REGULATUTOR: A THEORY-INFORMED FRAMEWORK FOR LONG-TERM PERSONALIZED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-term personalized tutoring requires language models to translate a learner's evolving history into appropriate instructional support. Existing LLM-based tutoring systems use memory and retrieval to personalize responses, but the pedagogical rationale linking historical evidence to teaching strategies is not always explicit. We introduce *RegulaTutor*, a theory-informed framework for long-term personalized tutoring that connects historical evidence, learner interpretation, and learner-facing instruction. Guided by self-regulated learning (SRL) research, its tutoring harness operationalizes a cognitive, affective, metacognitive, and motivational (CAMM) framework for deriving evidence-supported learner interpretations and teaching policies. The resulting policies retain links to their supporting evidence and are updated as new observations arrive. This staged workflow also supports systematic refinement through *PairRoute*, an offline procedure that uses intermediate execution records to distinguish harness, model, and mixed failures. *PairRoute* proposes and evaluates targeted harness revisions, then constructs reviewed stage-level supervision for confirmed model failures remaining under the selected harness. We focus this refinement on historical evidence acquisition, which underpins personalized instruction and allows outputs to be checked against learning records and reference answers. RegulaTutor improves historical-evidence accuracy, learner diagnosis, and judged teaching quality on average across evaluated model configurations. PairRoute results show an endpoint improvement in standalone-model historical-evidence accuracy over the refinement trajectory. Together, these results highlight *RegulaTutor*'s potential to support lifelong personalized education through instructional policies that evolve with learner histories and targeted improvements to the tutoring system.

1 INTRODUCTION

Large language models (LLMs) are increasingly being developed into tutors that explain concepts, provide feedback, and personalize instruction (Zhao et al., 2026; Perczel et al., 2025). Yet learning unfolds across repeated attempts and interactions: a learner's difficulties may persist, recur, or change as understanding develops. Long-term personalized tutoring must therefore relate the current task to accumulated learner history and adapt as new observations arrive. However, recent evaluations highlight the difficulty of translating long-term learning histories into accurate diagnosis and appropriate instructional decisions (Li et al., 2026a).

Existing LLM-based tutoring systems use memory, retrieval, and learner profiles to support history-based personalization (Zhao et al., 2026; Wu et al., 2025). However, retrieving relevant records does not by itself specify how those records should guide teaching (Cohn et al., 2026; Gures et al., 2026). Repeated errors, for example, identify a difficulty worth investigating, but do not by themselves determine whether the tutor should reteach a concept, scaffold a procedure, or prompt the learner to inspect their own reasoning. Choosing an appropriate response therefore requires interpreting such evidence in terms of the learner's knowledge and learning processes, and aligning instructional support with that interpretation (Gures et al., 2026). This motivates an explicit, theory-informed connection from historical evidence to learner interpretation and instructional action.

We introduce *RegulaTutor*, a theory-informed framework for long-term personalized tutoring that connects historical evidence, learner interpretation, and learner-facing instruction (Figure 1). Guided

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.

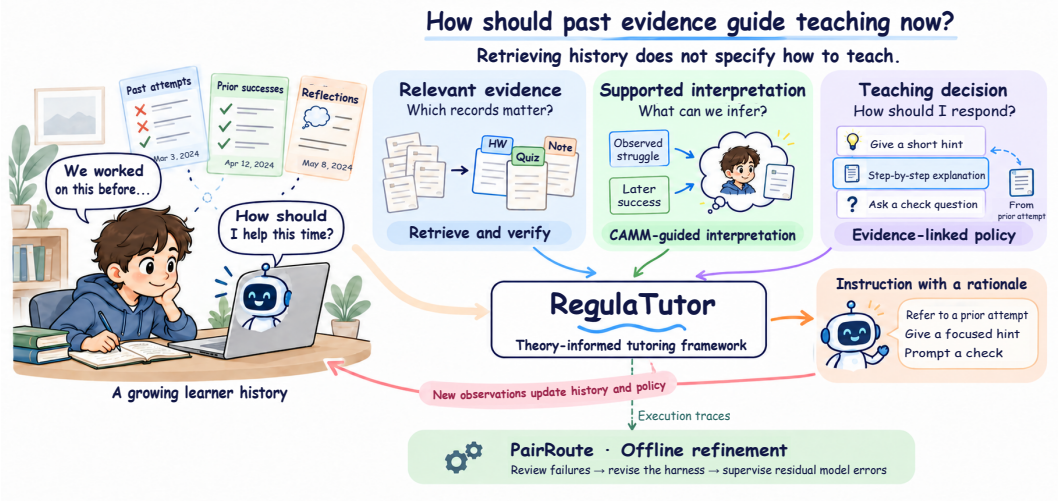


Figure 1: **RegulaTutor connecting learner history to instructional decisions.** The learner–tutor setting (left) motivates three questions (top): which records are relevant, what interpretations they support, and how those interpretations should guide teaching. RegulaTutor makes these connections explicit through CAMM-guided learner interpretation and evidence-linked teaching policies, with learner history and policies updated as new observations arrive. PairRoute (bottom) uses execution traces to guide offline harness revisions and construct reviewed stage-level supervision for residual model errors.

by self-regulated learning (SRL) research, its tutoring harness draws on cognitive, affective, metacognitive, and motivational (CAMM) processes to derive evidence-supported learner interpretations (Molenaar et al., 2023; Azevedo et al., 2022). These interpretations inform learner-specific teaching policies that specify the support mode, explanation depth, references to prior learning episodes, and metacognitive or motivational support. The harness translates these policies into instructional directives while retaining links to the supporting records.

Across interactions, the harness maintains learner history and updates these policies as new observations become available. Its shared system prompt also encourages learner goal setting, self-checking, and reflection when appropriate. Together, these mechanisms provide an inspectable path from a learner's accumulated history to the tutor's instructional choices.

Making these intermediate stages explicit also supports improvement of the tutoring system itself. An incorrect final response alone does not reveal whether the failure comes from the harness's access to or representation of evidence, the model's execution of a feasible stage, or both. RegulaTutor records intermediate model inputs and outputs so that failures can be examined at the stages where they occur. To guide targeted improvements, the framework includes *PairRoute*, an offline refinement procedure building on trajectory-informed harness optimization and model–harness co-evolution (Lee et al., 2026; Chen et al., 2026a;b). PairRoute uses these execution records to diagnose harness, model, and mixed failures, evaluate targeted harness revisions, and construct reviewed stage-level supervision for model failures that remain under the selected harness.

We focus this refinement study on historical evidence acquisition, which supplies the factual basis for personalized instruction and provides targets that can be checked against learning records and reference answers (Li et al., 2026a). RegulaTutor thus supports adaptation at two timescales: new learner observations inform interpretations and teaching policies during tutoring, while execution failures guide offline refinement of the harness and model.

We evaluate *RegulaTutor* on LONGTUTOR, a benchmark for tutoring over long learning histories (Li et al., 2026a). Fixed-backbone comparisons and component analyses examine the contribution of the harness and its instructional mechanisms. Across the evaluated model configurations, RegulaTutor improves historical evidence acquisition and learner diagnosis on average. Our focused refinement study uses Qwen3.5-4B to examine successive model checkpoints under different har-

ness versions. PairRoute's development-set trajectory shows gains in evidence processing under the selected harness; crossed model–harness evaluations further characterize how model updates interact with harness revisions.

Our contributions are threefold:

- We introduce *RegulaTutor*, a theory-informed tutoring framework that connects historical evidence, learner interpretations, and instructional directives, while updating learner history and tutoring policies across interactions.
- We develop *PairRoute*, an offline refinement procedure within *RegulaTutor* that uses intermediate execution records to diagnose failures, guide harness revisions, and construct reviewed stage-level supervision for model failures remaining under the selected harness.
- We evaluate *RegulaTutor* on long-history tutoring, examining evidence acquisition, diagnosis, and teaching. A focused study with Qwen3.5-4B further examines how harness revisions and model updates affect tutoring performance.

2 METHOD

RegulaTutor connects historical evidence to learner interpretations and instructional decisions through cognitive, affective, metacognitive, and motivational (CAMM) processes. We further develop *PairRoute* to use execution traces for harness revision and stage-level model supervision. Figure 2 summarizes both workflows. We first describe the tutoring harness, then introduce *PairRoute*, which uses its execution records to guide harness revisions and stage-level model supervision.

2.1 FROM LEARNER HISTORY TO INSTRUCTIONAL DECISIONS

At interaction i with learner u, RegulaTutor uses the current task $q_{u,i}$ and history $\mathcal{L}_{u,i}$ to derive a learner interpretation, update tutoring policy $P_{u,i}$, and generate a response. Model weights and shared harness rules remain unchanged during tutoring; learner history and policy evolve across interactions.

Learner history. RegulaTutor accumulates observations, dialogue, and policy state during ongoing sessions, or accepts a self-contained history for retrospective tasks. From this history, it computes learner statistics $S_{u,i}$ and a bounded retrieval view $\mathcal{B}_{u,i}$, separating accumulated learner information from the context supplied to individual model calls.

Historical evidence acquisition. The Evidence Resolution Loop (ERL) identifies the information needed for $q_{u,i}$. For questions about a particular attempt, it retrieves focused records; for questions spanning multiple records, mapper calls select observations from history chunks for consolidation. Resolver calls combine the evidence into answers with source identifiers. Citation checks and bounded retrieval expansion address unresolved information needs. We write

$$(E_{u,i}, A_{u,i}) = \mathrm{ERL}_{M,H}(\mathcal{B}_{u,i}, q_{u,i}), \tag{1}$$

where M and H denote the language model and tutoring harness, respectively, $E_{u,i}$ preserves source evidence, and $A_{u,i}$ contains per-need answers and support status. These outputs provide the evidence for the subsequent learner interpretation.

CAMM-guided learner interpretation. We draw on self-regulated learning (SRL) and its Metacognitive and Affective Model (MASRL) to connect observations with task-specific support (Zimmerman, 2002; Efklides, 2011), using CAMM as the organizing framework (Molenaar et al., 2023). RegulaTutor constructs the current learner interpretation $C_{u,i}$ from admitted evidence and statistics of the corresponding source events. In the task-verified configuration, this evidence consists of records cited by ERL.. Each tentative learner claim retains its supporting evidence, uncertainty, instructional implications, and unsupported inferences to avoid.

Evidence-linked teaching policies. Our guidelines map supported interpretations to policy proposals $\mathcal{A}_{u,i}$ for support mode, explanation depth, history references, and metacognitive or motivational support (Table 6 in Appendix A.2). For example, repeated errors on a concept support an

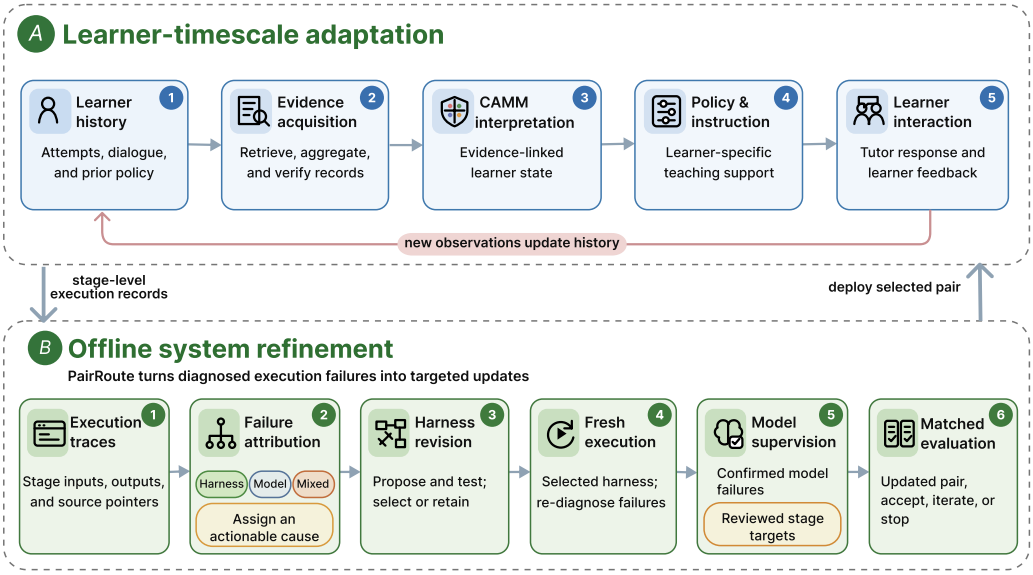


Figure 2: RegulaTutor adapts at two timescales. During tutoring, accumulated learner history supports evidence acquisition, CAMM-guided interpretation, and learner-specific instructional policies that update with new interactions. Offline, PairRoute attributes execution failures to the harness, model, or both and routes them to targeted harness revision or reviewed model supervision. The updated model–harness pair is evaluated before the next round or deployment.

interpretation of local difficulty, leading to stepwise explanation and a focused check grounded in a prior attempt. These errors do not by themselves establish low motivation. RegulaTutor updates the previous policy by screening proposals for evidence support and confidence, penalizing unnecessary switching, and resolving competing assignments to the same field. It retains policy provenance and re-examines learner interpretations as new evidence arrives. Appendix A.2 gives the update rules and representation details.

Instructional directives and response generation. To translate the interpretation and policy into learner-facing instruction, RegulaTutor combines $C_{u,i}$ and $P_{u,i}$ into instructional directives $D_{u,i}$. It then constructs the input for the final response-generation call:

$$\begin{aligned} x_{u,i}^{\text{final}} &= \text{Prompt}(q_{u,i}, S_{u,i}, E_{u,i}, A_{u,i}, P_{u,i}, D_{u,i}, \mathcal{D}_{u,i}), \\ y_{u,i} &\sim M(\cdot \mid x_{u,i}^{\text{final}}). \end{aligned} \tag{2}$$

Recent dialogue $\mathcal{D}_{u,i}$ preserves speaker roles to support follow-up responses. A shared system prompt encourages task-appropriate goal setting, self-checking, and reflection. RegulaTutor validates output structure and records intermediate execution, providing the traces used for offline refinement.

2.2 PairRoute: Trajectory-Guided Offline System Refinement

PairRoute uses RegulaTutor's execution records to guide updates targeting historical evidence acquisition. At offline round t, it operates on model-harness pair (M_t, H_t) through two reviewing-agent roles: Router1 proposes and evaluates harness revisions; Router2 reviews residual failures and prepares model supervision under the selected harness. Selecting the harness before preparing supervision allows Router2 to assess failures against the inputs and requirements of the selected harness (Chen et al., 2026a;b).

Trajectory-based failure diagnosis. Both router agents examine the task, reference evaluation, and actual stage inputs and outputs. A harness diagnosis requires evidence of defective access, rep-

resentation, or call requirements; a model diagnosis requires sufficient visible support and a feasible local task that the model fails to execute. Mixed cases involve both. The routers separate source or evaluation problems from execution failures and defer unresolved cases. Software validates call identities and evidence pointers supporting each judgment (Appendix B.1).

Router1: harness revision. Router1 uses training discovery trajectories to propose bounded revisions addressing harness and mixed failures. It also examines successes to specify behavior to preserve, then evaluates candidates with M_t fixed using static checks, targeted controls, and full-task probes. These evaluations inform selection of H_{t+1} under the round's recorded protocol, which may retain H_t.

Router2: supervision preparation. Router2 examines execution records under (M_t, H_{t+1}) and diagnoses failures again, because harness changes can alter the evidence available to a call and invalidate earlier failure assignments. For confirmed model failures, it localizes eligible calls, obtains teacher targets, and reviews them against the actual input, evidence, and output requirements. Only supported corrections enter training; other cases remain outside the correction set.

Stage-level training and update evaluation. Each stage-level training example consists of a recorded model-call input and a reviewed target. We combine reviewed corrections with reviewed correct current-model outputs, including correct stages within failed tasks, and apply completion-only supervised fine-tuning to obtain M_{t+1}. Under the selected harness H_{t+1}, we compare M_{t+1} with its parent model M_t on evidence acquisition, diagnosis, and teaching. Appendix B.2 details training, and Appendix B.3 records the round-level update decisions.

3 EXPERIMENTS

We use LongTutor to evaluate how RegulaTutor acquires historical evidence, diagnoses learner difficulties, and supports instruction from long learner histories. We first compare each backbone with and without RegulaTutor, then examine the contributions of its components, and finally study PairRoute through successive model checkpoints and their crossed evaluations with harness versions.

3.1 EXPERIMENTAL SETUP

Benchmarks. We evaluate on LONGTUTOR's 1,000-case human-annotated split, which tests historical evidence acquisition, learner diagnosis, and teaching using 199-record learner histories (Li et al., 2026a). Appendix C specifies the metrics and judges.

Implementation details. We instantiate RegulaTutor with Qwen3.5-4B, DeepSeek-v4-pro, Gemini-3.7-flash, GPT-5.6-sol, Kimi-k3, and DeepSeek-v4.1-flash. All fixed-backbone comparisons use the final harness H_1 selected through PairRoute-guided refinement (Table 1).

PairRoute starts from the initial Qwen3.5-4B model M_0 and harness H_0. Three rounds of supervised refinement on reviewed stage-level trajectories yield model checkpoints M_1–M_3; the selected harness H_1 is retained in later rounds, so $H_2 = H_3 = H_1$. Full model identities and decoding settings are given in Appendix C. Data construction, training settings, and checkpoint identities appear in Appendices B.2 and B.3.

Baselines. We compare each model and checkpoint with and without RegulaTutor. We further compare with DeepTutor 1.5.1, a tutoring-harness baseline on DeepSeek-v4-pro (Zhao et al., 2026). For GPT-5.6-sol, we also evaluate the native Codex agent as a third harness configuration. To examine PairRoute, we evaluate the crossed combinations of M_0–M_3 with H_0 and H_1.

3.2 MAIN RESULTS

Long-history tutoring. We compare each model and checkpoint with and without the final RegulaTutor harness in Table 1.

Across all nine fixed-backbone comparisons, RegulaTutor improves Evidence accuracy by 13.2 percentage points and Diagnosis accuracy by 11.6 points on average. Table 1 reports these descriptive means separately for four local Qwen checkpoints and five API backbones. Six configurations show reliable Evidence gains, ranging from 10.9 points for DeepSeek-v4-pro (95% CI [8.7, 13.0]) to 36.2 points for the initial Qwen checkpoint (95% CI [34.1, 38.2]). These intervals quantify uncertainty in the estimated gains. For example, for DeepSeek-v4-pro, the plausible mean gain is 8.7–13.0 points. Both intervals exclude zero, indicating that the gains are statistically distinguishable from zero.

Table 1: LongTutor Human Gold1000 (199-record histories). RegulaTutor uses final harness H_1. Green arrows show gains over model-only inference; mean rows average gains within each group. Evidence and Diagnosis gains are percentage points; Teaching gains are score points. Bold marks each model's best score; * marks a positive gain with a paired 95% interval excluding zero. Codex uses GPT-5.6-sol; † marks the backbone also used as Teaching judge.

Model / system	Method	Evidence	Diagnosis	Teaching
Local Qwen checkpoints ($n = 4$)				
Qwen3.5-4B (M_0)	Model only	20.0	29.7	2.63
	RegulaTutor	**56.2*** (↑ 36.2)	**50.8*** (↑ 21.1)	**2.76** (↑ 0.13)
Qwen3.5-4B (M_1)	Model only	21.2	31.3	2.67
	RegulaTutor	**56.4*** (↑ 35.2)	**49.6*** (↑ 18.3)	**2.73** (↑ 0.06)
Qwen3.5-4B (M_2)	Model only	27.9	28.0	**2.79**
	RegulaTutor	**43.9*** (↑ 16.0)	**51.6*** (↑ 23.6)	2.72
Qwen3.5-4B (M_3)	Model only	28.2	27.2	**2.79**
	RegulaTutor	**43.7*** (↑ 15.5)	**51.3*** (↑ 24.1)	2.73
Mean RegulaTutor gain		+25.7	+21.8	+0.016
API backbones ($n = 5$)				
DeepSeek-v4-pro	Model only	41.5	48.2	3.11
	RegulaTutor	**52.3*** (↑ 10.9)	**49.1** (↑ 0.9)	3.33 (↑ 0.22)
	DeepTutor	48.7	37.9	**3.89**
Gemini-3.7-flash	Model only	57.4	**45.5**	3.41
	RegulaTutor	**75.6*** (↑ 18.2)	37.6	**3.92** (↑ 0.51)
GPT-5.6-sol	Model only	**54.9**	37.7	**3.56**
	RegulaTutor	50.1	41.0 (↑ 3.3)	3.26
	Native Codex	44.2	**42.8**	3.43
Kimi-k3	Model only	**47.2**	25.8	3.05
	RegulaTutor	46.0	**33.5*** (↑ 7.7)	**3.75** (↑ 0.70)
DeepSeek-v4.1-flash†	Model only	**56.4**	29.3	3.03
	RegulaTutor	48.9	**42.2*** (↑ 12.9)	**3.67** (↑ 0.64)
Mean RegulaTutor gain		+3.1	+3.4	+0.353
Overall mean gain ($n = 9$)		+13.2	+11.6	+0.203

Across the four Qwen checkpoints, RegulaTutor improves Evidence accuracy by 15.5–36.2 percentage points and Diagnosis accuracy by 18.3–24.1 points. All corresponding 95% confidence intervals exclude zero. At M_0, the Evidence gain is 36.2 points (95% CI [34.1, 38.2]), with improvements on 738 cases, regressions on 48, and ties on 214. Diagnosis improves by 21.1 points (95% CI [17.2, 24.9]), and Teaching rises by 0.132 points (95% CI [.064, .200]).

The API results show the same Evidence–Teaching combination on two backbones: DeepSeek-v4-pro gains 10.9 Evidence points and 0.220 Teaching points, and Gemini gains 18.2 Evidence points and 0.506 Teaching points. Kimi-k3 gains 7.7 Diagnosis points and 0.698 Teaching points. These paired gains have intervals excluding zero.

3.3 Ablation Studies

We evaluate component ablations on LongTutor with the ablation-batch reference held fixed. Removing CAMM interpretation lowers Evidence by 2.0 points and Teaching by 0.164, while fixing the policy lowers them by 1.5 points and 0.144 (Table 2). These within-batch contrasts support the contribution of evidence-grounded interpretation and policy adaptation to the tutoring pathway. The absolute reference score is batch-specific and is therefore used only for paired contrasts.

Table 2: Component ablations in the completed LongTutor ablation batch. The within-batch reference scores are 37.6 Evidence, 44.9 Diagnosis, and 3.340 Teaching; absolute scores are not merged with the main H_1 run because the batch configuration differs. Parentheses report the ablated variant minus the reference.

Variant	Evidence	Diagnosis	Teaching
Full RegulaTutor	37.6	44.9	3.340
Without CAMM interpretation	35.6 (-2.0)	44.8 (-0.1)	3.176 (-0.164)
Static policy	36.1 (-1.5)	44.8 (-0.1)	3.196 (-0.144)
Without compiled directives	43.3 (+5.7)	44.2 (-0.7)	3.228 (-0.112)

3.4 Trace-Guided Refinement and Compatibility

Refinement protocol. PairRoute uses stage-level execution records to distinguish failures in evidence access or presentation from failures in the model's use of available evidence. For harness-related failures, it proposes focused revisions and tests each candidate while holding the model fixed. Under the selected harness, it reviews remaining model failures and correct executions to construct stage-level training examples.

We roll out the current model–harness pair on 369 discovery cases and analyze its execution trajectories to identify harness failures and model errors. A separate 32-case probe set screens candidate harness revisions, while a disjoint 100-case development set checks the selected configurations across refinement rounds. Only reviewed stage-level examples from the discovery cases enter model training.

Appendix B.2 details the review and training procedure. The present study asks whether this supervision improves historical-evidence processing and how the resulting checkpoints behave under different harness versions.

Development-set results. On the 100-case development set, Evidence accuracy under the selected H_1 rises in every model-training round, from 212/300 (70.7%) with M_0 to 250/300 (83.3%) with M_3 (Table 3). The archived (M_0, H_0) run provides an initial-pair reference on the same cases. Later harness proposals were not retained, so $H_2 = H_3 = H_1$. The first model update gains 6.0 percentage points under H_1, with a paired case-bootstrap 95% interval of [1.7, 10.3].

Table 3: PairRoute trajectory on the repeatedly used dev100 set. M_0H_0 is the initial pair; the model-training series uses the selected harness, with $H_2 = H_3 = H_1$. Evidence is correct answers among 300 queries; Diagnosis is accuracy (%); Teaching is the four-dimension mean. Bold indicates the highest value in each column.

Pair	Model update	Evidence	Evidence (%)	Diagnosis	Teaching
M_0H_0	Initial model and harness	208/300	69.3	43	2.690
M_0H_1	Initial model, selected harness	212/300	70.7	40	2.955
M_1H_1	PairRoute SFT, round 1	230/300	76.7	**50**	**3.055**
M_2H_1	PairRoute SFT, round 2	234/300	78.0	**50**	2.965
M_3H_1	PairRoute SFT, round 3	**250/300**	**83.3**	48	2.935

LongTutor Gold1000 results. Across Qwen checkpoints, Evidence accuracy without RegulaTutor rises from 20.0% at M_0 to 28.2% at M_3 (Table 4), a paired gain of 8.2 points (95% CI [6.3,

10.0]). Teaching rises from 2.631 to 2.789, a gain of 0.158 points (95% CI [.088, .226]). Both endpoint scores are included in Table 4. Together with the monotonic development-set Evidence trajectory, these results show that reviewed execution traces provide useful supervision for evidence acquisition and instructional response quality beyond inference-time scaffolding.

Table 4: LongTutor Gold1000 results for Qwen3.5-4B checkpoints crossed with inference conditions. M_0 is the pretrained model; M_1–M_3 follow successive PairRoute stage-supervised training rounds. H_0 is the initial harness and H_1 the final RegulaTutor harness. E and D are Evidence and Diagnosis accuracy (%); T is judged Teaching (1–5). Bold is reserved for the highest values supporting the positive results discussed in the main text.

Model checkpoint	**Model only**			H_0			H_1		
	E	**D**	**T**	**E**	**D**	**T**	**E**	**D**	**T**
M_0 (pretrained)	20.0	29.7	2.631	59.5	51.6	2.752	56.2	50.8	**2.763**
M_1 (round 1)	21.2	**31.3**	2.671	**67.3**	**52.7**	2.721	**56.4**	49.6	2.731
M_2 (round 2)	27.9	28.0	**2.793**	49.3	52.1	2.715	43.9	**51.6**	2.721
M_3 (round 3)	**28.2**	27.2	2.789	59.9	52.0	**2.774**	43.7	51.3	2.733

4 Related Work

Intelligent Tutoring Systems Learning theory and learner modeling have long informed intelligent tutoring systems. MetaTutor supports metacognitive strategies through assessment and feedback (Rus et al., 2010), while Wayang Outpost/MathSpring addresses cognition, metacognition, and affect in adaptive mathematics tutoring (Arroyo et al., 2014). More recently, CLASS explicitly connects learning-science principles with the design of LLM-based tutoring (Sonkar et al., 2023). These systems provide foundations for connecting observations of learner activity to instructional support.

LLM tutoring frameworks extend these ideas with flexible generation, planning, and memory. DeepTutor combines grounded tutoring with dynamic learner memory (Zhao et al., 2026). ScaffoldLM maintains a pedagogical plan and updates tutoring memory through assessment of learner progress (Li et al., 2026b). On the model side, Dinucu-Jianu et al. (2025) train tutors with simulated interactions and rewards that balance pedagogical support and student problem solving. Complementing these system and model developments, LongTutor evaluates the use of learning histories across evidence acquisition, diagnosis, and teaching (Li et al., 2026a). Together, these directions motivate examining how historical evidence informs teaching and how the model executes the resulting support.

Building on these foundations, RegulaTutor connects longitudinal evidence management with theory-informed instructional decisions. Its harness combines multi-stage evidence resolution, CAMM-guided interpretation, persistent learner-specific teaching policies, and learner-facing SRL support. Within this workflow, stored observations support learner interpretations, and policy guidelines translate those interpretations into instructional directives. Incremental session memory allows learner history and policies to be maintained across interactions.

Model–Harness Co-Evolution A model update and a harness update modify different parts of an agent. The former changes reusable execution capabilities, whereas the latter changes the information and procedures available during execution. Their effects need not be independent. Lin et al. (2026) distinguish the ability to produce useful harness updates from the ability to benefit from an updated harness, including failures to activate or follow relevant artifacts. This distinction motivates evaluating the executing model and its interface together.

HarnessForge represents an agent as a harness–policy pair and combines fault-guided harness tailoring with harness-conditioned policy alignment (Chen et al., 2026a). Co-Harness alternates validated local harness updates with model optimization on trajectories produced by the improved harness (Chen et al., 2026b). SIA similarly uses feedback to update both external scaffolding and model weights (Hebbar et al., 2026). Failure-guided adaptation and joint optimization are direct precedents for our pipeline; they motivate allocating repairs according to the component responsible for a failure and validating the resulting pair.

Within RegulaTutor, PairRoute applies failure-guided offline refinement to historical evidence acquisition. It diagnoses failures before harness revision and reassesses them under the selected harness, checking the input scope, output requirements, and evidence available at actual stage calls. Confirmed residual model failures supply stage-level correction targets, which are combined with reviewed current-model outputs for retention. This design connects component-level attribution to supervision under the selected harness, while monitoring downstream tutoring performance.

5 Discussion

Revising learner interpretations over time. A past error remains a valid observation even after a learner has overcome the difficulty, and continuing to remediate it may then be inappropriate. Long-term personalization therefore requires managing the lifetime of an interpretation separately from that of its supporting records. RegulaTutor's evidence-linked policies make this distinction inspectable. A useful extension is to associate each interpretation with both its supporting evidence and conditions for revision, so that accumulating history enables a tutor to revise outdated interpretations.

Evaluating evidence accuracy and teaching quality jointly. In the ablation batch, removing compiled directives increases Evidence while reducing judged Teaching (Table 2). This exposes a distinction between recovering historical facts and deciding how those facts should shape instruction. A component's pedagogical value can therefore be missed by an evidence-only evaluation. Future optimization could seek evidence-processing gains subject to explicit Diagnosis and Teaching criteria, keeping trade-offs visible when selecting model or harness updates.

Maintaining personalization across system updates. The crossed checkpoint results show that the effects of model updates depend on the inference configuration (Table 4). In a long-term tutoring service, learner histories and policies may persist after the model or harness changes. This creates a continuity requirement: the updated system should still interpret and execute those policies appropriately. Future compatibility tests could replay held-out learner histories together with their stored policies before and after an update, checking whether still-supported decisions are preserved in the resulting instruction. The evaluation unit would then include persistent learner state alongside the model and harness.

6 Conclusion

We presented *RegulaTutor*, a theory-informed framework connecting learner history, evidence-supported interpretations, and adaptive teaching policies, together with *PairRoute* for trace-guided offline refinement. LongTutor evaluations show average gains in evidence acquisition and diagnosis, with Teaching gains on several backbones. The refinement study shows development-set gains in evidence processing and characterizes model–harness compatibility. Together, these contributions offer a foundation for exploring lifelong personalized tutoring in which instructional support can evolve alongside learners as their knowledge, goals, and learning contexts change.

Limitations

LongTutor evaluates independent history snapshots and does not measure real learners' retention, transfer, or progress over time. Teaching scores rely on model judges, with some configurations sharing the evaluator's model family.

The refinement study uses one local backbone, three supervised rounds, and an adaptively reused development set. Crossed model–harness evaluations characterize compatibility, without budget-matched controls for alternative refinement strategies. GRPO and stage-level SFT also differ in supervision, objectives, budgets, and checkpoint selection, preventing attribution of score differences to the optimization method alone.

Finally, the harness represents learning-related evidence through engineering heuristics; it does not provide calibrated estimates of psychological traits or establish causal effects of instructional interventions.

AI USE STATEMENT

We used generative AI tools to polish the writing and improve readability. We reviewed all AI-assisted edits and take responsibility for the final content of this work.

ETHICS STATEMENT

Tutoring systems make inferences about learners that may be incomplete or wrong. Our design separates observed information from episode-specific interpretations and discourages unsupported claims about motivation or affect. These safeguards do not establish fairness or educational safety. Use with real learners requires appropriate data permissions, privacy protections, and review of the consequences of instructional decisions. This draft reports no new human-participant study and makes no claim of institutional approval. Any subsequent study must document its actual review and consent procedures.

REPRODUCIBILITY STATEMENT

Appendices A–C specify the current implementation boundary, two-stage round procedure, and benchmark protocols. A companion evidence ledger maps draft claims to local sources, server implementation records, and primary literature. The final release should include immutable model and harness revisions, learner-level split manifests, compiled snapshots, evaluator settings, and outputs for each pair comparison. Formal result admission requires matched generation/scoring artifacts and an overlap audit; obsolete or incompatible runs are excluded rather than used to fill missing results. No anonymous public artifact link is claimed in this version.

REFERENCES

Ivon Arroyo, Beverly Park Woolf, Winslow Burleson, Kasia Muldner, Dovan Rai, and Minghui Tai. A multimedia adaptive tutoring system for mathematics that addresses cognition, metacognition and affect. *International Journal of Artificial Intelligence in Education*, 24(4):387–426, 2014. doi: 10.1007/s40593-014-0023-y. URL https://doi.org/10.1007/s40593-014-0023-y.

Roger Azevedo, François Bouchet, Melissa Duffy, Jason Harley, Michelle Taub, Gregory Trevors, Elizabeth Cloude, Daryn Dever, Megan Wiedbusch, Franz Wortha, and Rebeca Cerezo. Lessons Learned and Future Directions of MetaTutor: Leveraging Multichannel Data to Scaffold Self-Regulated Learning With an Intelligent Tutoring System. *Frontiers in Psychology*, 13:813632, June 2022. ISSN 1664-1078. doi: 10.3389/fpsyg.2022.813632.

Mingju Chen, Can Lv, Guibin Zhang, Heng Chang, and Shiji Zhou. HarnessForge: Joint Harness and Policy Evolution for Adaptive Agent Systems. arXiv preprint arXiv:2606.01779, June 2026a. URL https://arxiv.org/abs/2606.01779.

Zhengyu Chen, Teng Xiao, Huaisheng Zhu, Yige Yuan, Luan Zhang, and Jingang Wang. Co-Harness: Co-Evolving Harnesses and Model Weights for LLM Agents. arXiv preprint arXiv:2607.22688, July 2026b. URL https://arxiv.org/abs/2607.22688.

Zixin Chen, Peng Liu, Rui Sheng, Haobo Li, Jianhong Tu, Xiaodong Deng, Kashun Shum, Dayiheng Liu, and Huamin Qu. TeachArena: Are Language Agents Ready for Realistic Teaching Work? arXiv preprint arXiv:2605.14322, August 2026c. URL https://arxiv.org/abs/2605.14322.

Clayton Cohn, Surya Rayala, Namrata Srivastava, Joyce Horn Fonteles, Shruti Jain, Xinying Luo, Divya Mereddy, Naveeduddin Mohammed, and Gautam Biswas. A Theory of Adaptive Scaffolding for LLM-Based Pedagogical Agents. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(3):1757–1765, March 2026. ISSN 2374-3468, 2159-5399. doi: 10.1609/aaai.v40i3.37154.

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 Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose,

and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 272–292, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.15. URL https://aclanthology.org/2025.emnlp-main.15/.

Anastasia Efklides. Interactions of metacognition with motivation and affect in self-regulated learning: The MASRL model. *Educational Psychologist*, 46(1):6–25, 2011. doi: 10.1080/00461520.2011.538645. URL https://doi.org/10.1080/00461520.2011.538645.

Fatma Betul Gures, Tanya Nazaretsky, and Tanja Kaser. Tailoring Scaffolding to Diagnostic Strategies: Theory-Informed LLM-Based Agents. arXiv preprint arXiv:2605.04996, May 2026. URL https://arxiv.org/abs/2605.04996.

Prannay Hebbar, Yogendra Manawat, Samuel Verboomen, Alesia Ivanova, Selvam Palanimalai, Kunal Bhatia, and Vignesh Baskaran. SIA: Self Improving AI with Harness & Weight Updates. *arXiv preprint arXiv:2605.27276*, 2026. URL https://arxiv.org/abs/2605.27276.

Yoonho Lee, Roshen Nair, Qizheng Zhang, Kangwook Lee, Omar Khattab, and Chelsea Finn. Meta-Harness: End-to-End Optimization of Model Harnesses. arXiv preprint arXiv:2603.28052, March 2026. URL https://arxiv.org/abs/2603.28052.

Ning Li, Zheng Zhang, Zhenya Huang, Rui Li, Yi Zhan, Yinbo Luo, Qi Liu, and Enhong Chen. LongTutor: Benchmarking Large Language Models for Long-term Personalized Tutoring. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29712–29737, San Diego, California, United States, July 2026a. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.1371.

Zechen Li, Qiannan Zhu, Mei Wang, Jia Li, and Hua Huang. Planning-guided tutoring with assessment-driven memory for pedagogical LLM tutors. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7165–7188, San Diego, California, United States, July 2026b. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.325. URL https://aclanthology.org/2026.acl-long.325/.

Minhua Lin, Juncheng Wu, Zijun Wang, Zhan Shi, Yisi Sang, Bing He, Zewen Liu, Tianxin Wei, Zongyu Wu, Zhiwei Zhang, Dakuo Wang, Xiang Zhang, Benoit Dumoulin, Cihang Xie, Yuyin Zhou, Suhang Wang, and Hanqing Lu. Harness Updating Is Not Harness Benefit: Disentangling Evolution Capabilities in Self-Evolving LLM Agents. arXiv preprint arXiv:2605.30621, May 2026. URL https://arxiv.org/abs/2605.30621.

Inge Molenaar, Susanne de Mooij, Roger Azevedo, Maria Bannert, Sanna Järvelä, and Dragan Gašević. Measuring self-regulated learning and the role of AI: Five years of research using multimodal multichannel data. *Computers in Human Behavior*, 139:107540, February 2023. ISSN 0747-5632. doi: 10.1016/j.chb.2022.107540.

Janos Perczel, Jin Chow, and Dorottya Demszky. TeachLM: Post-Training LLMs for Education Using Authentic Learning Data. arXiv preprint arXiv:2510.05087, October 2025. URL https://arxiv.org/abs/2510.05087.

Vasile Rus, Mihai Lintean, and Roger Azevedo. Computational aspects of the intelligent tutoring system MetaTutor. In *Proceedings of the Twenty-Third International Florida Artificial Intelligence Research Society Conference*, pp. 522–527, 2010. URL https://digitalcommons.memphis.edu/facpubs/2641/.

Shashank Sonkar, Naiming Liu, Debshila Mallick, and Richard Baraniuk. CLASS: A design framework for building intelligent tutoring systems based on learning science principles. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 1941–1961, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.130. URL https://aclanthology.org/2023.findings-emnlp.130/.

Yang Wu, Rujing Yao, Tong Zhang, Yufei Shi, Zhuoren Jiang, Zhushan Li, and Xiaozhong Liu. Teaching According to Students' Aptitude: Personalized Mathematics Tutoring via Persona-, Memory-, and Forgetting-Aware LLMs. arXiv preprint arXiv:2511.15163, November 2025. URL https://arxiv.org/abs/2511.15163.

Bingxi Zhao, Jiahao Zhang, Xubin Ren, Zirui Guo, Tianzhe Chu, Yi Ma, and Chao Huang. DeepTutor: Towards Agentic Personalized Tutoring. arXiv preprint arXiv:2604.26962, July 2026. URL https://arxiv.org/abs/2604.26962.

Barry J. Zimmerman. Becoming a self-regulated learner: An overview. *Theory Into Practice*, 41 (2):64–70, 2002. doi: 10.1207/S15430421TIP4102_2. URL https://doi.org/10.1207/S15430421TIP4102_2.

A REGULATUTOR IMPLEMENTATION DETAILS

A.1 HISTORY AND EVIDENCE MANAGEMENT

RegulaTutor maintains a learner history $\mathcal{L}_{u,i}$ across interactions and selects evidence from it for the current task. Each learning event contains its content, timestamp, and source identifier, with concept and correctness fields when available.

We support two history modes. In *snapshot mode*, each request supplies the complete history for an independent task, and learner state is reconstructed from that history without importing a stored policy. For example, a LongTutor case supplies past attempts and asks about a separate current question. In *incremental mode*, a request supplies a new observation or dialogue turn, which is combined with the history and policy already stored for the same learner and session. This supports an ongoing tutoring conversation without resending the entire history. Events are deduplicated and ordered chronologically, and the updated state is saved after a successful interaction. Tutor-generated dialogue remains separate from scored learner observations, so an unscored exchange does not create a mastery label.

Evidence acquisition. The Evidence Resolution Loop (ERL) obtains source evidence $E_{u,i}$ and answers $A_{u,i}$ to the task's information needs. A question about a particular attempt uses focused retrieval. A question spanning several sessions uses mapper calls to select relevant observations from history chunks, followed by consolidation and resolution. Each resolution includes supporting source identifiers. The harness checks these references and the output structure, and may expand retrieval or repeat resolution when support remains insufficient. This procedure bounds evidence acquisition while leaving interpretation and reasoning to the model.

Table 5: Core runtime settings. History and retrieval budgets count events; output limits count tokens per model call.

Setting	**Value**
Bounded history / recent-history window	10,000 / 10 events
Initial retrieval / ERL evidence top-k	8 / 48
Mapper chunk size	40 events
Maximum ERL rounds	2
Internal output limit / temperature	12,000 tokens / 0
CAMM interpretation / evidence admission	Deterministic / ERL-cited events
Maximum policy proposals	4
Policy acceptance threshold / switching cost	0.60 / 0.15

A.2 FROM LEARNER EVIDENCE TO TEACHING DECISIONS

CAMM connects observed learner evidence to instructional support, drawing on SRL and MASRL (Zimmerman, 2002; Efklides, 2011; Molenaar et al., 2023). It represents cognitive and metacognitive interpretations separately, with motivational and affective signals sharing an implementation field. Each interpretation records its supporting observations and the limits of the inference. Table 6 summarizes how these interpretations guide teaching.

Interpretation and policy updates. CAMM uses ERL-cited events and recomputes learner statistics over this admitted evidence. Repeated difficulty requires at least two incorrect outcomes on the same concept; prior-success support requires at least two scored events and accuracy of at least 0.7. Confidence values and acceptance thresholds are fixed engineering heuristics, rather than calibrated psychological probabilities.

RegulaTutor converts supported interpretations into policy proposals, ranked by confidence, guideline relevance, and evidence support. A switching cost favors continuity with the previous policy. When proposals conflict, the highest-ranked eligible assignment determines each policy field; the no-inference guard remains active even when other proposals are rejected. The resulting policy retains links to its supporting observations.

Table 6: CAMM links between learner evidence and instructional support. Rules apply only to admitted observations; absence of a signal does not establish a learner trait.

Observed evidence	Instructional support	Inference boundary
Repeated errors on a concept	Stepwise explanation and a focused check, grounded in a prior attempt	Local difficulty does not imply global inability or low motivation
Supported prior success	Concise explanation and retrieval practice	Prior accuracy does not establish general mastery
Explicit strategy, reflection, confidence, or self-check signal	Strategy comparison and self-checking	An observed action does not establish a stable metacognitive trait
Explicit motivational or affective self-report	Acknowledge the signal and adapt support to its context	Incorrect answers alone do not establish motivation or emotion

Instruction compilation. RegulaTutor combines the CAMM interpretation $C_{u,i}$ and policy $P_{u,i}$ into teaching directives $D_{u,i}$ specifying support mode, explanation depth, history references, and inferences to avoid. The final model input combines these directives with the current task, learner statistics, resolved evidence, and recent dialogue. A shared system prompt also encourages goal setting, self-checking, and reflection when appropriate. Learner history and policy can change between interactions, while model weights and shared harness rules remain fixed during tutoring.

B PairRoute Refinement Study and Model Training Details

B.1 Failure Diagnosis and Harness Refinement

PairRoute studies evidence-processing refinement through candidate harness revision and model training. At round t, Router1 examines execution under (M_t, H_t) and proposes repairs to evidence access or presentation. Candidates are evaluated with M_t fixed to select or retain H_{t+1}. Router2 then reviews residual failures under (M_t, H_{t+1}) and identifies model calls eligible for supervision. Figure 3 summarizes the executed procedure. The current experiments evaluate the resulting trajectory and crossed model–harness pairs; they do not estimate an advantage over alternative routing or update algorithms.

Table 7: Routing decisions based on the evidence visible to the implicated model call.

Disposition	Criterion and action
Harness	Evidence access or the call interface is defective; propose a harness repair.
Model	Evidence and instructions are sufficient, but execution is incorrect; review a correction target.
Mixed	Both defects occur; repair the interface before reassessing model supervision.
Success	The task meets its evaluation criterion; independently review stages for retention.
External	A source, reference, or evaluation problem prevents a clean correction target.
Defer	Attribution remains uncertain; withhold the case from confirmed corrections.

We instantiate the two routers as Codex agents with separate review instructions. They inspect the task, reference evaluation, and actual model-call inputs and outputs. The offline pipeline checks call identities, source references, and review structure; the agents judge semantic correctness. Router1's proposed repairs receive manual refinement before candidate evaluation. Router2 reviews calls under the selected harness so that training targets reflect the interface used by the student.

B.2 Stage-Level Supervision and Training

We construct training data from the 369-case discovery subset of the LongTutor XES3G5M silver-data pipeline, separate from Human Gold1000 used for benchmark evaluation (Section C.2). Each discovery case supplies 199 history records, a current question, and three evidence queries: information extraction (IE), multi-session reasoning (MR), and hallucination control (HC). We execute the

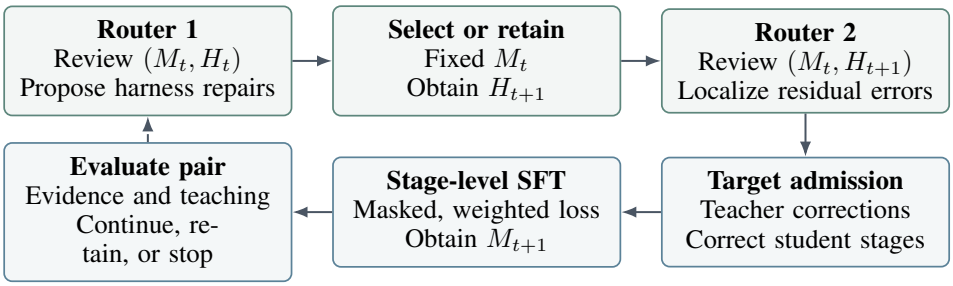


Figure 3: A PairRoute round. Harness selection precedes the construction of model supervision. Trajectories may be reused when the selected model–harness pair is unchanged.

student under the selected harness on these cases and use the resulting calls to construct supervision; the silver reference answers are evidence for review rather than targets copied directly into training.

Answering a query may involve several model calls. A *stage-level* example supervises one call—a mapper selecting records, a resolver answering from evidence, an audit checking support, or a final response generator—using its recorded input and a reviewed target.

The training set $\mathcal{D}_t = \mathcal{D}_t^{\text{corr}} \cup \mathcal{D}_t^{\text{ret}}$ combines teacher corrections with correct outputs from the current student. Retention is reviewed at the call level, including correct stages within failed tasks. Teacher targets must be valid in full, even when only part of the output receives loss. If a teacher trajectory uses different upstream context, its compatibility with the training obligation is checked explicitly. Hidden references and routing judgments are excluded from student inputs.

Teacher and loss. Kimi K3 supplies teacher targets with low thinking effort, temperature 1, and a 12,000-token output limit. Compatible reviewed targets may be reused across rounds. For input x_r, target y_r, admitted-token mask $m_{r\ell}$, and token weight $w_{r\ell}$, the microbatch objective is

$$\mathcal{L}_{\mathcal{B}}(\theta) = -\frac{\sum_{r\in\mathcal{B}}\sum_{\ell} m_{r\ell} w_{r\ell} \log p_\theta(y_{r\ell} \mid x_r, y_{r,<\ell})}{\sum_{r\in\mathcal{B}}\sum_{\ell} m_{r\ell} w_{r\ell}}. \tag{3}$$

Prompt and padding tokens receive no loss. We balance total supervised-token mass across IE, MR, and HC; the three categories need not have equal numbers of examples.

Training configuration. All updates use LoRA with rank 16, scaling factor 32, and dropout 0.05 on attention, linear-attention, and feed-forward projections. We use AdamW, a linear learning-rate schedule with 5% warmup, gradient clipping at 1, BF16 precision, and gradient checkpointing. Each update runs for two epochs with global batch size 4, seed 42, and a maximum sequence length of 34,000 tokens; no admitted example is truncated. The first update initializes a new adapter on Qwen3.5-4B. Later updates continue the preceding adapter with a fresh optimizer. Every round uses the final epoch, without development-based checkpoint selection.

Table 8: Training data and model updates. Corrections and retention count unique stage examples; each corpus has equal IE/MR/HC supervised-token mass.

Update	Cases	Corrections	Retention	Total	Learning rate
$M_0 \to M_1$	256	264	137	401	2×10^{-5}
$M_1 \to M_2$	121	87	70	157	10^{-5}
$M_2 \to M_3$	111	60	60	120	10^{-5}

B.3 Selected Harness and Refinement Rounds

M_0 denotes Qwen3.5-4B and H_0 the initial harness with incremental learner-state management. The reported H_0 is frozen at tag `H\_0`, commit `81a7e37`, with configuration-hash prefix `88c2ad8f`. The first round produces H_1 through Router1-guided revision and manual refinement. Later harness

candidates are not retained, so H_1 is the final PairRoute-refined harness reported as RegulaTutor in the main fixed-backbone results. It adds current-context binding, which clarifies the current task's referent, and an indexed factual-support audit. The audit checks answered resolutions for queries that a language-based detector does not classify as collection operations; detected collection queries bypass it. Both classification and support checking remain fallible model/harness operations.

The revisions target two evidence failures observed during development: binding a query to the wrong current context and accepting an answer without sufficient source support. After retaining H_1 for the next stage, PairRoute continues to review residual failures and construct model supervision. Later harness candidates do not meet the screening criteria, so $H_2 = H_3 = H_1$ while the model advances to M_2 and M_3. These version transitions describe the executed refinement trajectory; whether any candidate improves the complete system must be determined from the matched evaluations rather than inferred from its position in the sequence.

B.4 Matched Comparisons

To distinguish harness and model changes, adjacent-round comparisons consider (M_{t-1}, H_{t-1}), (M_{t-1}, H_t), (M_t, H_{t-1}), and (M_t, H_t) under the same evaluation protocol. For a score s, their interaction contrast is

$$\begin{aligned}\Delta_{\mathrm{int},t} = s(M_t, H_t) - s(M_t, H_{t-1}) \\ - s(M_{t-1}, H_t) + s(M_{t-1}, H_{t-1}).\end{aligned} \tag{4}$$

When the harness is retained, identical configurations collapse and the contrast is zero if the same observations are reused. In round 3, Router2 reuses the (M_2, H_2) trajectories after verifying that they also represent (M_2, H_3). These comparisons characterize the updates made within PairRoute; establishing a benefit from routing itself would require controls with matched data and review budgets. Table 4 reports the completed full-benchmark checkpoint–harness grid. Because these are crossed evaluations of one training trajectory rather than independently optimized control branches, we use them to characterize compatibility and do not estimate $\Delta_{\mathrm{int},t}$ as evidence that routing outperforms an alternative update algorithm.

C Datasets and Evaluation Protocols

This appendix records the supplementary protocols and analyses that are not used as primary claims in the main text. LongTutor is the primary benchmark for the paper. TeachArena is retained as an exploratory transfer study because it exercises simulated interaction and tool execution under a different runtime; its results should be read as protocol-specific evidence about interface compatibility, rather than as a second main benchmark.

C.1 Benchmarks and Metrics

LongTutor. We use XES3G5M Human Gold1000, comprising 1,000 cases with 199 history records and a separate current question per case (Li et al., 2026a). All systems receive the same public history and task inputs. Reference answers, hidden diagnostic statistics, and evaluator metadata are withheld. IE, MR, and HC are scored separately, and their arithmetic mean is the Evidence score. Diagnosis uses accuracy and macro-F1 over four classes. Teaching uses the mean of History Utilization, Strategy Alignment, Coherence, and Appropriateness, each scored on a 1–5 scale. The separately returned overall judgment is an auxiliary measure.

TeachArena. We run the `edu_agent` domain in the TeachArena τ^2-bench fork with `EDU_CANVAS_MODE=mock` (Chen et al., 2026c). We report Paper Stage 2 (repository split `s0`), which contains 114 situated multi-turn tutoring tasks, and Paper Stage 3 (repository split `s2`), which contains 136 Canvas workflow tasks. Paper Stage 1 was not executed and is not included in any aggregate. Stage 2 evaluates the dialogue trajectory with a simulated learner; Stage 3 evaluates tool-mediated actions and the resulting LMS state. Rewards lie in $[0, 1]$.

C.2 Data Splits

PairRoute uses separate discovery, probe, and development sets from the XES3G5M silver-data pipeline (Table 9). For harness refinement, Router1 diagnoses failures on discovery trajectories and proposes interface changes. Candidates are tested with the model fixed, using selected discovery cases for targeted checks and probe for screening. Development assesses the selected model–harness pairs across rounds. For model training, Router2 reviews discovery trajectories under the selected harness and constructs stage supervision. The probe and development sets are reused adaptively during the executed refinement procedure.

The full-benchmark harness comparison evaluates direct inference, the initial harness H_0, and the refined harness H_1 with a fixed backbone. Component ablations assess the contributions of evidence resolution, CAMM, policy adaptation, and compiled directives (Section D.2). These benchmark comparisons are distinct from the candidate-screening experiments used to select a harness.

Table 9: Data used for refinement and evaluation. The three LongTutor adaptation sets are pairwise disjoint and have no overlap with Human Gold1000 in case, learner, or complete-history identity.

Set	Cases/tasks	Purpose
Discovery	369	Harness failure diagnosis, targeted candidate checks, and stage supervision
Probe	32	Harness candidate screening
Development	100	Adaptive assessment of model–harness updates
LongTutor Gold1000	1,000	System, harness, and model–harness pair evaluation
TeachArena dialogue	114	Situated multi-turn tutoring
TeachArena workflows	136	Tool-mediated Canvas tasks

We verify separation using case and learner identifiers and hashes of complete histories. Each LongTutor set contains one case per learner, and all admitted training episodes belong to discovery. TeachArena contributes no examples to the three stage-SFT corpora. These checks concern the refinement data, not the backbone's pretraining exposure.

C.3 Evaluation Settings

LongTutor cases use independent history snapshots. Its judge is `deepseek-v4.1-flash`, run at temperature zero. Invalid outputs remain in the 1,000-case denominator; the Kimi-k3/H_1 arm has 126 such cases. For paired comparisons, we align arms by case identifier and bootstrap the case-level difference with 3,000 resamples (seed 20260921), reporting percentile 95% intervals. Evidence, Diagnosis, and Teaching each use all 1,000 paired cases; Teaching is included only after all four rubric scores have been completed for every record. The intervals are descriptive and receive no multiplicity correction. PairRoute development comparisons instead use Codex reviews under fixed rubrics and are kept separate from the full benchmark.

The original content-scoring pass left 9,846 judgments missing across 20 arms after evaluator calls reached the routing service's rate limit. We re-issued only these missing calls with the same judge and decoding settings, retained every existing judgment unchanged, and merged the repaired records by case identifier. The completed records contain all four Teaching rubrics for every case; the original and repaired files remain separately archived.

TeachArena retains dialogue and public tool feedback within a task and resets state between tasks. The student simulator is `qwen3-max`; the judge is `gpt-5.1-2025-11-13` at temperature zero. Each agent receives at most 60 steps. Direct conditions use the stock `llm_agent`; H_1 wraps the same backbone as a τ^2 agent. We report both full-set means, where failed executions receive zero, and paired means over task identifiers completed by both conditions. Paired 95% confidence intervals resample task identifiers jointly across the two conditions and describe sample uncertainty for the recorded runs, rather than variation across seeds or judges.

D BASELINES AND ABLATION PROTOCOL

D.1 BASELINE CONFIGURATIONS

We compare model-only inference, DeepTutor, native Codex, the initial harness H_0, and the final PairRoute-refined harness H_1. Model-only inference exposes the model to the public task inputs without RegulaTutor's intermediate stages. DeepTutor supplies a tutoring-harness baseline on DeepSeek-v4-pro; native Codex supplies a GPT-5.6-sol agent configuration with its own runtime. The main RegulaTutor results use H_1; the appendix retains H_0/H_1 comparisons to characterize the harness revision and model–harness compatibility. Table 10 summarizes the current evaluation coverage.

Table 10: System configurations and recorded evaluation coverage. LT denotes LongTutor; TA-D and TA-W denote TeachArena dialogue and workflow tasks.

System	Recorded coverage	Configuration boundary
Direct inference	LT: nine model/checkpoint settings; TA-D/TA-W: three backbones	Public inputs and benchmark agent contract
DeepTutor	LT: deepseek-v4-pro only	Real v1.5.1 chat engine; task-local sessions
Native Codex	LT: GPT-5.6-sol agent condition	Own agent runtime and execution budget
RegulaTutor H_0	LT: M_0–M_3 and deepseek-v4-pro	Frozen initial harness, distinct from direct inference
RegulaTutor H_1	LT: nine settings; TA-D/TA-W: three backbones	Final PairRoute-refined harness; later H_2, H_3 retain it

LongTutor uses the public 199-record histories and scoring protocol in Section C. The condition using the tested model as Teaching judge is identified in result tables and included in descriptive group means. Although native Codex uses GPT-5.6-sol, its distinct runtime means that comparison with model-only inference or RegulaTutor does not isolate a harness effect.

GPT and Gemini use JSON-object generation after strict JSON-schema requests are rejected. Response validation and retry settings are therefore recorded alongside decoding parameters. The prepared protocol permits 3,000 output tokens for direct and harness final responses and 12,000 for internal harness calls. DeepTutor and Codex use their respective execution budgets. We account for all calls and retries rather than assuming equal computation.

D.2 ABLATION DEFINITIONS

Each ablation uses the same backbone, task set, evaluator, and starting state within the ablation batch. Table 11 defines the interventions, and Table 2 reports the corresponding aggregate contrasts. The ablation batch has a different absolute reference score from the main H_1 run, so we interpret its paired differences rather than merge its absolute values with the main table.

Table 11: Component interventions for testing the evidence-to-instruction pathway. The completed score table is reported in Table 2.

Variant	Intervention	What the contrast measures
Without CAMM pathway	Disable interpretation and its downstream policy/directive pathway; retain evidence inputs and static support	Combined contribution of the CAMM-conditioned pathway, not the compiler alone
Static policy	Keep CAMM and its direct implications but fix policy fields to their default values	Contribution of adaptive policy-field selection beyond direct CAMM implications
Without compiled directives	Keep internal interpretation/policy computation but remove compiled directives from the final prompt; freeze other field exposure	Contribution of explicit instruction compilation under the recorded exposure policy

The completed batch shows positive support for CAMM interpretation and adaptive policy selection: removing CAMM reduces Evidence by 2.03 points and Teaching by 0.164, while a static policy

reduces Evidence by 1.50 points and Teaching by 0.144. Diagnosis changes are not detected for these contrasts.

D.3 GRPO Training-Path Probe

As a supplementary training-path probe, we compare two cumulative GRPO checkpoints from a separate training batch under the same recorded H_1 interface. G1 is the first GRPO update and G2 is a second adaptive update from G1; both use LoRA adapters over the same frozen base. Within this batch, Evidence increases from 47.23% to 64.57%, a paired gain of 17.33 points (95% CI [15.9, 18.7]), while Diagnosis and Teaching remain statistically unchanged. This result shows that the training path can interact strongly with the evidence interface; its absolute scores are not merged with the main SFT checkpoint table.

Table 12: GRPO training-path probe on LongTutor Gold1000. The same checkpoints are evaluated with direct inference and with the final H_1 harness. Parentheses report the paired change from G1 to G2; Evidence and Diagnosis are percentages, and Teaching is on a 1–5 scale.

Checkpoint	Interface	Evidence	Diagnosis	Teaching
G1	Direct	30.9	31.6	2.768
G2	Direct	30.1	28.0	2.758
G1	RegulaTutor H_1	47.2	49.0	2.917
G2	RegulaTutor H_1	**64.6**	48.3	2.884
G1 $\rightarrow$ G2 under H_1		+17.3	-0.7	-0.034

D.4 Interface Controls

For TeachArena workflows, output validity and task reward measure different aspects of performance. Table 13 reports response-contract variants recorded within the H_1 agents. Relaxing strict schema enforcement sharply improves completion for DeepSeek and Gemini, showing that interface reliability must be separated from instructional quality.

Table 13: TeachArena workflow response-mode controls within H_1. Reward uses all 136 tasks; scored cases reached evaluation. The dash denotes an unavailable count, not zero.

Backbone	Response mode	Reward	Scored cases
Qwen3.5-4B	Strict schema	.282	130/136
	JSON object	.001	2/136
DeepSeek-v4-pro	Strict schema	.430	101/136
	JSON object	.531	127/136
Gemini-3.7-flash	Strict schema	$\approx$ .000	—
	JSON object	.554	120/136

The response-mode effect differs across backbones. Qwen's JSON-object run records 93 unterminated-string errors and only 2/136 scored tasks. For DeepSeek, relaxing the schema reduces unscored executions from 35 to 9. Gemini's strict-schema condition produces empty tool arguments; its exact scored-task count is not available in the report. These controls change the response mode within H_1 and do not constitute a matched Direct/H_1 validation-and-retry ablation. The main transfer comparison uses the better-performing reported mode for each backbone, so it characterizes the selected interface configuration.

E Additional Results and Analyses

This section provides analyses that complement the main results without repeating their aggregate tables. All paired intervals use the same 1,000 LongTutor cases and the completed Teaching judgments described in Section C. Development scores use a different split and evaluator and are reported only in Table 3.

E.1 LongTutor: Paired and Query-Level Effects

Table 1 gives the aggregate Evidence, Diagnosis, and Teaching scores. Table 14 adds paired uncertainty for each fixed-backbone contrast; Table 15 separates the three Evidence query types. Evidence averages information extraction (IE), multi-session reasoning (MR), and hallucination control (HC); Diagnosis is four-class accuracy; Teaching averages History Utilization, Strategy Alignment, Coherence, and Appropriateness on a 1–5 scale. Invalid generations remain in the 1,000-case denominator for Evidence and Diagnosis.

Table 14: Paired effects of the final RegulaTutor harness H_1 relative to Direct on LongTutor. Evidence and Diagnosis differences are percentage points; Teaching differences use the 1–5 scale. Every interval uses 3,000 paired case-bootstrap resamples over the same 1,000 cases. † marks the evaluator backbone, excluded from across-backbone averages.

Backbone	Evidence Δ [95% CI]	Diagnosis Δ [95% CI]	Teaching Δ [95% CI]
Qwen M_0	+36.17 [34.1, 38.2]	+21.10 [17.2, 24.9]	+.132 [.064, .200]
Qwen M_1	+35.23 [33.1, 37.2]	+18.30 [14.1, 22.3]	+.060 [–.006, .127]
Qwen M_2	+16.00 [13.9, 18.1]	+23.60 [19.3, 27.9]	–.072 [–.138, –.006]
Qwen M_3	+15.50 [13.4, 17.6]	+24.10 [19.9, 28.2]	–.057 [–.124, .012]
DeepSeek-v4-pro	+10.87 [8.7, 13.0]	+.90 [–2.1, 3.8]	+.220 [.172, .268]
Gemini-3.7-flash	+18.23 [16.6, 20.0]	–7.90 [–11.7, –4.1]	+.506 [.458, .553]
GPT-5.6-sol	–4.87 [–7.0, –2.8]	+3.30 [–.1, 6.9]	–.302 [–.347, –.255]
Kimi-k3	–1.23 [–3.8, 1.3]	+7.70 [4.4, 10.9]	+.698 [.600, .791]
DeepSeek-v4.1-flash†	–7.50 [–9.4, –5.5]	+12.90 [9.5, 16.2]	+.645 [.591, .699]

Table 15: Per-query Evidence differences for final RegulaTutor H_1 relative to Direct. Values are percentage points with 95% paired bootstrap intervals over 1,000 cases. IE, MR, and HC denote information extraction, multi-session reasoning, and hallucination control. † marks the evaluator backbone.

Backbone	IE Δ [95% CI]	MR Δ [95% CI]	HC Δ [95% CI]
Qwen M_0	+43.2 [39.4, 47.0]	+17.9 [14.9, 20.8]	+47.4 [44.1, 50.7]
Qwen M_1	+39.7 [35.8, 43.6]	+18.3 [15.3, 21.4]	+47.7 [44.6, 51.0]
Qwen M_2	+2.5 [–1.5, 6.5]	+1.1 [–1.6, 3.8]	+44.4 [41.1, 47.7]
Qwen M_3	+3.1 [–.8, 7.2]	+1.9 [–1.0, 4.7]	+41.5 [38.1, 44.7]
DeepSeek-v4-pro	–33.4 [–37.5, –29.2]	+23.6 [19.8, 27.2]	+42.4 [39.3, 45.5]
Gemini-3.7-flash	+25.0 [22.1, 27.9]	+25.3 [22.0, 28.5]	+4.4 [2.2, 6.6]
GPT-5.6-sol	–57.9 [–61.2, –54.7]	+9.5 [5.9, 13.1]	+33.8 [30.8, 36.7]
Kimi-k3	–7.7 [–11.8, –3.7]	+5.6 [1.8, 9.3]	–1.6 [–4.8, 1.7]
DeepSeek-v4.1-flash†	–64.9 [–68.1, –61.7]	+18.7 [14.7, 22.4]	+23.7 [20.8, 26.6]

The query-level results qualify the aggregate Evidence gains: M_0, M_1, and Gemini improve on all three query types, while the response is uneven on other backbones. Hallucination control improves reliably on seven of the eight configurations judged independently of the tested backbone. Excluding the self-judged DeepSeek-v4.1-flash condition, the eight Direct/H_1 pairs average gains of 15.7 Evidence and 11.4 Diagnosis percentage points. These are descriptive averages over configurations, not an estimate of learner-level effects.

E.2 Interactive Transfer and Interface Stress Test

TeachArena tests simulated dialogue and tool-mediated workflows under a different protocol from LongTutor. Table 16 reports full-task rewards and outcomes conditional on both configurations completing a task. The interface response-mode controls in Table 13 help interpret differences in execution coverage.

Table 16: TeachArena results. Full means assign zero to failed executions. Paired means and confidence intervals use task identifiers completed by both Direct and H_1. W/L/T gives H_1 wins, losses, and ties.

Task	Backbone	Direct	H_1	n	Δ	95% CI	W/L/T
Dialogue	Qwen3.5-4B	.572	.587	106	+.015	[–.003,.034]	36/24/46
	DeepSeek-v4-pro	.672	.621	107	–.010	[–.024,.002]	22/29/56
	Gemini-3.7-flash	.666	.641	111	–.007	[–.021,.006]	23/28/60
Workflows	Qwen3.5-4B	.547	.282	122	–.290	[–.362,–.217]	18/87/17
	DeepSeek-v4-pro	.682	.531	127	–.129	[–.201,–.061]	30/59/38
	Gemini-3.7-flash	.646	.554	120	–.011	[–.072,.051]	29/43/48

Full-task rewards include failed executions as zero; paired intervals condition on tasks completed by both arms. Under H_1, dialogue failures are 4/114, 7/114, and 3/114 for Qwen, DeepSeek, and Gemini, respectively; workflow failures are 6/136, 9/136, and 16/136. All three dialogue paired intervals include zero, while Qwen and DeepSeek workflow intervals exclude zero in the negative direction. The paired intervals therefore cannot be read as uncertainty intervals for the full-task differences.

E.3 Model–Harness Compatibility

The crossed Gold1000 scores appear in Table 4; Table 17 adds paired intervals for the $H_1 - H_0$ contrast at fixed checkpoints. Direct inference is a separate public-input baseline, not H_0. Later harness aliases collapse to H_1 because $H_2 = H_3 = H_1$.

Table 17: Harness compatibility on the same 1,000 cases. Differences are $H_1 - H_0$; Evidence and Diagnosis use percentage points. Teaching uses score units. Intervals are unadjusted paired case-bootstrap intervals.

Comparison	Evidence Δ [95% CI]	Diagnosis Δ [95% CI]	Teaching Δ [95% CI]
M_0: $H_1 - H_0$	-3.37 $[-5.30, -1.50]$	-0.80 $[-2.90, 1.20]$	$+0.011$ $[-0.035, 0.061]$
M_1: $H_1 - H_0$	-10.83 $[-12.60, -9.10]$	-3.10 $[-5.10, -1.20]$	$+0.010$ $[-0.041, 0.061]$
M_2: $H_1 - H_0$	-5.40 $[-7.30, -3.40]$	-0.50 $[-2.30, 1.30]$	$+0.006$ $[-0.045, 0.061]$
M_3: $H_1 - H_0$	-16.20 $[-18.20, -14.30]$	-0.70 $[-2.40, 0.90]$	-0.041 $[-0.094, 0.011]$
DeepSeek: $H_1 - H_0$	$+1.97$ $[-0.10, 4.10]$	$+4.60$ $[1.60, 7.70]$	-0.010 $[-0.067, 0.050]$

H_1 scores below H_0 on Evidence at every Qwen checkpoint, with all four paired intervals excluding zero; the largest decrease is 16.20 percentage points at M_3. Local Teaching contrasts are inconclusive, while DeepSeek's Diagnosis improves by 4.60 points. These crossed evaluations characterize compatibility along one refinement trajectory; they do not compare PairRoute with an independently trained alternative update method.