuchkin, David Holmes, Alexandra Rodriguez-Romero, Frank Sommerhage, Anji Zhang, Richard Moat, Keith Schneider, Zakayo Kazibwe, Don Clarke, Dae Hyun Kim, Felipe Meneguitti Dias, Sara Fish, Veit Elser, Tobias Kreiman, Victor Efren Guadarrama Vilchis, Immo Klose, Ujjwala Anantheswaran, Adam Zweiger, Kaivalya Rawal, Jeffery Li, Jeremy Nguyen, Nicolas Daans, Haline Heidinger, Maksim Radionov, Václav Rozhoň, Vincent Ginis, Christian Stump, Niv Cohen, Rafał Poświata, Josef Tkadlec, Alan Goldfarb, Chenguang Wang, Piotr Padlewski, Stanislaw Barzowski, Kyle Montgomery, Ryan Stendall, Jamie Tucker-Foltz, Jack Stade, T. Ryan Rogers, Tom Goertzen, Declan Grabb, Abhishek Shukla, Alan Givré, John Arnold Ambay, Archan Sen, Center for AI Safety, Scale AI, and HLE Contributors Consortium. A benchmark of expert-level academic questions to assess ai capabilities. *Nature*, 649(8099):1139–1146, 2026. doi: 10.1038/s41586-025-09962-4. URL https://doi.org/10.1038/s41586-025-09962-4.

Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas Guibas, and Jascha Sohl-Dickstein. Deep knowledge tracing. In *Advances in Neural Information Processing Systems*, volume 28, pp. 505–513, 2015. URL https://papers.neurips.cc/paper/5654-deep-knowledge-tracing.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. Measuring attribution in natural language generation models. *Computational Linguistics*, 49(4):777–840, 2023. doi: 10.1162/coli_a_00486. URL https://direct.mit.edu/coli/article/49/4/777/116438/Measuring-Attribution-in-Natural-Language.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=Ti67584b98.

Alexander Scarlatos, Jaewook Lee, Simon Woodhead, and Andrew Lan. Simulated students in tutoring dialogues: Substance or illusion? *arXiv preprint arXiv:2601.04025*, 2026. URL https://arxiv.org/abs/2601.04025.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems*, 36:68539–68551, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html.

Valerie J. Shute. Focus on formative feedback. *Review of Educational Research*, 78(1):153–189, 2008. doi: 10.3102/0034654307313795. URL https://doi.org/10.3102/0034654307313795.

John P. Smith III, Andrea A. diSessa, and Jeremy Roschelle. Misconceptions reconceived: A constructivist analysis of knowledge in transition. *The Journal of the Learning Sciences*, 3(2):115–163, 1994. doi: 10.1207/s15327809jls0302_1. URL https://doi.org/10.1207/s15327809jls0302_1.

Evangelia Spiliopoulou, Riccardo Fogliato, Hanna Burnsky, Tamer Soliman, Jie Ma, Graham Horwood, and Miguel Ballesteros. Play favorites: A statistical method to measure self-bias in LLM-as-a-judge. *arXiv preprint arXiv:2508.06709*, 2025. URL https://arxiv.org/abs/2508.06709.

Zhen Tan, Jun Yan, I-Hung Hsu, Rujun Han, Zifeng Wang, Long Le, Yiwen Song, Yanfei Chen, Hamid Palangi, George Lee, Anand Rajan Iyer, Tianlong Chen, Huan Liu, Chen-Yu Lee, and Tomas Pfister. In prospect and

retrospect: Reflective memory management for long-term personalized dialogue agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8416–8439, 2025. doi: 10.18653/v1/2025.acl-long.413. URL https://aclanthology.org/2025.acl-long.413/.

Artificial Analysis Team. Artificial analysis long context reasoning benchmark(lcr), 2025. URL https://huggingface.co/datasets/ArtificialAnalysis/AA-LCR.

Lev S Vygotsky. *Mind in society: The development of higher psychological processes*, volume 86. Harvard university press, 1978. ISBN 9780674576292. URL https://www.hup.harvard.edu/books/9780674576292.

Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, Bo Zhang, Liqun Wei, Zhihao Sui, Wei Li, Botian Shi, Yu Qiao, Dahua Lin, and Conghui He. MinerU: An open-source solution for precise document content extraction. *arXiv preprint arXiv:2409.18839*, 2024a. URL https://arxiv.org/abs/2409.18839.

Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao. YOLOv9: Learning what you want to learn using programmable gradient information. In *Computer Vision – ECCV 2024*, Lecture Notes in Computer Science, pp. 1–21. Springer Nature Switzerland, 2024b. doi: 10.1007/978-3-031-72751-1_1. URL https://link.springer.com/chapter/10.1007/978-3-031-72751-1_1. First online: 2024-10-26.

Tianfu Wang, Yi Zhan, Jianxun Lian, Zhengyu Hu, Nicholas Jing Yuan, Qi Zhang, Xing Xie, and Hui Xiong. LLM-powered multi-agent framework for goal-oriented learning in intelligent tutoring system. In *Companion Proceedings of the ACM on Web Conference 2025*, pp. 510–519, 2025a. doi: 10.1145/3701716.3715244. URL https://doi.org/10.1145/3701716.3715244.

Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 63897–63911. PMLR, 2025b. URL https://proceedings.mlr.press/v267/wang25bx.html.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022. URL https://papers.neurips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

Colin White, Samuel Dooley, Manley Roberts, Arka Pal, Benjamin Feuer, Siddhartha Jain, Ravid Shwartz-Ziv, Neel Jain, Khalid Saifullah, Sreemanti Dey, Shubh-Agrawal, Sandeep Singh Sandha, Siddartha Venkat Naidu, Chinmay Hegde, Yann LeCun, Tom Goldstein, Willie Neiswanger, and Micah Goldblum. Livebench: A challenging, contamination-limited LLM benchmark. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=sKYHBTAxVa.

David Wood, Jerome S. Bruner, and Gail Ross. The role of tutoring in problem solving. *Journal of Child Psychology and Psychiatry*, 17(2):89–100, 1976. doi: 10.1111/j.1469-7610.1976.tb00381.x. URL https://doi.org/10.1111/j.1469-7610.1976.tb00381.x.

Junde Wu, Jiayuan Zhu, Yuyuan Liu, Min Xu, and Yueming Jin. Agentic reasoning: A streamlined framework for enhancing LLM reasoning with agentic tools. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 28489–28503, 2025. doi: 10.18653/v1/2025.acl-long.1383. URL https://aclanthology.org/2025.acl-long.1383/.

Fangzhou Yao, Sheng Chang, Weibo Gao, and Qi Liu. Conversational learning diagnosis via reasoning multi-turn interactive learning. *arXiv preprint arXiv:2603.03236*, 2026. URL https://arxiv.org/abs/2603.03236.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Zonghai Yao, Aditya Parashar, Huixue Zhou, Won Seok Jang, Feiyun Ouyang, Zhichao Yang, and Hong Yu. MCQG-SRefine: Multiple choice question generation and evaluation with iterative self-critique, correction, and comparison feedback. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 10728–10777, 2025. doi: 10.18653/v1/2025.naacl-long.538. URL https://aclanthology.org/2025.naacl-long.538/.

Guibin Zhang, Muxin Fu, Kun Wang, Guancheng Wan, Miao Yu, and Shuicheng YAN. G-memory: Tracing hierarchical memory for multi-agent systems. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=mmIAp3cVS0.

Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th International Conference on World Wide Web*, pp. 765–774, 2017. doi: 10.1145/3038912.3052580. URL https://doi.org/10.1145/3038912.3052580.

Ningning Zhang, Xingxing Yang, Zhizhong Tan, Weiping Deng, and Wenyong Wang. HiMem: Hierarchical long-term memory for LLM long-horizon agents. *arXiv preprint arXiv:2601.06377*, 2026. URL https://arxiv.org/abs/2601.06377.

Bingxi Zhao, Lin Geng Foo, Ping Hu, Christian Theobalt, Hossein Rahmani, and Jun Liu. LLM-based agentic reasoning frameworks: A survey from methods to scenarios, 2025. URL https://arxiv.org/abs/2508.17692.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html.

A Extended Related Work

This appendix expands the literature review in §2 on agentic tool use, personalization, learner modeling, and LLM-based educational agents.

A.1 Tool-Augmented LLM Agents

Equipping language models with external tools has progressed from *static* integration to *dynamic* reasoning. Toolformer (Schick et al., 2023) showed that LLMs can learn to insert API calls through self-supervised fine-tuning. ReAct (Yao et al., 2023) introduced a more flexible paradigm by interleaving chain-of-thought reasoning with environment-facing actions, significantly improving multi-step retrieval and decision-making. Subsequent work has expanded planning capabilities in various directions (Zhao et al., 2025; Wu et al., 2025; Li et al., 2025d), while long-horizon agents such as IterResearch (Chen et al., 2026) tackle coherent state maintenance across many reasoning steps. These methods motivate tool use within a task. DeepTutor also carries learner context between tutoring and practice requests, so later actions can use evidence from earlier interactions.

A.2 Personalized Agents and Learner Context

Personalized agents can carry user context across conversations through reflective summaries (Tan et al., 2025), accumulated user models (Li et al., 2025a), and persona inference (Cho et al., 2022). Retrieval-based approaches adapt evidence selection to user preferences (Li et al., 2025b), including across modalities (Guo et al., 2025b). Long-horizon agent systems organize prior experience as reusable procedures (Wang et al., 2025b), paged context (Packer et al., 2023), or hierarchical interaction records (Zhang et al., 2026; 2025). These works inform how an agent can retain and revisit evidence. In DeepTutor, searchable traces support the Profile Agent's revisions of learner difficulties and pedagogical reflections; the resulting profile guides both tutoring and practice generation.

A.3 Knowledge Tracing and Intelligent Tutoring Systems

The need for structured learner modeling has a long history in education. Classical *knowledge tracing* (KT) estimates per-skill mastery probabilities from interaction logs: Bayesian Knowledge Tracing (Corbett & Anderson, 1995) uses two-state hidden Markov models, Deep Knowledge Tracing (Piech et al., 2015) replaces hand-crafted features with recurrent networks, DKVMN (Zhang et al., 2017) separates concept keys from evolving mastery values in a key-value memory, and context-aware attention models (Ghosh et al., 2020) further improve interpretability through monotonic attention over learner histories. Early *Intelligent Tutoring Systems* (ITS) such as Cognitive Tutor (Anderson et al., 1995) were already providing step-level guidance through domain-specific production rules,

enabling fine-grained diagnosis of learner actions at the cost of substantial manual domain engineering. Recent work points to a growing shift toward LLM-based tutoring, which brings broad domain coverage and flexible natural-language interaction while often placing less emphasis on explicit, inspectable learner models of the kind common in classical ITS (Létourneau et al., 2025; Chu et al., 2025).

A.4 LLM-Based Educational Agents

Building on this shift, recent work has explored LLMs in dedicated tutoring roles along several complementary directions. Park et al. (2024) develop a conversational system with explicit student modeling, showing that tracking student state improves learning outcomes. Wang et al. (2025a) decompose the tutoring task across specialized agents for curriculum planning, dialogue, and assessment. Frankford et al. (2024) apply LLM tutoring to software engineering education, suggesting that richer student models can support higher-quality feedback. Open TutorAI (Hajji et al., 2026) provides an open-source platform with customizable avatars for multimodal interaction, while ParLD (Yao et al., 2026) introduces a preview–analyze–reason pipeline for conversational learning diagnosis via multi-agent collaboration. TutorLLM (Li et al., 2025c) couples knowledge tracing with RAG-based recommendation, and Dinucu-Jianu et al. (2025) align tutoring LLMs with pedagogical principles via reinforcement learning from simulated student interactions. Taken together, this literature now spans feedback generation, curriculum design, adaptive assessment, pedagogically aligned training, and open-ended evaluation (Chu et al., 2025; Macina et al., 2025). Our design contributes to this broader space by assigning different execution steps to problem tutoring and practice generation while making course evidence and trace-derived learner context available to both (§3.4).

B Implementation Details of DeepTutor System

B.1 Algorithm

Algorithm 1 gives the execution order of DeepTutor's shared-context update cycle. Here a *session* denotes one Solve or Generate invocation; a TutorBench evaluation session can include multiple dialogue turns followed by a practice request. Starting from an initial learner profile $\mathcal{D}^{(0)}$ and an empty trace forest $\mathcal{F}^{(0)}$, each new session uses the current profile as learner context $\mathcal{C}_{\text{prof}}$. The algorithm then branches according to the task type: problem tutoring uses investigation with course evidence and prior traces, planning, step-by-step tool use, and iterative answer writing, while question generation explores a topic before idea selection and critic-guided question, answer, and explanation construction. In both branches, the completed interaction is converted into a trace tree and used by the Profile Agent to revise the learner profile, so subsequent sessions can use both the updated profile and the interaction evidence supporting it. The BuildTrace arguments in the pseudocode identify the request and final output; trace construction also uses the intermediate records produced by that branch.

B.2 Tool Definitions

Table 2 summarizes the six actions available to the Solver Agent. The `done` and `replan` actions function as control signals rather than external tool calls.

C TutorBench Construction Details

This appendix supplements §4 with full construction details. Each benchmark *entry* combines a learner profile, a set of knowledge gaps, an interactive task, and the supporting source content. Below we describe the source PDF inventory (§C.1), the rejection-sampling procedure that produces high-quality tasks (§C.2), and the relationship between Trace Forest and benchmark-time simulation (§C.3).

Algorithm 1 Two tutoring paradigms with learner-profile updates

Require: learner profile $\mathcal{D}^{(0)}$; trace forest $\mathcal{F}^{(0)} = \emptyset$
for session $j = 1, 2, \ldots$ **do**
  $\mathcal{C}_{\text{prof}} \leftarrow \mathcal{D}^{(j-1)}$
  if task$_j$ $= (\textsc{Solve}, q)$ **then**
    $\mathcal{C}_{\text{rag}} \leftarrow \textsc{RagTool}(q)$
    $\mathcal{C}_{\text{trace}} \leftarrow \textsc{SearchTrace}(\mathcal{F}^{(j-1)}, q)$
    $\mathcal{E}_{\text{plan}} \leftarrow \textsc{Investigate}(q,\ \mathcal{C}_{\text{rag}},\ \mathcal{C}_{\text{trace}},\ \mathcal{C}_{\text{prof}})$ ▷ ①
    $\mathcal{P}{=}\langle s_1, \ldots, s_K\rangle \leftarrow \textsc{Plan}(q,\ \mathcal{E}_{\text{plan}},\ \mathcal{C}_{\text{prof}})$
    $\mathcal{S} \leftarrow \emptyset$ ▷ step scratchpad
    for $k = 1$ **to** K **do** ▷ ②
      while s_k not resolved **do**
        $(\alpha_t,\ \mathbf{s}_t) \leftarrow \textsc{ToolAgent}(q, \mathcal{P}, \mathcal{S}, \mathcal{C}_{\text{rag}}, \mathcal{C}_{\text{prof}})$
        $\mathcal{S} \leftarrow \mathcal{S} \cup \{\mathbf{s}_t\}$
        if $\alpha_t = \textsc{Replan}$ **then**
          $\mathcal{P} \leftarrow \textsc{Revise}(\mathcal{P}, \mathcal{S})$
        end if
      end while
      $\mathcal{S} \leftarrow \textsc{Compress}(\mathcal{S}, k)$
    end for
    $\mathcal{E}_{\text{write}} \leftarrow \textsc{ExtractEvidence}(\mathcal{S})$
    $a \leftarrow \textsc{IterativeWrite}(q,\ \mathcal{E}_{\text{write}},\ \mathcal{C}_{\text{prof}})$ ▷ ③
    $T_j \leftarrow \textsc{BuildTrace}(q, a)$
  else ▷ task$_j$ $= (\textsc{Generate}, \tau)$
    $\mathcal{C}_{\text{rag}} \leftarrow \textsc{RagTool}(\tau)$
    $\mathcal{C}_{\text{trace}} \leftarrow \textsc{SearchTrace}(\mathcal{F}^{(j-1)}, \tau)$
    $\mathcal{Q}_\tau \leftarrow \textsc{DecomposeTopic}(\tau)$
    $f_{\text{idea}} \leftarrow \emptyset$
    repeat
      $\mathcal{I} \leftarrow \textsc{IdeaAgent}(\mathcal{Q}_\tau,\ \mathcal{C}_{\text{rag}},\ \mathcal{C}_{\text{trace}},\ \mathcal{C}_{\text{prof}},\ f_{\text{idea}})$ ▷ ④
      $(\textit{continue},\ f_{\text{idea}},\ \mathcal{T}_{1..N}) \leftarrow \textsc{EvaluateAndSelect}(\mathcal{I},\ \tau,\ \mathcal{C}_{\text{prof}})$
    until $\neg$ *continue*
    for $i = 1$ **to** N **do** ▷ ⑤
      repeat
        $(q_i, a_i, e_i) \leftarrow \textsc{Generator}(\mathcal{T}_i,\ \mathcal{C}_{\text{rag}},\ \mathcal{C}_{\text{trace}},\ \mathcal{C}_{\text{prof}})$
        $(\textit{pass}, f_i) \leftarrow \textsc{Validator}(q_i, a_i, e_i, \mathcal{T}_i)$
        if $\neg$ *pass* **then** $\mathcal{T}_i \leftarrow \textsc{Revise}(\mathcal{T}_i, f_i)$
        end if
      until *pass*
    end for
    $T_j \leftarrow \textsc{BuildTrace}(\tau, \{(q_i, a_i, e_i)\}_{i=1}^{N})$
  end if
  $\mathcal{F}^{(j)} \leftarrow \mathcal{F}^{(j-1)} \cup \{T_j\}$
  $\mathcal{D}^{(j)} \leftarrow \textsc{ProfileAgent}(\mathcal{D}^{(j-1)}, \mathcal{F}^{(j)}, T_j)$
end for

C.1 PDF Source Inventory

TutorBench is built from 30 KBs derived from source PDFs. For textbook sources, each book is partitioned into three chapter-contiguous PDFs before KB construction; for research sources, each paper forms a single KB. The resulting PDF inventory is listed below.

Textbook-derived PDFs.

- All textbook PDFs are sourced from OpenStax.
- *Calculus Volume 2* (OpenStax, `https://openstax.org/details/books/calculus-volume-2/`): Chapters 1–2, Chapters 3–4, and Chapters 5–6.
- *Calculus Volume 3* (OpenStax, `https://openstax.org/details/books/calculus-volume-3/`): Chapters 1–2, Chapters 3–4, and Chapters 5–6.
- *Principles of Economics 3e* (OpenStax, `https://openstax.org/details/books/principles-economics-3e/`): Chapters 1–6, Chapters 7–12, and Chapters 13–18.

Tool	When to Use	Action Input
`rag_search`	Need information from the knowledge base (definitions, formulas, theorems, examples)	A precise search query string
`web_search`	Need external / up-to-date information not available in the KB	A search query string
`code_execute`	Need computation, numerical derivation, plotting, or data processing	A clear description of *what* to compute
`reason`	Need deep logical deduction, mathematical derivation, or synthesis of already-gathered evidence—no new external data needed	A description of the reasoning focus
`replan`	The original plan is fundamentally flawed (wrong premise, missing critical info); use *sparingly*	A brief reason explaining why replan is needed
`done`	Sufficient information is already available to resolve the current subgoal	Leave empty (`""`)

Table 2: Available Solver Agent actions. `done` and `replan` are control signals; `reason` is an internal reasoning action.

- *Foundations of Information Systems* (OpenStax, `https://openstax.org/details/books/foundations-information-systems`): Chapters 1–3, Chapters 4–6, and Chapters 7–9.
- *Introduction to Computer Science* (OpenStax, `https://openstax.org/details/books/introduction-computer-science`): Chapters 1–3, Chapters 4–6, and Chapters 7–9.
- *Introduction to Philosophy* (OpenStax, `https://openstax.org/details/books/introduction-philosophy`): Chapters 1–4, Chapters 5–8, and Chapters 9–12.
- *Introduction to Business* (OpenStax, `https://openstax.org/details/books/introduction-business`): Chapters 1–5, Chapters 6–10, and Chapters 11–15.
- *Writing Guide with Handbook* (OpenStax, `https://openstax.org/details/books/writing-guide`): Chapters 1–6, Chapters 7–12, and Chapters 13–18.

Paper-derived PDFs.

- *Memory in the Age of AI Agents* (Hu et al., 2025).
- *DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning* (Guo et al., 2025a).
- *LiveCodeBench: Holistic and Contamination Free Evaluation of Large Language Models for Code* (Jain et al., 2025).
- *OpenVLA: An Open-Source Vision-Language-Action Model* (Kim et al., 2024).
- *Towards Reasoning Era: A Survey of Long Chain-of-Thought for Reasoning Large Language Models* (Chen et al., 2025).
- *YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information* (Wang et al., 2024b).

C.2 Task Generation and Rejection Sampling

Given a profile p and its associated knowledge gaps G, the task generator creates interactive learning tasks that naturally expose one or more gaps during a tutoring conversation. Four task types are supported—*concept understanding* (30%), *problem solving* (30%), *application* (20%), and *comparison* (20%)—using fixed sampling weights. A batch rejection sampler then filters out tasks whose gaps are incoherent, misaligned with the task description, or conversationally unnatural / internally inconsistent, and regenerates until three tasks have been accepted. Algorithm 2 gives the full procedure.

The RejectSample oracle verifies three criteria:

1. **Gap coherence.** The gaps targeted by the task must be logically related and form a coherent learning topic; tasks requiring the student to jump between disconnected concepts are rejected.
2. **Task–gap fit.** The task description, initial student message, and success criteria must organically connect to the targeted gaps—pursuing the task should naturally expose those gaps without artificial steering.

Algorithm 2 Task Generation with Rejection Sampling

Require: Profile p, gaps G, knowledge scope scope, source pages S; target task count $T = 3$

accepted $\leftarrow \emptyset$
repeat
  $\mathcal{T}^{\text{cand}} \leftarrow$ GENERATETASKS(p, G, scope) ▷ batch LLM generation
  for each candidate task $t \in \mathcal{T}^{\text{cand}}$ **do**
    ok $\leftarrow$ REJECTSAMPLE(t, G, S) ▷ 3-criterion check (below)
    if ok **and** |accepted| $< T$ **then**
      accepted $\leftarrow$ accepted $\cup \{t\}$
    end if
  end for
until |accepted| $= T$
return accepted

3. **Conversational naturalness and consistency.** The resulting task should read like a plausible learner request for the given profile, rather than an artifact assembled solely to surface benchmark gaps. Items whose gaps contradict one another or conflict with the task description are rejected under this criterion.

The oracle is instructed to apply strict criteria: when in doubt, reject. This conservative stance accepts a higher regeneration rate in exchange for higher task quality, a trade-off justified by the one-time offline nature of benchmark construction. Each accepted task carries a structured record: task type, title, description, initial student message (matched to the profile's personality), targeted gap IDs, expected gap exposure trajectory, success criteria, and difficulty rating. Step 1 then packages the profile, selected gaps, accepted task, and the corresponding sampled `source_content` into one benchmark entry, which becomes the unit consumed by the simulation and evaluation pipeline.

C.3 Trace Forest and Runtime Profile Evolution

Benchmark entry files contain learner specifications, gaps, and tasks; they do not contain a tutor-maintained profile or Trace Forest. During Step 2 simulation, DEEPTUTOR builds interaction traces and uses them to revise the profile across sessions. Subsequent problem-tutoring and practice-generation requests can use the current profile and inspect relevant traces. The ELP ablation disables these profile updates and access to prior traces while keeping the benchmark entries unchanged. Thus the same entry objects are evaluated across systems. Within the DEEPTUTOR variants, ELP determines whether the tutor-maintained profile evolves across sessions.

D Evaluation Protocol

This section documents the interactive evaluation (§D.1), the human preference alignment study (§D.2), and the general problem-solving benchmarks (§D.3).

D.1 First-Person Interactive Evaluation Details

Figure 4 summarizes the full interactive evaluation loop used for TutorBench. Each evaluation begins from a benchmark entry containing a learner profile, diagnosed knowledge gaps, an interactive task, and relevant source materials. The student simulator turns these inputs into first-person beliefs and continues the dialogue in the learner's voice. Once its questions are addressed, it is encouraged to request practice before ending. Tutor responses are saved in the session transcript and later judged with personalized rubrics derived from the same learner state and task context.

D.1.1 Student Simulator

The student simulator is an LLM agent initialized from a TutorBench entry (profile + gaps + task). Each gap is pre-transformed into a first-person belief statement so the simulator *acts as* the student rather than narrating errors from an outside view. The system prompt below governs simulator behavior throughout the interaction; the template variables are filled from the entry at initialization time.

Listing 1: Student Simulator System Prompt

```
You are role-playing as a student seeking help from an AI tutor.
Stay in character throughout the entire conversation. Never break
character.

# Who You Are
{personality}

# Your Background
{education_background}

# Why You Are Here
{learning_purpose}

# What You Know
You are confident about these topics:
{known_well}

You have vague or partial understanding (you might get details wrong):
{partially_known}

You have NO knowledge of the following (never encountered):
{unknown}

# What You Believe (IMPORTANT -- these feel true to you)
{beliefs}

# Behavioral Rules
1. NEVER display knowledge listed as "unknown". If the tutor asks
   about something you don't know, say so honestly.
2. When the tutor teaches something new, try to rephrase it in your
   own words. It's okay to rephrase imperfectly.
3. If the tutor asks "do you understand?", be honest based on whether
   the explanation actually addressed your confusion.
4. Keep responses concise: 1-4 sentences typically.
5. Stay consistent with your personality throughout.
6. You may ask follow-up questions or request examples.
7. When you feel you've understood, PREFER asking for a practice
   problem before ending -- testing yourself helps solidify learning.

# Ending the conversation (ONLY when done)
Use [ACTION: task_complete] ONLY if ALL conditions are true:
  - You are explicitly done.
  - You have zero remaining questions.
  - You are NOT requesting anything else.
  - Your message is a natural closing/goodbye.

CRITICAL: If your message contains any question, uncertainty, or
request for more help, do NOT use [ACTION: task_complete].
```

First-person gap beliefs. The `{beliefs}` placeholder contains first-person formulations of the benchmark entry's knowledge gaps. The prompt asks the simulator to remain consistent with those beliefs and to request practice before ending when appropriate.

D.1.2 Evaluation Rubrics

TutorBench assesses tutoring and generated practice along ten dimensions, each scored on a 1–5 scale by an LLM judge. The ten metrics are organized into two groups—five **tutoring-side** metrics that evaluate every tutor turn during the dialogue, and five **practice-side** metrics that score the generated questions after each session. In addition, the evaluator records raw turn counts as a non-LLM structural statistic. The ten dimensions are reported individually. Overall quality (OQ) in Table 1 is their unweighted arithmetic mean; the tutoring-side and practice-side averages each summarize five dimensions. The dimensions operationalize constructs from source-grounded generation, learner

modeling and formative feedback, multiple representations for explanation, and educational question assessment (Maynez et al., 2020; Shute, 2008; Corbett & Anderson, 1995; Ainsworth, 2006; Kurdi et al., 2020).

Tutoring-Side Metrics (turn-level, 1–5 each). Each metric below is evaluated on every `student` → `tutor` turn pair. When practice questions are present, the final dialogue turn that emits them is excluded from tutoring-side scoring to keep the tutoring and question-generation phases separate.

- **Source Faithfulness (SF)** captures factual consistency with the source material provided in each benchmark entry, following work on faithful and attributable generation (Maynez et al., 2020; Rashkin et al., 2023). The judge examines four sub-dimensions: (a) *factual consistency*—whether tutor claims agree with the source rather than contradicting it; (b) *terminological consistency*—whether the tutor follows the source's definitions and framing; (c) *source grounding*—whether the response shows evidence of drawing on source-specific content, not just generic knowledge; and (d) *source attribution*—whether the tutor explicitly attributes information via citation markers or natural-language references. Pedagogical elaboration beyond the source—analogies, worked examples, motivating applications—is explicitly *not* penalized, as long as it does not distort the source. A score of 5 requires accuracy, consistent terminology, clear source grounding, and explicit attribution; a score of 1 indicates core claims that are wrong or that directly conflict with the source.
- **Personalization (PER)** reflects how specifically the tutor addresses the learner's current confusion while remaining aligned with the learner state developed over the session (Vygotsky, 1978; Corbett & Anderson, 1995). The judge checks whether the response is tailored to the student's profile, the misconceptions surfaced in prior turns, and the particular question just asked—rather than offering a generic textbook answer. A score of 5 requires strong, visible tailoring to this specific learner's misunderstanding and conversational trajectory; a score of 3 is partially tailored but still generic in key parts; a score of 1 indicates a fully generic response that ignores learner-specific context entirely.
- **Applicability (APP)** evaluates whether the guidance is concrete and immediately actionable for the student (Hattie & Timperley, 2007; Shute, 2008). High scores require the tutor to provide specific, executable content such as worked examples, step-by-step procedures, verifiable calculations, or diagnostic questions that let the student make immediate progress on their problem. A score of 5 means the student can directly act on the instruction; a score of 1 means the response is vague, abstract, or impossible to verify.
- **Vividness (VID)** rewards representationally rich, engaging presentation that goes beyond monolithic plain-text prose (Ainsworth, 2006; Mayer, 2003). The judge assesses modality diversity: effective use of structured lists, equations paired with intuitive explanations, comparison tables, code demonstrations, or other representational forms that genuinely aid understanding. A score of 5 indicates rich multimodal delivery whose format choices clearly enhance learning; a score of 1 indicates flat, undifferentiated text with no meaningful modality variation.
- **Logical Depth (LD)** assesses the presence of explicit multi-step causal or analytical reasoning, as opposed to bare assertions (Chi et al., 1989; Koedinger et al., 2012). The judge looks for coherent reasoning chains: does the tutor explain *why* a result holds, trace cause-and-effect relationships, or walk the student through intermediate steps? A score of 5 requires clear, well-structured multi-step reasoning; a score of 1 means the tutor provides conclusions or facts without any substantive justification.

Practice-Side Metrics (session-level, 1–5 each). After the main tutoring dialogue concludes, the tutor generates five practice multiple-choice questions (MCQs). Each question is evaluated independently across the following five dimensions:

- **Fitness (FIT)** measures alignment with the session-level diagnosed weaknesses at appropriate difficulty (Wood et al., 1976; Lord, 1980). The judge checks whether the question targets the knowledge gaps surfaced during the dialogue, matches the student's proficiency level, and is neither trivially easy nor unreasonably hard given the learner profile. A score of 5 means the question precisely targets the student's weak points at a suitable challenge level; a score of 1 means the question is irrelevant to the diagnosed gaps or grossly miscalibrated in difficulty.
- **Groundedness (GND)** requires factual anchoring in the source material, mirroring the structure of Source Faithfulness on the solve side (Lewis et al., 2020; Gao et al., 2023). The judge checks four

parallel sub-dimensions: factual consistency, terminological consistency, source grounding, and source attribution of the question stem, correct answer, and explanation. A score of 5 means the question content is fully consistent with and clearly derived from the source; a score of 1 means the question uses fabricated claims or definitions that conflict with the source.

- **Diversity (DIV)** rewards novelty in angle and cognitive demand (Anderson et al., 2001; Kurdi et al., 2020). The judge evaluates whether the question presents a distinct perspective, question format, or cognitive action (e.g., application, analysis, comparison) compared with typical rote recall items. A score of 5 reflects a creative, non-templated question that engages higher-order thinking; a score of 1 indicates a standard fill-in-the-blank or direct-recall item indistinguishable from routine exercises.
- **Answer Quality (ANS)** evaluates the correctness of the reference answer and the plausibility of distractors (Haladyna et al., 2002). The judge verifies that the correct option is unambiguously right, that the explanation is clear and consistent with the chosen answer, and that distractors are wrong yet plausible enough to be pedagogically informative. A score of 5 requires a flawless key, a lucid explanation, and well-crafted distractors; a score of 1 indicates an incorrect key, a contradictory explanation, or nonsensical options.
- **Cross Concept (CC)** tests whether the question meaningfully integrates multiple concepts surfaced throughout the session, rather than testing a single isolated fact (Chi et al., 1981; Barnett & Ceci, 2002). The judge checks whether the question requires the student to synthesize ideas from different parts of the material or to reason about relationships between concepts. A score of 5 means the question demands genuine cross-concept reasoning; a score of 1 means it tests only one narrow, isolated point.

Auxiliary statistics. Beyond the ten judge-scored metrics, the evaluator records raw turn counts (student turns, tutor turns, and paired turns) as a structural measure of dialogue length. These counts are reported directly and are not normalized to the 1–5 scale. Additionally, the evaluator tracks judge reliability statistics such as parse-failure rates per metric; these are used to monitor evaluation stability but are not part of the quality scores themselves.

D.1.3 Protocol Details

The full benchmark pipeline runs on all 270 TutorBench tasks in three stages. **Step 1** (entry generation): TutorBench entries are generated offline using the pipeline described in Appendix C. **Step 2** (multi-session simulation): for each KB/profile/backend tuple, the student simulator is initialized from the entry set and runs up to `max_turns` = 30 student turns per session in the default batch pipeline; transcripts are saved together with five practice questions produced at session end. **Step 3** (LLM-as-judge): each saved transcript is scored for source faithfulness, four teaching-quality dimensions, turn counts, and practice-question quality.

For the backbone-matched interactive evaluation, *Gemini-3-Flash* powers the student simulator, DeepTutor, and the four prompting controls, while *Claude Sonnet 4.6* serves as the fixed judge with temperature = 0.0. To mitigate judge variance, each transcript is scored three times and the scores are averaged.

The ten 1–5 dimensions are aggregated to system-level means before forming the unweighted OQ summary used in Table 1.

Rubric-composition sensitivity. We recompute the Table 1 score differences after omitting Vividness and Logical Depth from the unweighted mean. Figure 9 shows all seven baselines. The eight-metric differences remain positive for the four backbone-matched baselines (0.26–0.28 points) and for the three independent agent systems (0.10–0.34 points). This calculation uses the displayed two-decimal metric means and does not control response length.

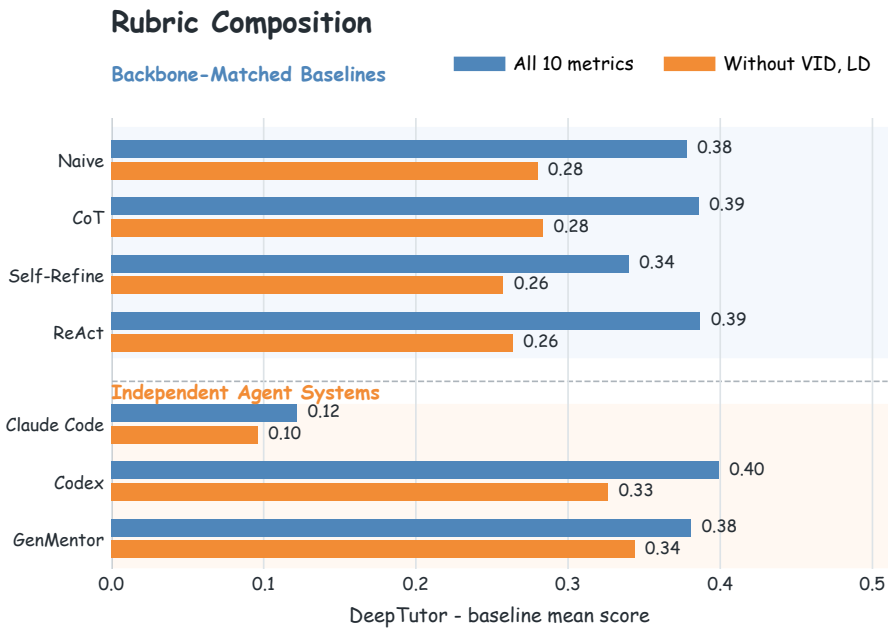


Figure 9: Rubric-composition sensitivity for all Table 1 baselines. Blue bars show DeepTutor-minus-baseline differences in the ten-metric mean; orange bars omit Vividness (VID) and Logical Depth (LD) and average the remaining eight metrics. Values are recomputed from the displayed metric means. Independent agent systems retain their own workflows and backbones.

D.2 Human Preference Alignment Study

The human alignment study in §5.3 is designed to test whether the LLM judge's relative system preferences agree with independent human judgments, rather than to measure downstream learning gains.

Evaluation set and materials. We use all 270 TutorBench tasks across the five domains in Figure 8. For each task, we collect the full DeepTutor and GenMentor outputs generated under the same benchmark entry: identical learner profile, task description, source material, student-simulator protocol, and practice-generation request. The comparison packet shown to annotators contains the learner profile, task description, source material, tutoring transcript, and generated practice questions for both systems.

Human annotation. We recruit ten student annotators with relevant academic training: six undergraduate students and four graduate students (two master's students and two Ph.D. students). Each domain is assigned to two annotators. Research sessions are evaluated by the two Ph.D. students; Engineering and Business each include one domain-related master's student; the remaining domain assignments are completed by domain-related undergraduates. Each packet is evaluated by both assigned annotators in a blind A/B interface. System names are hidden, and the order of A and B is randomly flipped per session to reduce order bias. Annotators are given the same ten metric definitions used by the LLM judge: Source Faithfulness, Personalization, Applicability, Vividness, Logical Depth, Fitness, Groundedness, Diversity, Answer Quality, and Cross Concept. For each metric, the annotator selects one of three labels: A is better, B is better, or the two systems are indistinguishable. This pairwise design avoids the scale-calibration problem of absolute 1–5 human scoring while directly testing the system preference implied by the automatic judge.

Aggregation and LLM comparison. Human labels are first mapped back to system identities and then aggregated for each session–metric pair; when the two annotators do not select the same system preference, the aggregate label is treated as a tie. For the LLM side, we use the corresponding

LLM-judge preference label for the same session and metric. Figure 7 reports, for each metric, the share of comparisons where humans prefer DEEPTUTOR, mark a tie, or prefer GenMentor, alongside the corresponding LLM-judge preference shares.

Alignment statistics. The correlation statistics reported in Figure 7 and §5.3 are computed over the ten metric-level DEEPTUTOR win-rate pairs, rather than over all individual session–metric labels. Pearson $r = 0.82$ ($p = 0.0038$) and Spearman $\rho = 0.83$ ($p = 0.0027$) therefore test whether the rubric dimensions most favored by human raters are also the dimensions most favored by the LLM judge. At the same metric-summary level, the directional winner agrees on all ten metrics: both human raters and the LLM judge assign the largest preference share to DEEPTUTOR for every metric.

D.3 GENERAL PROBLEM-SOLVING SETUP

D.3.1 BENCHMARK DESCRIPTIONS

Table 3 summarizes the five public benchmarks used in the additional problem-solving experiment reported in Appendix D.3.4.

Benchmark	Category	Size	Subset	Format
HLE (Phan et al., 2026)	STEM reasoning	∼2,500	500 (text+mm)	open-ended
GPQA-Diamond (Rein et al., 2024)	STEM reasoning	198	all 198	4-choice MCQ
LiveBench (White et al., 2025)	reasoning	∼200	reasoning cat.	mixed
GAIA (Mialon et al., 2024)	agentic	∼165	validation (L1–L3)	open-ended
AA-LCR (Team, 2025)	long-context	100	all 100	open-ended

Table 3: Public benchmarks used in the general problem-solving evaluation. "Size" refers to the total available test items; "Subset" is the number selected for evaluation. mm = multimodal (text + images).

D.3.2 EVALUATION PROTOCOL

Correctness is assessed via a two-stage pipeline. **Stage 1 (Extract)**: an LLM extracts the model's final answer from its free-form output (model: Claude Sonnet 4.6, temperature = 0.0, max tokens = 16384). **Stage 2 (Judge)**: a second LLM call compares the extracted answer to the ground truth using benchmark-specific rubrics (same model and settings as Stage 1 for consistency). Benchmark-specific rubrics cover exact match for GAIA, option matching or text equivalence for GPQA and HLE, and textual equivalence checks for open-ended benchmarks such as LiveBench and AA-LCR.

D.3.3 PIPELINE CONFIGURATION

Table 4 lists the tool configuration and concurrency settings for the DEEPTUTOR pipeline on each benchmark. This is the solver-only configuration used in Appendix D.3.4: the Hybrid Personalization Engine is disabled, including both Static Knowledge Grounding (SKG) and Evolving Learner Profiles (ELP). Thus no course knowledge base, learner profile, or trace-forest retrieval is active in the public benchmark runs.

Benchmark	Tools Enabled	Max Tokens	Concurrency
HLE	code_execute, reason	16384	30
GPQA-D	code_execute, reason	16384	50
LiveBench	code_execute, reason	16384	10
GAIA	code_execute, reason, web_search	16384	20
AA-LCR	code_execute, reason	16384	25

Table 4: Per-benchmark solver-only pipeline configuration. `done` and `replan` are always available.

D.3.4 SOLVER-ONLY RESULTS

With SKG and ELP disabled, the investigate–solve–write scaffold is evaluated on HLE (Phan et al., 2026), GPQA-Diamond (Rein et al., 2024), LiveBench (White et al., 2025), GAIA (Mialon et al., 2024), and AA-LCR (Team, 2025). Table 5 reports the original backbone comparisons.

	STEM Reasoning			General Agent (GAIA)†				Long Ctx	
Pass@1	HLE*	GPQA-D**	LiveBench***	L1	L2	L3	Overall	AA-LCR	*Avg* Δ
Gemini-3-Flash	19.40	81.31	70.00	43.40	40.70	15.38	37.58	63.00	**+29.24%**
+DEEPTUTOR	**30.80**	**84.85**	**96.00**	**56.60**	**50.00**	**23.08**	**47.88**	**74.67**	
Sonnet-4.5	8.40	72.22	64.33	37.74	30.23	7.69	29.09	53.33	**+32.03%**
+DEEPTUTOR	**14.60**	**73.23**	**82.00**	**50.94**	**48.84**	**23.08**	**45.45**	**54.00**	
Qwen-3.5-Plus	16.80	**88.38**	69.00	44.23	35.71	13.64	33.94	66.00	**+25.69%**
+DEEPTUTOR	**24.20**	87.88	**93.00**	**50.94**	**53.49**	**32.00**	**49.09**	**69.67**	
GPT-5-Mini	16.46	**80.81**	71.00	32.08	30.23	7.69	27.27	68.67	**+28.51%**
+DEEPTUTOR	**21.20**	80.30	**93.00**	**54.72**	**52.33**	**26.92**	**49.09**	**71.00**	
Minimax-M2.5	14.00	82.83	59.30	35.29	22.78	12.00	23.64	66.00	**+31.50%**
+DEEPTUTOR	**19.40**	**83.33**	**73.00**	**52.94**	**44.30**	**32.00**	**42.42**	**76.40**	

Table 5: Results of general agentic problem-solving **Pass@1 scores**. Each item yields one answer; repeated judgments, when used, are averaged. Denominators use the saved outputs scored in each run. *We use a fixed 500-question subset. **GPQA-Diamond. ***We use the LiveBench reasoning subset. †GAIA scores are reported *per difficulty level (L1–L3)*. *Avg* Δ is defined below.

For each backbone model m, we summarize performance by averaging the relative gain over its base model across these benchmark groups:

$$\text{Avg}\,\Delta_m = \frac{1}{|\mathcal{B}|}\sum_{b\in\mathcal{B}} \frac{s_{m,b}^{\text{DEEPTUTOR}} - s_{m,b}^{\text{base}}}{s_{m,b}^{\text{base}}} \times 100\%, \tag{1}$$

where $\mathcal{B}$ denotes the benchmark groups listed above. GAIA L1–L3 scores are reported only as diagnostic breakdowns.

As shown in Table 5, the solver-only pipeline improves all five backbone families, with average relative gains ranging from 25.69% to 32.03%. With SKG and ELP disabled, these results assess the investigate–solve–write scaffold in general problem solving; they do not evaluate learner personalization.

E LIMITATIONS

TutorBench measures the quality of explanations and generated practice under simulated interactions and rubric-based assessment. It does not measure learning gains in human students or the accuracy of the tutor-maintained profile as a model of a student's true knowledge. Generated questions are assessed as outputs; students' later performance on them is outside the current protocol. The blind human study evaluates preferences over produced materials, with the reported correlation computed across rubric dimensions rather than individual judgments.

The SKG and ELP ablations remove complete components. They show how the assessed outputs change but do not isolate the effect of profile updating or trace organization relative to a simpler way of maintaining learner context. The multi-stage pipelines also add model and tool calls, and the reported quality comparisons are not cost matched. These distinctions matter when extending the system to longer courses or real classrooms, where learner outcomes, profile fidelity, and inference cost must be measured directly.

Broader impacts. Personalized tutoring systems may broaden access to adaptive educational support, especially when high-quality human tutoring is scarce. At the same time, any LLM-based tutor can inherit general LLM risks such as hallucinated facts, overconfident explanations, or subtle reasoning errors. Real-world use should therefore treat generated tutoring content as assistive rather than authoritative, preserve source-grounded verification, and encourage students and educators to check important claims against trusted course materials or human instructors.