EMBEDDING-ONLY KNOWLEDGE TRACING: PREDICTING RESPONSES WITHOUT KNOWLEDGE COMPONENT ANNOTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Knowledge tracing (KT) predicts a learner's next response from the response history, and adaptive systems use it to choose the next question and to find learners who need support. Standard KT methods rely on expert-defined knowledge components (KCs) with a costly concept vocabulary and on a learned question-ID table, which has no row for questions absent from training. Embedding-only Knowledge Tracing (EoKT) feeds fixed general-purpose question-text embeddings to a compact recurrent predictor, so it needs neither KC annotations nor an ID table. On MOOCRadar and XES3G5M, we change only the question-side input of this predictor and score warm questions and cold questions withheld from training. In standard AUC, which pools test responses to almost entirely seen questions, EoKT reached the validation-selected KC-based method, with paired differences of +0.0056 on MOOCRadar and +0.0046 on XES3G5M under unequal search budgets. In the same predictor, a learned ID table was at least as good on warm questions, and adding learned or pretrained KC inputs did not raise AUC. In cold AUC, which ranks each learner's responses to questions withheld from training, text embeddings reached 0.6041 on MOOCRadar and 0.6489 on XES3G5M, against 0.5047 and 0.5896 for learned IDs, which cannot distinguish withheld questions; the ordering held for three embedding models. Text embeddings from gemini-embedding-2 also exceeded permuted embeddings on both datasets by +0.0848 and +0.0833 in cold AUC, with 95% intervals that exclude zero when learners and questions are both resampled, so matching each question to its own text contributes to the gain. With question representations precomputed, EoKT returns one prediction from a retained state in about 0.50 ms on a single CPU thread. EoKT can be trained across datasets by simple concatenation: one model pooled over three datasets that differ in language and subject lost at most 0.0012 AUC against per-dataset models in a single-fold check. These results support EoKT as a practical KC-free knowledge tracing method whose text input also supports prediction on questions absent from training.

1 INTRODUCTION

Predicting whether a learner will answer the next question correctly lets an adaptive system choose the next question and identify learners who need support. Knowledge tracing (KT) makes this prediction from response histories. Since DKT, a range of KT methods has been proposed (Piech et al., 2015).

Unpredictable response variability can limit the area under the receiver operating characteristic curve (AUC), even for a Bayes-optimal predictor (Ushio et al., 2026). In our evaluation dominated by previously seen questions, ID-only qDKT (Sonkar et al., 2020) is within about 0.01 AUC of the best configuration on both datasets (Table 1). How much further can prediction improve? For XES3G5M, assuming a uniform 7.1% probability of a wrong answer despite knowing the answer (slip) yields a conditional upper bound of 0.8516 (about 0.85), given the answer-format constraints and observed correctness rates. This is an assumption-dependent reference, not an empirically measured ceiling or proof of saturation (see Appendix H for the calculation and sensitivity). It motivates asking what question text contributes on questions absent from training, alongside gains in pooled AUC.

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.

Preparing knowledge component (KC) annotations for new material requires a concept vocabulary and assignments of questions to concepts. On MOOCRadar, 40 teachers and teaching assistants assigned 5,600 concepts to 2,513 questions, with two independent annotators per question and further expert checking (Yu et al., 2023). ID-only qDKT needs no annotations but lacks a trained representation for an unseen question (Sonkar et al., 2020). Question text and response history allow new questions to enter a predictor without KC annotation or question-specific ID training. We group recent KT methods as follows.

KC- and ID-based methods. Recurrent and attention-based KT models learner histories from KCs or question IDs. DKT updates a recurrent state (Piech et al., 2015), AKT and simpleKT attend to past responses (Ghosh et al., 2020; Liu et al., 2023b), and IEKT models learner-specific cognition and acquisition sensitivity (Long et al., 2021). These methods exploit observed questions and histories, but representing a new question requires a KC assignment or a trained question-specific representation.

Question-text representation methods. Representations derived from text allow questions absent from training to enter a predictor. EERNN jointly trains a text encoder and history model; EKT adds KC-indexed states (Su et al., 2018; Liu et al., 2021). ESKT freezes Longformer representations after tuning on RACE but requires option responses and answer keys (Cheng and Li, 2025). Text-LENS combines mathBERT question representations with a partial VAE for unseen-question generalization (Khan et al., 2025), while EAKT uses LLM-inferred attributes and graph attention for unseen or sparsely answered questions (Guo et al., 2024). KC-free prediction and fixed representations are thus existing options. We do not rerun SINKT (Fu et al., 2024), Text-LENS, or EAKT on our holdouts: their input requirements, encoder training, and history models differ at once, so accuracy gaps would not isolate the contribution of text, and a cold comparison with them remains future work.

Language-model predictors. Methods such as LKT process textualized questions and histories with a pretrained language model, using language representations directly for response prediction (Lee et al., 2024). They execute that model for each prediction, so precomputing question representations alone cannot replace the per-response computation. EoKT precomputes question representations and predicts with a compact history model. Appendix G details input requirements and evaluation settings, including KC-generation and graph-based methods and few-student or short-history settings.

This paper evaluates Embedding-only Knowledge Tracing (EoKT), which combines fixed general-purpose question-text embeddings with a compact recurrent predictor. We ask how closely a model without KC inputs approaches KC-based methods and when question text contributes. We compare text, learned question IDs, and added KCs under a shared predictor, fusion point, and partition, and include EERNN-A as a KC-free prior configuration. We separate warm-dominated evaluation from cold evaluation on withheld questions, also comparing warm and cold responses of the same holdout-trained models. Public embedding replacements, permuted text-to-question correspondence, numeric-variant strata, and a history-free control on first encounters test the conditions under which text contributes. We report two AUCs. Standard AUC pools all test responses of normally trained models, almost all of which concern questions seen in training. Cold AUC computes AUC over each learner's responses to questions withheld from training and averages over learners (pooled warm AUC and within-learner cold AUC in the appendices).

2 Embedding-only Knowledge Tracing

EoKT turns the text of a question into one vector and predicts the response from that vector and the learner state before the response. Let x_q be the question text, v_q its representation, and $h_{u,t-1}$ the state of learner u before response t (Figure 1). A fixed pretrained model gives the embedding e_q. A bias-free projection $P \in \mathbb{R}^{512\times 3072}$ maps it to 512 dimensions, and the Input Refinement Block (IRB) described below transforms it into v_q (Eq. 1). Below, v_{q_t} denotes v_q for the question q_t of response t. The predicted probability is a scaled dot product between the state and the question vector, plus a term $b(v_{q_t})$ that passes the question vector through a small 512→64→1 MLP, and a scalar b_0 (Eq. 2). After the response y_t is observed, a GRU updates the state from the sum of four 512-dimensional terms (Eq. 3). The first term is the question vector. The second is a learnable embedding E_y of the binary response. The third is a time-gap term time_t, which maps $\log(1+\Delta t)$ of the seconds since the previous response through a 1→16→512 MLP; the first response uses seven days. The fourth embeds a binary flag for text-embedding availability. It is always 1 in both main datasets, adding the same learned 512-dimensional vector to every history input; missing embeddings are untested. EoKT has

no trainable learner-ID or question-ID table. Unless otherwise specified, question-text embeddings are generated by Google's gemini-embedding-2. We use fixed 3072-dimensional representations of question text and answer choices, without KC fields, correct answers, or explanations. Two public 1024-dimensional embedding models are evaluated as replacements in Section 4.3.

$$\begin{aligned} e_q &= E_{\text{pretrained}}(x_q), \\ v_q &= \text{IRB}(Pe_q) \end{aligned} \tag{1}$$

$$p_{u,t} = \sigma\left(h_{u,t-1}^{\mathsf{T}} v_{q_t} / \sqrt{512} + b(v_{q_t}) + b_0\right) \tag{2}$$

$$h_{u,t} = \text{GRU}(h_{u,t-1},\ v_{q_t} + E_y(y_t) + \text{time}_t + \text{available}_{q_t}) \tag{3}$$

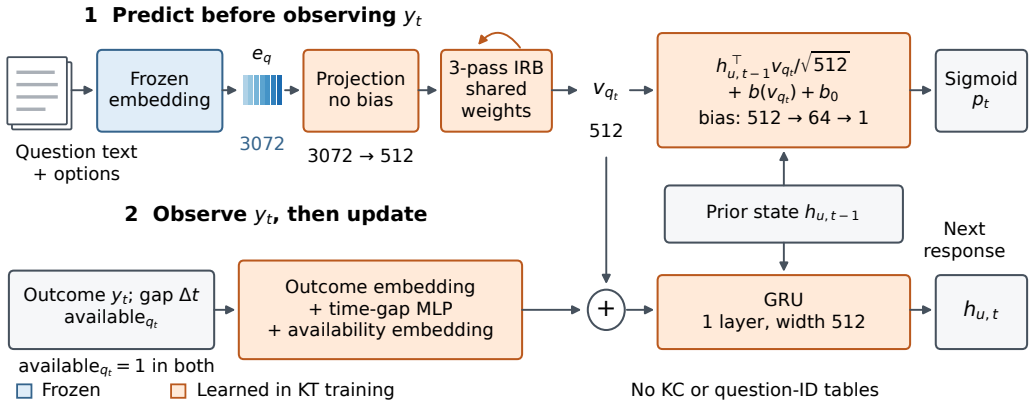

Figure 1: EoKT prediction and post-response state update.

The IRB is a small attention block that refines the projected vector; we treat it as a fixed part of the predictor, not as a contribution. It splits the projected 512-dimensional vector into two tokens, applies a shared three-layer attention block three times, and adds the change back to the input (Appendix A gives the block and Eq. 4). An equal-size MLP changes standard AUC by −0.0005 [−0.0008, −0.0003] and cold AUC on MOOCRadar by +0.0050, an interval that includes zero once questions are also resampled [−0.0058, +0.0153] (Table 3); we claim no IRB-specific advantage. The input controls of Section 4 (question IDs, permutation, KC additions, and embedding models) hold this block fixed and change only the question-side input; Table 16 covers the IRB-free version.

The conditions that add KC inputs add a KC representation to the same question vector after the IRB. The KC input is the only difference from EoKT. Learned KC inputs average trainable vectors of the KCs assigned to the question. Pretrained KC inputs average fixed 3072-dimensional embeddings of the KC names projected to 512 dimensions. Both enter the prediction and the state update. A KC unseen in training contributes zero, including the projection bias. The condition without the IRB keeps the projection and the GRU and drops only the block. Every condition is trained independently.

3 Evaluation Setup

The main evaluation uses MOOCRadar and XES3G5M, which differ in how the questions were collected and in their KC systems (Yu et al., 2023; Liu et al., 2023a). Every condition in this paper is trained and scored on the same partitions and the same scoring keys. Throughout, a question is warm when its responses appear in training and cold when they were withheld from training. MOOCRadar pools 24 courses under one fine-grained concept inventory. XES3G5M is a mathematics dataset with leaf KCs, and it keeps its official pyKT partitions (Liu et al., 2022). We chose these two datasets because their question text is public and their KC systems differ. EdNet does not release question text and is not evaluated here (Norris et al., 2025). For MOOCRadar we use our own partition: 2,794 previously separated learners form the fixed test set, and the remaining learner IDs are hashed into five folds. Following pyKT, we train on four folds and validate on the fifth (Liu et al., 2023b; 2022). A

third dataset, NIPS34, serves only as an auxiliary dataset in the pooled-training check of Experiment 6 (Section 4.6; Appendix A).

The evaluation follows the question-level protocol of pyKT. Each response after the first is scored once with a sliding window of up to 199 preceding responses. For questions with several KCs, probabilities are averaged without exposing same-question responses across KCs. Test learners with at least one scored response and their scored responses number 2,654 and 165,917 on MOOCRadar and 3,613 and 1,099,197 on XES3G5M (Table 8). Scoring keys are identical across all conditions and folds. KC-based methods read the history as KC rows, so their 199-row window holds fewer responses than the 199-response window of question-level models (Appendix A).

Experiment 1 (Section 4.1) compares EoKT with standard KT methods on the normal evaluation. It removes nothing beyond the fixed test learners, so almost every scored question is warm.

Experiment 2 (Section 4.2) removes the responses to a fixed 10% of the question vocabulary from training and validation and leaves the test set untouched. The withheld questions are cold at scoring time. It uses the same predictor, the same holdout, and ten runs from five folds and two training seeds, and compares cold AUC (Figure 2; Table 3).

Experiment 3 (Section 4.3) repeats Experiments 1 and 2 with two public embedding models in place of gemini-embedding-2. The predictor and the training procedure are shared, and only the input dimension of the projection follows the embedding. Both warm and cold evaluations are repeated. Both evaluations in Table 4 report mean $\pm$ sample SD over ten runs (five folds $\times$ two seeds), without validation seed selection.

Experiment 4 (Section 4.4) splits the XES3G5M cold questions of Experiment 2 by whether a variant differing only in its numbers remains on the training side. No model is retrained: the saved predictions of the ten EoKT, question-ID, and history-free text-control runs on the same holdout are re-aggregated by the strata defined in Appendix D.

Experiment 5 (Section 4.5) times one prediction with precomputed question representations and a retained state. It compares against recomputing a 200-response window.

Experiment 6 (Section 4.6) compares one EoKT trained jointly on three datasets with per-dataset models. Scoring uses the warm-dominated test sets, and the question is compatibility, not a gain from pooling or a transfer between datasets.

Standard AUC and cold AUC differ both in the scored questions and in the aggregation. Standard AUC is computed over all test responses of a model trained without a question holdout, so it is a pooled AUC over almost entirely warm responses. Cold AUC computes AUC separately for each learner over responses to withheld questions and averages equally over learners with both correct and incorrect cold responses; it is closer than a pooled AUC to how an adaptive system ranks candidate questions for one learner. Cold refers to questions withheld from model training and validation, not to new learners or necessarily to a learner's first encounter with a question. Because it ranks responses across time points, cold AUC also reflects changes in learner state. We chose it after the first MOOCRadar holdout comparison and preregistered it for every later confirmatory run (XES3G5M, public models, added holdouts). Normal and holdout training use separate models. Only Section 4.2 also reports within-learner warm AUC and pooled cold AUC of the same holdout-trained models, named explicitly, and we do not compare magnitudes across aggregation methods. Reference points for cold AUC on MOOCRadar are question-ID inputs, which cannot distinguish withheld questions (0.5047), and a learner's own running accuracy (0.4410; Appendix C).

Candidates share one fixed configuration, and validation AUC alone selects among them. Test scores never enter selection. Folds and validation follow the simpleKT protocol (Liu et al., 2023b); training seeds are 42 and 3407, plus 2026 for MOOCRadar EoKT, its version without the IRB, and their KC additions. For each fold, question-level validation AUC selects among one to three successful candidates, and ties go to the lower seed. EoKT was fixed at three IRB passes after exploring 17 configurations on MOOCRadar; the contribution of unequal search budgets to the reported differences is unquantified. That exploration used only learners outside the fixed test set of 2,794, which had been separated beforehand and was never inspected. The comparison conditions are not symmetric. The standard methods run with their public settings, and only AKT, IEKT, and the MOOCRadar simpleKT were additionally tuned on validation. Table 1's IEKT uses a setting selected on fold-0 validation (on XES3G5M, from 19 recorded candidates) and is 0.0038 below its published XES3G5M

AUC. EERNN-A has no time-gap input, its history model differs in capacity from that of EoKT, and on XES3G5M it stops under patience 5, not 10. The standard methods lack EoKT's time-gap input, whose removal Section 4.1 reports. None of the differences in Section 4 is adjusted for these asymmetries; they describe attainment under the stated conditions, not the superiority of one method. Appendix A gives the search records and stopping rules.

Intervals for paired differences come from resampling the original learners. Except in Table 7, which reports half-ranges, every $\pm$ is a sample SD over the runs stated for each table. Paired differences carry the percentile interval of 10,000 bootstrap draws of the original learners on identical scoring keys, conditional on the selected runs. For claims about withheld questions we also report a two-way bootstrap that resamples questions, and Appendix A lists what each interval conditions on.

4 RESULTS

4.1 EXPERIMENT 1: PREDICTIVE ACCURACY WITHOUT KC ANNOTATIONS

In standard AUC, EoKT without KC inputs reached the validation-selected KC-based method on both main datasets (Table 1). The paired difference is +0.0056 [+0.0048, +0.0063] on MOOCRadar and +0.0046 [+0.0042, +0.0050] on XES3G5M (Table 2). Validation selected IEKT in every fold, so this is also the difference to the best available KC-based method (0.9015 against 0.8959 and 0.8288 against 0.8242). Removing the time-gap MLP (8,736 parameters) lowers standard AUC by 0.0042 and 0.0045 on MOOCRadar and by 0.0013 and 0.0010 on XES3G5M (seeds 42 and 3407, five folds each). Without it, EoKT still reaches 0.8968/0.8967 and 0.8275, above the 0.8959 and 0.8242 of the validation-selected IEKT, although these are unselected seed means rather than validation-selected values (Appendix A). EERNN-A reaches 0.8949 and 0.8166, respectively, the latter under patience 5 rather than 10. Table 1 marks this stopping-rule difference and the unconverged XES3G5M LKT runs stopped at a 36-hour cap with †.

Table 1: Validation-selected standard AUC

Method	MOOCRadar	XES3G5M
simpleKT	0.8671 ± 0.0009	0.8157 ± 0.0004
IEKT	0.8959 ± 0.0003	0.8242 ± 0.0003
AKT	0.8601 ± 0.0004	0.8178 ± 0.0006
qDKT (ID only)	0.8903 ± 0.0007	0.8203 ± 0.0006
EERNN-A	0.8949 ± 0.0004	$0.8166 \pm 0.0013^{\dagger}$
LKT (KC + question text; Appendix G)	0.7505 ± 0.0025	$0.7898 \pm 0.0021^{\dagger}$
EoKT	0.9015 ± 0.0004	0.8288 ± 0.0003
Question-ID table + IRB (I3)	0.9015 ± 0.0004	0.8311 ± 0.0006
EoKT + learned KC	0.9005 ± 0.0004	0.8250 ± 0.0002
EoKT + pretrained KC	0.9015 ± 0.0002	0.8286 ± 0.0003

Replacing the text embeddings with a question-ID table inside the same predictor did not lower standard AUC. The ID condition (I3) keeps the IRB and the GRU and exchanges only the input. Its paired difference to EoKT is 0.0000 [-0.0005, +0.0005] on MOOCRadar and +0.0023 [+0.0021, +0.0026] on XES3G5M (Table 1). Permuted text embeddings, which break the link to the content, still reach 0.8993 on MOOCRadar. Replacing the IRB by an MLP of the same size moves AUC by 0.0005, and KC vectors alone as the question input reach 0.8979 (Appendix B).

Adding KC inputs to the same question representation did not raise standard AUC (Table 2). Learned KC inputs change standard AUC by -0.0010 [-0.0014, -0.0006] on MOOCRadar and -0.0037 [-0.0040, -0.0035] on XES3G5M. Pretrained KC-name embeddings change it by 0.0000 and -0.0001, with intervals that include zero. Without the IRB, the same two additions move AUC by at most 0.0002, so the result is not specific to the IRB (Appendix B, Table 14).

4.2 EXPERIMENT 2: PREDICTION ON QUESTIONS UNSEEN DURING TRAINING

EoKT had higher cold AUC than a permutation control on both datasets and than KC-reading methods on XES3G5M (Table 3; Figure 2, right). Figure 2, right, compares EoKT, question-ID

Table 2: Paired differences in standard AUC

Comparison (Δ AUC)	MOOCRadar	XES3G5M
EoKT − standard KT	+0.0056	+0.0046
EoKT: + learned KC	-0.0010	-0.0037
EoKT: + pretrained KC	0.0000	-0.0001

inputs, permuted text embeddings, and EERNN-A trained on the same holdout, and Table 3 gives the differences with intervals that resample learners and questions. Reassigning the same embeddings to other questions by a fixed permutation lowers cold AUC from 0.6041 to 0.5193 on MOOCRadar and from 0.6489 to 0.5656 on XES3G5M, and neither interval includes zero, supporting a contribution from correct text-to-question correspondence. Methods reading KC annotations (simpleKT, AKT, and a KC-only input) reach 0.5286–0.5535 on MOOCRadar and 0.6067–0.6230 on XES3G5M; all three gaps to EoKT exclude zero on XES3G5M but clearly only AKT's on MOOCRadar, whose holdout informed the metric choice, so that comparison is exploratory. EoKT and the KC-only control both use time gaps, whereas AKT and simpleKT do not. The KC-only control has no IRB, and the standard methods differ in inputs and search budgets, so these compare complete configurations. The question-ID table has no trained representation for a withheld question and cannot distinguish cold questions at a fixed learner state, so it is a structural reference rather than a competitor (0.5047 on MOOCRadar and 0.5896 on XES3G5M). EERNN-A (0.5950 and 0.6132) differs in inputs and capacity, and on XES3G5M in stopping rule.

Table 3: Cold AUC of EoKT minus each comparison. EoKT reaches 0.6041 on MOOCRadar and 0.6489 on XES3G5M. Intervals are two-way bootstrap 95% intervals that resample learners and questions; learner-only intervals are in Table 19. MOOCRadar, whose holdout informed the metric choice, is exploratory. — denotes no two-way interval or no run.

Comparison	MOOCRadar Δ	Two-way 95% CI	XES3G5M Δ	Two-way 95% CI
Permuted text embeddings	+0.0848	[+0.0050, +0.1696]	+0.0833	[+0.0520, +0.1169]
KC tags only (reads KCs)	+0.0626	[+0.00001, +0.1533]	+0.0259	[+0.0041, +0.0513]
simpleKT (reads KCs)	+0.0505	[-0.0198, +0.1253]	+0.0404	[+0.0184, +0.0653]
AKT (reads KCs)	+0.0754	[+0.0107, +0.1427]	+0.0423	[+0.0206, +0.0669]
EERNN-A	+0.0091	—	+0.0357	—
IRB replaced by an equal-size MLP	-0.0050	[-0.0153, +0.0058]	—	—
Question-ID table (structural reference)	+0.0994	[+0.0257, +0.1483]	+0.0593	[+0.0346, +0.0871]

Under the same holdout-trained models, the EoKT–ID difference is small in within-learner warm AUC and larger in cold AUC. EoKT minus the ID table is −0.0008 warm/+0.0994 cold on MOOCRadar and −0.0020/+0.0593 on XES3G5M (Table 20). Warm is not necessarily train-observed in every fold, and eligible learners differ between subsets; the contrast is descriptive.

EoKT also exceeded the history-free text control TEXTONLY on first encounters with cold questions. A first encounter is the first occurrence of a learner–question pair in the test sequence; responses to other questions may precede it. Cold AUC is 0.6048 for EoKT versus 0.5665 for TEXTONLY on MOOCRadar, and 0.6425 versus 0.5879 on XES3G5M (Appendix I, Table 26). Mean paired differences are +0.0383 and +0.0546; the paired difference is positive in each of the ten runs per dataset. The runs share test learners and are not independent samples. These are descriptive comparisons of configurations: capacity, IRB, and time-gap inputs also differ, so they do not isolate the effect of history.

Under pooled cold AUC the same comparison reverses: question-ID inputs reach 0.8393 and text 0.8381 (Table 16). Pooled cold AUC also ranks responses of different learners, and the ID condition still orders learners by their state, so learner-level differences dominate. On the two public models it does not reverse consistently: across three holdouts, 17 of 24 pooled differences to IDs are positive (−0.014 to +0.021; Table 18).

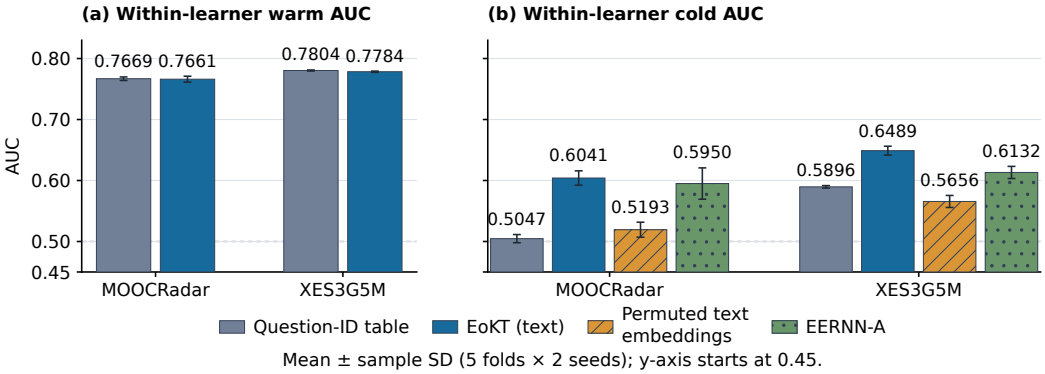

Figure 2: Within-learner warm AUC (left) and cold AUC (right), both macro-averaged from the same holdout-trained models (Table 20), with permuted text embeddings and EERNN-A added on cold responses (Table 19). The question-ID table cannot distinguish withheld questions at a fixed learner state and serves as a structural reference. Bars are means over ten runs (five folds $\times$ two seeds); error bars are sample SDs, not confidence intervals. The shared axis starts at 0.45. Warm denotes the holdout complement; eligible learners differ between subsets.

4.3 Experiment 3: Replacing the Embedding Model

Replacing the embedding model changed the results little: the three models gave similar AUC, and all of them exceeded question-ID inputs on cold questions (Table 4). Standard AUC ranges from 0.9002 to 0.9011 on MOOCRadar and from 0.8261 to 0.8286 on XES3G5M, and the ID condition is at least as high. Cold AUC on MOOCRadar is 0.6041 for gemini-embedding-2, 0.5938 for BGE-M3, and 0.5979 for multilingual-e5-large, against 0.5047 for IDs. On XES3G5M it is 0.6489, 0.6492, and 0.6462, against 0.5896. gemini-embedding-2's point estimate is higher on MOOCRadar, but each interval against the public models includes zero, and the rank flips across XES3G5M seeds. Across the three holdouts per dataset, the two public models exceed IDs in all 24 conditions with positive individual intervals (Appendix C).

Table 4: Embedding-model comparison

Dataset	Model	Standard AUC	Cold AUC
MOOCRadar	gemini-embedding-2	0.9011 ± 0.0004	0.6041 ± 0.0118
MOOCRadar	BGE-M3	0.9005 ± 0.0004	0.5938 ± 0.0106
MOOCRadar	multilingual-e5-large	0.9002 ± 0.0004	0.5979 ± 0.0160
MOOCRadar	Question-ID table (I3)	0.9013 ± 0.0004	0.5047 ± 0.0067
XES3G5M	gemini-embedding-2	0.8286 ± 0.0004	0.6489 ± 0.0073
XES3G5M	BGE-M3	0.8285 ± 0.0004	0.6492 ± 0.0041
XES3G5M	multilingual-e5-large	0.8261 ± 0.0007	0.6462 ± 0.0035
XES3G5M	Question-ID table (I3)	0.8308 ± 0.0006	0.5896 ± 0.0020

4.4 Experiment 4: Questions That Differ Only in Their Numbers

Text embeddings gave higher cold AUC than question-ID inputs both on cold questions with a training-side variant that differs only in its numbers and on those without one. Of the 693 scored held-out XES3G5M questions (fold mean), 51 have no image and a numeric variant in training, with 906 eligible learners (Table 5; Appendix D). On them, cold AUC is 0.6612 for EoKT and 0.6030 for I3, a difference of +0.0582 [+0.0357, +0.0804]. On the 543 questions with no variant or identical text in training, it is +0.0565 [+0.0501, +0.0630]. Their difference, +0.0017 [−0.0214, +0.0249], includes zero, so the effect of a variant is undetermined at this scale.

The advantage of text holds on cold questions with no training-side match after whitespace removal and number replacement, and tends to be larger for textually closer ones. Splitting those still unmatched after unifying notation into thirds by character-trigram similarity to the nearest training question gives +0.0331 [+0.0218, +0.0447] in the most distant third and +0.0969 [+0.0857, +0.1081]

Table 5: Examples of numeric variants from XES3G5M

Held-out (cold) question	Training-side variant	Solution check
An orchard has 38 more pear trees than apple trees and 3 times as many pear trees as apple trees. How many of each?	Same with 50 more and 6 times.	Same: difference / (multiple − 1)
Maximum intersections of 15 lines in a plane?	Same with 8 lines.	Same: $1 + 2 + \ldots + (n - 1)$
Compute 17×99.	Compute 37×101.	Same: distributive law near 100
How many numbers without repeated digits can be formed from 1, 3, 6?	Same with 0, 2, 3.	May change: 0 cannot lead

in the closest. Where an identical text is in training (91 questions), it is +0.0924 [+0.0769, +0.1074]. With questions also resampled, the distant-third interval and the between-stratum gaps include zero, so these patterns are undetermined under question-level variability (Appendix D).

4.5 Experiment 5: Inference Speed

EoKT returns one prediction in about 0.50 ms on a single AWS CPU thread. Table 6 reports inference latency on AWS m6i.large with one CPU thread and batch size 1. It includes measurements taken at different times; Appendix F gives the variation across measurements and detailed conditions. IEKT uses the XES3G5M width-128 configuration. Incremental paths for simpleKT and AKT are not measured (—); window recomputation is the reference implementation path, not an inherent model constraint. Weights and inputs are random. Question representations are precomputed; incremental EoKT updates its retained 512-dimensional GRU state by one step. Embedding a new question is not timed. Re-scoring all test responses with trained EoKT through the incremental path reproduces sliding-window AUC to four decimals on both datasets (Table 23, Appendix F).

Table 6: Inference latency of the main methods (ms)

Model	Window 200	Incremental
EoKT	65.13	0.50
simpleKT	5.83	—
AKT	24.12	—
IEKT (width 128)	312.96	1.482

4.6 Experiment 6: Generality of One Model across Several Datasets

Pooled and separate training gave numerically close standard AUCs in this check. The EoKT pooled over the three datasets stays within 0.0012 of the per-dataset models on every dataset (Table 7). Table 7 is a separate fold-0 check that compares EoKT trained per dataset with one EoKT trained jointly on all three datasets, using dataset-prefixed question vocabularies (Appendix E).

Table 7: Per-dataset versus three-dataset pooled training

Dataset	Single dataset	Three-dataset pooled	Pooled − single
MOOCRadar	0.9012	0.9006 ± 0.0005	-0.0006
XES3G5M	0.8288	0.8277 ± 0.0007	-0.0012
NIPS34	0.8065	0.8058 ± 0.0002	-0.0007

5 Discussion

EoKT lets a trained predictor score an added question from its text and a learner's existing response history. It requires neither KC annotations nor a fitted ID vector for each added question, shifting the work of adding questions to text preparation and embedding generation. This design is useful for question banks that change over time. A common embedding input also supports training across datasets without changing the architecture (Section 4.6). Adding a question to an existing predictor differs from adding a dataset to training: the latter still requires response data from that dataset.

The clearest evidence for question-text inputs comes from within-learner prediction on cold questions. ID inputs match or exceed text in standard AUC, whereas all three embedding models outperform the ID table in cold AUC, and permuting text-to-question correspondence reduces accuracy. EoKT also exceeds history-free TEXTONLY on first encounters (Section 4.2). Standard KT methods differ in inputs, search budgets, and effective history lengths; TEXTONLY also differs in capacity, IRB, and time-gap inputs. These comparisons do not isolate causal effects of history or text. The reversal in pooled cold AUC and gains from adding KCs in some cold settings restrict the claim to within-learner discrimination, not general redundancy of KC information.

Automatic variant generation offers a use case for scoring candidate questions in the context of a learner's history. Cold prediction on observed numeric variants (Section 4.4) supports a design that scores candidates from text without a trained question-ID representation. Yet changing numbers can change solution steps or difficulty. Re-scoring LLM rewordings yielded an AUC-difference interval containing zero, whereas notation changes reduced AUC (Appendix D). This tests input sensitivity using the original observed labels; it does not evaluate responses to generated questions or establish matched difficulty. The next step is to collect responses to generated questions and evaluate within-learner discrimination and probability calibration. This would test whether scoring generated candidates can support selecting questions appropriate for a learner.

Limitations. The evaluation scope and data provenance impose further limitations. Cold denotes questions withheld from predictor training, not a learner's first encounter. Only four MOOCRadar learners qualify for the repeat-encounter analysis (Appendix I). Evaluation content may appear in embedding pretraining data, so indirect leakage remains possible. Both main datasets are Chinese; NIPS34 is auxiliary to Experiment 6, and transfer to unseen subjects is untested. Logs without question text (e.g., EdNet) are out of scope, and missing or degraded text is untested. Effective-history controls cover MOOCRadar but not XES3G5M. For the textually most distant third of questions, the margin over ID inputs includes zero when questions are also resampled (Section 4.4). EoKT provides neither KC-level diagnosis nor concept explanations by design; other-subject histories and actual learning outcomes remain untested.

6 Conclusion

EoKT predicts responses from fixed question-text embeddings and a response history, without KC annotations or a question-ID table. In cold AUC (fixed after the first MOOCRadar holdout), gemini-embedding-2 embeddings lead three KC-reading methods on XES3G5M under stated conditions and permuted embeddings on both datasets; three embedding models lead the ID table, though not consistently in pooled cold AUC. On warm questions EoKT, like an ID table, reaches the KC-based methods. Fixed text embeddings support prediction without manual KC annotation on these two datasets, including withheld questions, though less for textually distant ones. EoKT is thus a fast predictor that requires no KC annotations, and its common embedding input supports training on additional subjects without changing the model architecture.

Reproducibility Statement

The anonymous supplement contains reproduction code and aggregate results. It is attached to the submission as supplementary material on OpenReview. Its README.md and PAPER_MAP.md map each table and figure to the corresponding records and code and, where the recorded values allow it, to the command that regenerates or checks it. REPRODUCE.md documents data acquisition, preprocessing, embedding generation, the per-fold runs, verification of the scoring targets, and aggregation in the reader's own environment; optional helper scripts for cloud execution are also included. Except for the permuted-text-embedding control in Table 15, the archive contains the per-run metrics behind the reported tables, including the additional AKT and IEKT confirmations and the XES3G5M time-gap results in Appendix A, the open-embedding comparison and additional question holdouts in Appendix C, and the history-free control in Appendix I. For the numeric-variant analysis in Section 4.4 and Appendix D, it contains the per-question stratum assignments and the computed intervals together with the analysis code, but not the rewritten question texts. It also contains split manifests, evaluation-window and scoring-key rules, selected-run settings, and paired intervals, and bundled scripts recompute the tables from these metrics. Per-response predictions and per-learner bootstrap draws are not included, so paired intervals are provided as computed. Raw response histories, question text, embeddings, and weights are excluded in accordance with source-data distribution terms; retraining therefore regenerates the embeddings, and bit-identical results are not expected. Short picture-based explainers of the paper in English, Simplified Chinese, and Japanese are also included.

Ethics Statement

This study evaluates correctness prediction on public educational datasets. We conduct no learner intervention and include no identifiable learner response information in the supplement. Each source dataset remains subject to its distribution terms.

Statement on AI Usage

Human authors originated the research questions and model architecture, and developed experimental plans through dialogue with AI. Claude Code was used primarily to implement EoKT, convert data, execute training and evaluation jobs, and aggregate results. Codex also supported code maintenance, re-aggregation of saved results, candidate-run audits, and checks between numerical results and figures or tables. Under author direction, these agents generated and revised code and analysis procedures and assisted execution and validation. gemini-embedding-2 generated question-text embeddings. Gemini was used to transcribe question images for supplementary experiments and translate example questions into English. Claude Code translated the variant examples in Table 5 and checked the 30 variant pairs of Experiment 4 against worked solutions. Gemini generated the reworded questions in Appendix D. Section 2 and Appendix A specify the embedding inputs and their fixed status during KT training.

AI also supported prose generation, bilingual manuscript preparation, and revision. The authors specified paragraph claims and generation instructions in Paper Design Markup Language (PDML), a markup language the authors defined for manuscript writing, from which AI generated prose. Coding agents also assisted Japanese–English wording, generation of outputs containing tables, figures, and equations, and consistency checks. AI-assisted literature search and summarization were used to discover and screen candidate related work; the authors read the original sources and verified every cited work. Simulated reviews with Codex supported checks for inconsistencies and the scope of claims and suggested revisions; they are not human conference peer reviews. AI output itself is not evidence for experimental results or their validity: numerical claims are grounded in saved execution results and re-aggregation. Human authors retain responsibility for the scientific claims, experimental design, interpretation, final wording, and accuracy of the submitted manuscript.

References

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu (2024). BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through

Self-Knowledge Distillation. arXiv:2402.03216. https://arxiv.org/abs/2402.03216

Zhi Cheng and Jinlong Li (2025). Exercise Semantic Embedding for Knowledge Tracking in Open Domain. Information, 16(4):302. https://doi.org/10.3390/info16040302

Zhangqi Duan, Nigel Fernandez, Arun Balajiee Lekshmi Narayanan, Mohammad Hassany, Rafaella Sampaio de Alencar, Peter Brusilovsky, Bita Akram, and Andrew Lan (2026). Automated Knowledge Component Generation for Interpretable Knowledge Tracing in Coding Problems. Findings of the Association for Computational Linguistics: ACL 2026. arXiv:2502.18632v4. https://arxiv.org/abs/2502.18632v4

Lingyue Fu, Hao Guan, Kounianhua Du, Jianghao Lin, Wei Xia, Weinan Zhang, Ruiming Tang, Yasheng Wang, and Yong Yu (2024). SINKT: A Structure-Aware Inductive Knowledge Tracing Model with Large Language Model. Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. https://arxiv.org/abs/2407.01245

Rao Fu, Zixuan Yang, Jiankun Zhang, Jing Ma, Hechang Chen, Yu Li, and Yi Chang (2026). Simply Stabilizing the Loop via Fully Looped Transformer. arXiv:2605.18797v2. https://arxiv.org/abs/2605.18797v2

Aritra Ghosh, Neil Heffernan, and Andrew S. Lan (2020). Context-Aware Attentive Knowledge Tracing. Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2330–2339. https://doi.org/10.1145/3394486.3403282

Yuxiang Guo, Shuanghong Shen, Qi Liu, Zhenya Huang, Linbo Zhu, Yu Su, and Enhong Chen (2024). Mitigating Cold-Start Problems in Knowledge Tracing with Large Language Models: An Attribute-aware Approach. Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24), 727–736. https://doi.org/10.1145/3627673.3679664

Zhongfeng Jia, Wei Su, Jiamin Liu, and Wenli Yue (2023). Attentive Q-Matrix Learning for Knowledge Tracing. arXiv:2304.08168v2. https://arxiv.org/abs/2304.08168v2

Heeseok Jung, Yohaan Yoon, Jaesang Yoo, and Yeonju Jang (2025). CLST: Cold-Start Mitigation in Knowledge Tracing by Aligning a Generative Language Model as a Students' Knowledge Tracer. Journal of Educational Data Mining, 17(2):86–117. https://jedm.educationaldatamining.org/index.php/JEDM/article/view/854

Arisha Khan, Nathaniel Li, Tori Shen, and Anna N. Rafferty (2025). Just Read the Question: Enabling Generalization to New Assessment Items with Text Awareness. Proceedings of the 18th International Conference on Educational Data Mining (EDM 2025), Poster and Demo Papers. https://educationaldatamining.org/EDM2025/proceedings/2025.EDM.poster-demo-papers.274/index.html

Unggi Lee, Jiyeong Bae, Dohee Kim, Sookbun Lee, Jaekwon Park, Taekyung Ahn, Gunho Lee, Damji Stratton, and Hyeoncheol Kim (2024). Language Model Can Do Knowledge Tracing: Simple but Effective Method to Integrate Language Model and Knowledge Tracing Task. arXiv:2406.02893v2. https://arxiv.org/abs/2406.02893v2

Wonbeen Lee, Channyoung Lee, Junho Sohn, and Hansam Cho (2025). PICKT: Practical Interlinked Concept Knowledge Tracing for Personalized Learning using Knowledge Map Concept Relations. arXiv:2512.07179v1. https://arxiv.org/abs/2512.07179v1

Qi Liu, Zhenya Huang, Yu Yin, Enhong Chen, Hui Xiong, Yu Su, and Guoping Hu (2021). EKT: Exercise-Aware Knowledge Tracing for Student Performance Prediction. IEEE Transactions on Knowledge and Data Engineering, 33(1):100–115. https://doi.org/10.1109/TKDE.2019.2924374

Zitao Liu, Qiongqiong Liu, Jiahao Chen, Shuyan Huang, Jiliang Tang, and Weiqi Luo (2022). pyKT: A Python Library to Benchmark Deep Learning based Knowledge Tracing Models. Advances in Neural Information Processing Systems 35, Datasets and Benchmarks Track. https://arxiv.org/abs/2206.11460

Zitao Liu, Qiongqiong Liu, Teng Guo, Jiahao Chen, Shuyan Huang, Xiangyu Zhao, Jiliang Tang, Weiqi Luo, and Jian Weng (2023). XES3G5M: A Knowledge Tracing Benchmark Dataset with Auxiliary Information. NeurIPS 36, Datasets and Benchmarks. https://proceedings.neurips.cc/paper_files/paper/2023/hash/67fc628f17c2ad53621fb961c6bafcaf-Abstract-Datasets_and_Benchmarks.html

Zitao Liu, Qiongqiong Liu, Jiahao Chen, Shuyan Huang, and Weiqi Luo (2023). simpleKT: A Simple But Tough-to-Beat Baseline for Knowledge Tracing. ICLR 2023. https://arxiv.org/abs/2302.06881

Ting Long, Yunfei Liu, Jian Shen, Weinan Zhang, and Yong Yu (2021). Tracing Knowledge State with Individual Cognition and Acquisition Estimation. Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, 173–182. https://doi.org/10.1145/3404835.3462827

Max Norris, Kobi Gal, and Sahan Bulathwela (2025). Next Token Knowledge Tracing: Exploiting Pretrained LLM Representations to Decode Student Behaviour. arXiv:2511.02599 (v2, 2026-01-26). https://arxiv.org/abs/2511.02599

Yilmazcan Ozyurt, Stefan Feuerriegel, and Mrinmaya Sachan (2024). Automated Knowledge Concept Annotation and Question Representation Learning for Knowledge Tracing. arXiv:2410.01727v2 (revised March 2025). https://arxiv.org/abs/2410.01727v2

Shalini Pandey and Jaideep Srivastava (2020). RKT: Relation-Aware Self-Attention for Knowledge Tracing. Proceedings of the 29th ACM International Conference on Information & Knowledge Management, 1205–1214. https://doi.org/10.1145/3340531.3411994

Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J. Guibas, and Jascha Sohl-Dickstein (2015). Deep Knowledge Tracing. Advances in Neural Information Processing Systems 28, 505–513. https://arxiv.org/abs/1506.05908

Shashank Sonkar, Andrew E. Waters, Andrew S. Lan, Phillip J. Grimaldi, and Richard G. Baraniuk (2020). qDKT: Question-centric Deep Knowledge Tracing. Proceedings of the 13th International Conference on Educational Data Mining, 677–681. https://arxiv.org/abs/2005.12442

Yu Su, Qingwen Liu, Qi Liu, Zhenya Huang, Yu Yin, Enhong Chen, Chris Ding, Si Wei, and Guoping Hu (2018). Exercise-Enhanced Sequential Modeling for Student Performance Prediction. Proceedings of the AAAI Conference on Artificial Intelligence, 32(1). https://ojs.aaai.org/index.php/AAAI/article/view/11864

Ryota Ushio, Takashi Ishida, and Masashi Sugiyama (2026). Bayes-Optimal BER and AUC: Estimation and Evaluation of Estimators. arXiv preprint, September 2026. https://arxiv.org/abs/2609.02304

Zichao Wang, Angus Lamb, Evgeny Saveliev, Pashmina Cameron, Jordan Zaykov, Jose Miguel Hernandez-Lobato, Richard E. Turner, Richard G. Baraniuk, Craig Barton, Simon Peyton Jones, Simon Woodhead, and Cheng Zhang (2021). Results and Insights from Diagnostic Questions: The NeurIPS 2020 Education Challenge. PMLR 133, 191–205. https://proceedings.mlr.press/v133/wang21a.html

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei (2024). Multilingual E5 Text Embeddings: A Technical Report. arXiv:2402.05672. https://arxiv.org/abs/2402.05672

Ziwei Wang, Jie Zhou, Qin Chen, Bo Jiang, Qingchun Bai, Liang Dou, and Liang He (2026). LLM-KT: Enhancing Large Language Models with Knowledge Tracing via Multi-Level Plug-and-Play Alignment. Findings of the Association for Computational Linguistics: ACL 2026, 35775–35792. https://aclanthology.org/2026.findings-acl.1781/

Yumou Wei, Paulo Carvalho, and John Stamper (2025). KCluster: An LLM-based Clustering Approach to Knowledge Component Discovery. Proceedings of the 18th International Conference on Educational Data Mining. https://educationaldatamining.org/EDM2025/proceedings/2025.EDM.long-papers.64/index.html

Liu Yang, Kangwook Lee, Robert Nowak, and Dimitris Papailiopoulos (2024). Looped Transformers are Better at Learning Learning Algorithms. ICLR 2024. https://arxiv.org/abs/2311.12424

Jifan Yu, Mengying Lu, Qingyang Zhong, Zijun Yao, Shangqing Tu, Zhengshan Liao, Xiaoya Li, Manli Li, Lei Hou, Hai-Tao Zheng, Juanzi Li, and Jie Tang (2023). MoocRadar: A Fine-grained and Multi-aspect Knowledge Repository for Improving Cognitive Student Modeling in MOOCs. SIGIR 2023. https://arxiv.org/abs/2304.02205

Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung (2017). Dynamic Key-Value Memory Networks for Knowledge Tracing. Proceedings of the 26th International Conference on World Wide Web, 765–774. https://doi.org/10.1145/3038912.3052580

A Evaluation Details

Text and KC embeddings are fixed from saved input assets and are never regenerated or updated. Text inputs are the title, content, and options on MOOCRadar, the content and options on XES3G5M, and the stem, choices, and figure description obtained by optical character recognition on NIPS34. Correct answers, explanations, and KC fields are never concatenated. The saved model identifier is gemini-embedding-2@vertex, 3072 dimensions, L2-normalized, with no task specification. KC names are embedded with the same model. Generation-time hashes of the complete API inputs and the service revision were not saved; the reproduction material records the asset hashes and the generation code. NIPS34 is a third dataset with Level 3 diagnostic-mathematics subjects as KCs and official pyKT partitions (Wang et al., 2021; Liu et al., 2022). Its AUC and readout scores informed the choice of the EoKT configuration, so its results are not an independent confirmation. It is excluded from the main evaluation and appears only in the pooled-training check (Table 7, Appendix E) and in the appendix tables that share the same aggregation.

The scoring targets are aligned to a single set across all conditions. Conditions that process question sequences and the KC-expanded simpleKT and AKT drop short learners by different units. The former counts questions and the latter counts expanded KC rows, so the KC-expanded scoring targets form a superset of the question-level ones. The aggregator takes the common set as the reference, removes the surplus, and records the removed count in its verification report. On MOOCRadar this surplus is 70 responses from 70 learners with exactly two responses. A window of 200 on the KC-expanded side counts KC rows, so the same 200 reaches fewer past questions than on the question-level side. In an audit of the MOOCRadar inputs, full 200-KC-row windows contained a median of 83 response events in training/validation and 82 in testing. After grouping KCs of each response and counting each response once, the 165,987 test responses before common-set alignment had a median available history of 53 preceding responses (10th/90th percentiles: 7/115). These window statistics do not identify the cause of an AUC difference. The pyKT-native KC-level validation AUC is not a performance figure, because the second and later rows of a multi-KC question expose the response already observed in the first row. Configuration and seed selection use question-level validation AUC, whereas AKT checkpoint selection and early stopping follow pyKT KC-level validation AUC. These are distinct selection stages.

The two evaluation windows differ by an almost constant amount across methods (Table 9). We evaluated checkpoints saved under the All-in-One protocol (XES3G5M, fold 0, seed 42) without changing their weights, both with non-overlapping 200-response chunks (All-in-One) and with the sliding window. The All-in-One evaluation matches the aggregate recorded with these checkpoints to at least four digits, which confirms that our evaluator reproduces the All-in-One path. The gain of the sliding window ranges from +0.0023 to +0.0026 across EoKT, qDKT, and IEKT, so the change of window does not favour a particular method. The scoring targets grow from 1,094,990 to 1,099,197, and the additional positions are those that had no history at a non-overlapping chunk boundary. This diagnostic is kept separate from the five-fold results.

Table 8: Evaluation populations (training questions: those observed in fold 0)

Dataset	Test learners	Scored responses	Correct rate	Train questions
MOOCRadar	2,654	165,917	0.8140	1,959
XES3G5M	3,613	1,099,197	0.7956	7,565
NIPS34	983	271,833	0.5496	948

Table 9: Evaluation-window diagnostic on XES3G5M

Method	All-in-One	Sliding window	Difference
EoKT	0.8263	0.8289	+0.0026
qDKT (ID only)	0.8172	0.8199	+0.0026
IEKT	0.8153	0.8176	+0.0023

The distance from a published value of the same method measures how closely this implementation reproduces the published setup. Table 10 lists only methods trained here that also have a published value, not the published values used as comparison targets. The XES3G5M dataset paper reports qDKT, and our qDKT, trained on the same partition and scored the same way, supplies the check. On XES3G5M, our ID-only qDKT implementation is 0.0022 AUC below the published qDKT value. This difference describes the reproduction gap for this implementation; it does not establish the direction of bias in comparisons involving EoKT and other methods. For IEKT, the gap is 0.0086 under the published setting and 0.0038 under the setting selected on validation alone. The simpleKT paper does not report qDKT on NIPS34, so the same check is unavailable there. On XES3G5M both sides use the sliding window, whereas the NIPS34 rows set published All-in-One values against our sliding-window scores, so they are descriptive, not matched reproduction checks. NIPS34 AKT is omitted because its All-in-One comparison targets have not been verified, and the validation-selected XES3G5M AKT is given in the attention-baseline paragraph below. The differences in Table 10 carry no paired intervals. The validation-selected XES3G5M IEKT uses dropout 0.3, gamma 0.8, and embedding size 128, selected from nineteen candidates using fold-0 validation alone. Its fold SD is 0.0003 (0.0017 under the published setting), the order of the published 0.0002. This table reports reproduction gaps for all methods with available published values, including IEKT. Section 3.1 of the dataset paper specifies two embedding sizes, three learning rates, and four dropout rates (24 combinations), whereas the toolkit guide describes over two hundred trials with stagnation-based stopping. The actual trial count and stopping split cannot be determined. Repeating settings and retaining the maximum can introduce selection optimism, so the residual gap is not attributed to method quality. Both the published and validation-selected settings are recorded separately in the IEKT search record of the supplement.

Table 10: Our AUC minus published AUC

Dataset	Method	Published AUC	Our AUC	Difference
XES3G5M	simpleKT	0.8163	0.8157	-0.0006
XES3G5M	AKT (published setting)	0.8207	0.8166	-0.0041
XES3G5M	IEKT (published setting)	0.8280	0.8194	-0.0086
XES3G5M	IEKT (validation-selected)	0.8280	0.8242	-0.0038
XES3G5M	qDKT (ID only)	0.8225	0.8203	-0.0022
NIPS34	simpleKT	0.8035	0.8033	-0.0002
NIPS34	IEKT (published setting)	0.8045	0.8016	-0.0029

The following are the published settings of the public implementations; the settings of the validation-selected main-table AKT are given below. simpleKT uses width 128, two blocks, eight heads, feedforward width 256, and dropout 0.3. IEKT uses width 64, ten cognition and ten acquisition levels, gamma 0.93, and policy-loss weight 40. AKT uses width 256, one block, eight heads, feedforward width 256, final width 512, dropout 0.05, and Rasch regularization 0.00001. qDKT uses width 300 and dropout 0.2; this ID-only implementation uses neither fastText initialization from response sequences nor KC-derived regularization. EoKT uses a one-layer 512-dimensional GRU, dropout 0.1 on the GRU and the readout, and fixed 3072-dimensional text embeddings. The additional MOOCRadar conditions use the AKT setting (width 256, two blocks), the simpleKT setting (dropout 0.5), and the IEKT setting chosen by validation-only searches.

The IRB is a small attention block applied three times with shared weights. The projected 512-dimensional input is split into two 256-dimensional halves, and a shared projection maps each half to a 128-dimensional token. Splitting one vector into two halves follows the sequence construction of the Fully Looped Transformer and gives the shortest sequence on which self-attention can exchange information between tokens; the first pass uses self-attention (Fu et al., 2026; Yang et al., 2024). Later passes use the final state of the previous pass as the Query at every layer, while Keys and Values come from the current layer input. Each pass reuses the same three-layer block, with two heads, feed-forward width 1024, and dropout 0.2. Let H_0 denote the two initial tokens and H_r the tokens after pass r ($r = 1, 2, 3$). The difference between the token means of H_3 and H_0 is projected back to 512 dimensions by W_{up} and added to the input (Eq. 4). The next paragraph records how three passes were selected.

$$v_q = Pe_q + W_{\mathrm{up}}(\mathrm{mean}(H_3) - \mathrm{mean}(H_0)) \tag{4}$$

The provenance of every configuration is recorded separately from the measurements in these tables. The EoKT configuration was fixed at three passes after exploring 17 conditions on MOOCRadar and comparing pass count and capacity on external datasets. This records a design choice, not a controlled demonstration that three passes are optimal. Runs for the main comparison use at most 200 epochs, patience 10, an improvement threshold of 0.001, batch size 64, and Adam. The additional MOOCRadar AKT condition uses the width 256 and two blocks selected by a validation search. The additional MOOCRadar IEKT condition was chosen on fold-0 validation. Its protocol lists eight planned settings; this is not a verified count of completed trials. Neither decision consults the test set, and comparisons with the published AKT and IEKT settings are kept in the research record. The EoKT configuration has a search history that used NIPS34 AUC and AP, so the NIPS34 evaluation in this paper is not an independent confirmation. The search histories differ across methods, and we state this as part of the scope of the comparison. None of these searches is an exhaustive grid or Bayesian optimization, and none is a budget-matched comparison across methods. Among the ten XES3G5M IEKT runs, the selected run of fold 0 (seed 42) is a search-stage run whose training was cut at 18 epochs by a 12-hour wall-clock cap; its selected checkpoint holds the epoch-14 weights, before early stopping was reached. It is the only one of these ten IEKT runs cut before early stopping; its score after training to early stopping was not observed. The simpleKT/AKT screen described below is separate from this history. The two-block AKT condition above is distinct from the MOOCRadar learning-rate candidate of that screen, which is evaluated in the five-fold, two-seed AKT confirmation (below); the MOOCRadar simpleKT candidate of that screen is confirmed in the same way. The ten XES3G5M runs of EERNN-A are trained under patience 5, an improvement threshold of 0.001, and at most 60 epochs. All ten stopped by the rule (patience 5) after 22 to 35 epochs, with selected checkpoints at epochs 17 to 30; none reached 60 epochs or a training-time budget. This rule stops earlier than that of EoKT (patience 10, at most 200 epochs), so the two are not compared under the same stopping rule. On the same splits, EoKT reached its selected checkpoint at 11 to 15 epochs and qDKT at 2 to 3 epochs, and both then stopped under patience 10. Appendix G compares these runs with runs of the same cells capped at 30 epochs. Evaluation uses the same window evaluation as every other condition. The configuration is that of Appendix G, without hyperparameter search.

A separate fixed-configuration confirmation of IEKT reached early stopping in every run on both datasets. We fixed grid candidates chosen by validation search and trained them on five folds per dataset with seeds 42 and 3407, plus seed 2026 on MOOCRadar. Each run used one Apple M2 Max CPU thread, Adam, batch size 256, at most 200 epochs, patience 10, and an improvement threshold of 0.001. Both datasets used embedding width 128, learning rate 0.001, gamma 0.8, and policy-loss weight 40; dropout was 0 on MOOCRadar and 0.3 on XES3G5M. This confirmation excludes the Bayesian-search candidate. All 25 runs ended by early stopping, after 23–42 epochs on MOOCRadar and 23–29 on XES3G5M.

Pooled warm AUC in this confirmation averaged 0.8955 on MOOCRadar and 0.8227 on XES3G5M across seeds. Per seed, the five-fold means and sample SDs are 0.8956 $\pm$ 0.0011 (seed 42), 0.8955 $\pm$ 0.0008 (3407), and 0.8954 $\pm$ 0.0007 (2026) on MOOCRadar, and 0.8225 $\pm$ 0.0002 (42) and 0.8230 $\pm$ 0.0006 (3407) on XES3G5M; the sample SDs of the seed means are 0.0001 (three seeds) and 0.0004 (two seeds). We used the main comparison's sliding-window evaluation over all scored responses after normal training without question holdout. We recomputed AUC from saved predictions and verified artifact hashes. Every MOOCRadar run finished within 12 hours for seed 42 and within 18 hours for seeds 3407 and 2026. The XES3G5M runs still training at their snapshot time (all five seed-42 folds and two seed-3407 folds) used the same checkpoint and gave identical AUC in the snapshot and the final evaluation. The seed-3407 limit was extended from the preregistered 18 hours to 23 hours while these two runs were training; their selected checkpoints matched the 18-hour snapshot, so the reported values are unchanged. The snapshots are an operational diagnostic, not the preregistered endpoint. Batch size, execution environment, and seed selection differ from Table 1; MOOCRadar hyperparameters also differ. These results therefore do not replace Table 1 or isolate the effect of the time limit. Folds share the test set, so the SDs are not confidence intervals. This supplementary check reuses the test set, and the paired intervals of the main comparison do not apply to these runs.

Search counts and stopping conditions differ across methods and datasets; budgets are not matched. The 17 EoKT configurations are a development-history count; we have not reconstructed completion counts per candidate. Separate from the MOOCRadar IEKT plan, the XES3G5M IEKT search records 19 trials including the published setting: 14 completed, two were truncated, and three stopped by interim racing. The fold-0, seed-42 screens include 13/12/13 simpleKT settings on MOOCRadar/XES3G5M/NIPS34 (all completed) and 11/13/13 AKT settings (two XES3G5M results missing), including the published settings. These counts do not represent the same unit of compute budget as the development search. Selection and five-fold confirmation are described below. Without a budget-matched search, the contribution of search-history differences to the main AUC differences remains unquantified.

We compare a time-gap module ablation under matched conditions. On MOOCRadar, we train EoKT with and without the time-gap MLP for 20 runs (five folds $\times$ seeds 42/3407 $\times$ two conditions), keeping data, training settings, and the code bundle fixed. Removing the $1\rightarrow16\rightarrow512$ MLP reduces trainable parameters by 8,736, from 4,383,650 to 4,374,914; other layers retain identical initial weights and random states. For seed 42, pooled warm AUC is 0.9010 $\pm$ 0.0006 with the MLP and 0.8968 $\pm$ 0.0002 without it; the mean paired fold difference (without minus with) is -0.0042. For seed 3407, the corresponding values are 0.9011 $\pm$ 0.0002, 0.8967 $\pm$ 0.0006, and -0.0045. The $\pm$ values are sample SDs across five folds per seed: descriptive training variation on a shared test set, not confidence intervals. Neither seed is selected. AUC pools all scored responses and is dominated by warm questions; it is not a pure warm evaluation. The effect of time-gap input under cold conditions remains untested. This module ablation removes both time-gap input and a small amount of capacity, so their effects cannot be separated here. After a Spot interruption, runs resumed on another L4 of the same type, with each fold/seed pair completed on one instance. We verified matching settings, major environments, code, scoring keys, labels, and masks, and hashes of saved artifacts.

Removing the time-gap MLP also lowers pooled warm AUC on XES3G5M. We also trained normal-training F3 on XES3G5M with and without time-gap input for 20 runs (five folds $\times$ seeds 42/3407 $\times$ two conditions). All runs used one code bundle and one L4; we audited data, training settings, environments, scoring keys, labels, and masks. The parameter difference is again 8,736. For seed 42, pooled warm AUC is 0.8288 $\pm$ 0.0003 with the MLP and 0.8275 $\pm$ 0.0006 without it, with a paired fold difference (without minus with) of -0.0013. For seed 3407, the values are 0.8286 $\pm$ 0.0005, 0.8275 $\pm$ 0.0005, and -0.0010. The $\pm$ values are sample SDs across five folds per seed, not confidence intervals. These controls pool all test responses after training without question holdout and retain both seeds; they differ from the validation-selected values in Table 1. The test set is shared with the other experiments, so this is not an independent confirmation. Removing the module changes time-gap input and capacity together. Its AUC decrease cannot be subtracted from the main difference of +0.0046 to obtain a causal residual, or extrapolated to cold questions.

The saved settings and numerical tables track implementation and asset provenance. The anonymous archive contains the training, comparison, and probe code; per-fold settings and aggregates; the rules for the evaluation window and the scoring key; asset manifests; and acquisition and generation instructions. For each selected run covered by that archive we store the hashes of its configuration, checkpoint, and predictions together with the validation AUC of the seed that was not selected. MOOCRadar documents an expert verification step for its annotations. XES3G5M and NIPS34 provide existing KC assignments whose annotators and process were not established here. Training times come from mixed hardware and are therefore not used to compare the speed of the methods.

We summarize dataset coverage and the scope of the evaluation. MOOCRadar uses our partition; NIPS34 informed development, and XES3G5M IEKT/published-setting AKT reproduction remains unresolved. The simpleKT candidate from the attention-baseline screen is confirmed on MOOCRadar; on XES3G5M and NIPS34 the screen retained the published setting, so no confirmation stage exists (below). Short texts, three artificial holdouts per dataset for public inputs versus I3 (gemini-embedding-2 comparisons retain the original one), and uneven natural-cold populations limit coverage. Long passages and textless benchmarks are untested. Public inputs differ in dimensions and parameter counts, with interactions with time-gap inputs untested (Appendix C). Candidate failures are recorded in the research record. Intervals exclude training variance. Concept explanations, proficiency, causal learning effects, and GRU-state interpretation remain untested.

The additional tuning of the attention baselines (AKT and MOOCRadar simpleKT) selected settings on validation alone, without access to test data, and confirmed them over five folds and two seeds. The screen used only fold-0 validation and seed 42 on MOOCRadar, XES3G5M, and NIPS34. It varied width, block count, head count, dropout, learning rate, and feed-forward width one axis at a time and then in combinations fixed before execution. Among candidates within 0.0005 of the best validation AUC, the rule preferred fewer changed axes, then higher AUC, then registration order. The validation gains were +0.0092 for AKT with learning rate 0.0001 and +0.0096 for simpleKT with dropout 0.5 on MOOCRadar, and +0.0028 for AKT with two blocks on XES3G5M. AKT and the MOOCRadar simpleKT candidate were confirmed over five folds and seeds 42/3407. The validation-selected AKT reaches 0.860145 ± 0.000441 on MOOCRadar and 0.817766 ± 0.000556 on XES3G5M. Its gains over the published settings are +0.005477 [+0.004688, +0.006267] and +0.001206 [+0.000972, +0.001438], and the differences to EoKT remain +0.041342 [+0.038881, +0.043826] and +0.010984 [+0.010507, +0.011480]. The four-block NIPS34 setting was also confirmed in ten runs and appears in Table 13. The validation-selected simpleKT reaches 0.867082 ± 0.000915 on MOOCRadar; its gain over the published setting (0.8626 ± 0.0007) is +0.004470 [+0.003740, +0.005215] and the difference to EoKT is +0.034405 [+0.032567, +0.036301]. Against the +0.0096 validation gain of the screen, the five-fold mean gain of the confirmation stage is +0.0044 on validation and +0.0045 on test. On MOOCRadar the validation-selected simpleKT is above the validation-selected AKT. simpleKT on XES3G5M and NIPS34 has no confirmation stage, because the screen retained the published setting. NIPS34 AKT with the published setting reaches 0.800628 ± 0.000682; the EoKT difference is +0.006103 [+0.005281, +0.006953]. Here $\pm$ is sample SD across five selected folds, and brackets are 95% paired learner-bootstrap intervals. These are measured test differences, not transferred validation gains; only question-level validation AUC selected settings and seeds.

On MOOCRadar, rescoring EoKT with a truncated history examines the history-length asymmetry. We reuse the ten existing MOOCRadar checkpoints (five folds by seeds 42/3407) without retraining and rescore them, giving each scoring position only the responses that accumulate to 199 KC rows counting backwards from it (Table 11). Because the GRU is causal, this is the same as delaying the start of the window. The main rule, partial, counts the response that straddles the boundary; the sensitivity rule, whole, does not. The history-length distribution under partial matches the definition of the input audit referenced in Section 3 in all five of its statistics. Mean history length falls from 65.6 to 54.9 responses, and the start of the history moves at 24.1% of the scoring positions. AUC falls in all ten runs, but the mean paired difference is -0.00025, with an SD of 0.00008 across runs. These ten checkpoints are not the runs of Table 1, which were selected from candidates including seed 2026, so the absolute AUCs in Table 11 differ from Table 1 and only the within-run paired differences are interpreted ($\pm$ in Table 11 is the SD across runs). The per-run 95% learner-bootstrap intervals together span [-0.00050, +0.00002], and nine of the ten exclude zero. Against the MOOCRadar difference of +0.0056 in Table 2, the mean paired shrinkage stays below 0.0003, so the history-length asymmetry does not account for that difference. Rescoring the same checkpoints with only the 6 preceding responses, the effective history of LKT (Appendix G), gives 0.8851 ± 0.0021 (paired difference -0.0160); 4 responses give 0.8747 and 8 give 0.8894 (Table 11). We did not retrain with the truncated history, so the value under that condition is not measured. The XES3G5M checkpoints were not available here, so this control covers MOOCRadar only.

Table 11: MOOCRadar EoKT rescored with shorter histories, without retraining

History rule	AUC	Delta vs 199-response window
199-response window (rule of Table 1)	0.9011 ± 0.0004	—
199 KC rows, partial (main)	0.9008 ± 0.0004	-0.00025 ± 0.00008
199 KC rows, whole (sensitivity)	0.9008 ± 0.0004	-0.00025 ± 0.00008
6 preceding responses (LKT median; Appendix G)	0.8851 ± 0.0021	-0.0160 ± 0.0021
4 preceding responses (LKT p10)	0.8747 ± 0.0032	-0.0264 ± 0.0033
8 preceding responses (LKT p90)	0.8894 ± 0.0015	-0.0117 ± 0.0016

Four kinds of uncertainty appear in this paper. (1) Main tables give five-fold means with the sample SD across folds (ddof 1) of the validation-selected runs. (2) A paired difference between two conditions is computed on identical scoring keys; its 95% interval is the percentile interval of 10,000 matched bootstrap draws over original learners. One draw is shared across all conditions and folds, and the interval conditions on the selected runs and excludes training-seed variability. (3) Question-holdout results, including the strata of Experiment 4, are means and SDs over ten runs (five folds $\times$ two training seeds). (4) The difference between text embeddings and permuted text embeddings, and the within- and between-stratum differences of Experiment 4, carry the percentile interval of 10,000 paired learner-bootstrap draws shared across the ten runs of the compared conditions, conditional on the fixed holdout, permutation, and runs, and excluding training-seed and holdout-set variability. Folds share the test learners and overlap in training data, so they are not independent samples. The two-way bootstrap of Table 3 and Appendix C resamples learners and questions independently and so also covers question-level variability.

B Paired Differences and Question-side Controls

All comparisons are listed with their point estimates and intervals. The main text shows only the representative comparison and the KC additions. Here we list every paired difference computed for each dataset (Table 12 for MOOCRadar, Table 13 for XES3G5M and NIPS34). All rows compare self-run conditions and carry a 95% learner-bootstrap interval. The representative standard method is selected per fold on validation; IEKT was selected in every fold of both datasets. The three MOOCRadar contrasts of EERNN-A use the same 2,654 learners and 165,917 scoring positions. The EERNN-A differences reported in these tables average 10,000 bootstrap draws, each computing weighted AUC within each fold and then the five-fold mean difference. EoKT minus EERNN-A has observed difference +0.006590, bootstrap mean +0.006595, and 95% interval [+0.005647,

+0.007620]. The two XES3G5M contrasts of EERNN-A (the patience-5 runs of Appendix A) use the same 3,613 learners and 1,099,197 scoring positions. There, EoKT minus EERNN-A has observed difference +0.012174, bootstrap mean +0.012180, and 95% interval [+0.011678, +0.012700]. EERNN-A minus qDKT has observed difference −0.003701, bootstrap mean −0.003704, and 95% interval [−0.004323, −0.003084]. Intervals condition on selected runs and exclude training-seed variability. For NIPS34, matching was verified for 983 original learners and 271,833 scored responses after restoring original IDs. Because one draw applies to every condition and fold, the difference between two identical conditions is exactly zero in every replicate.

Table 12: Paired differences in pooled warm AUC on MOOCRadar

Dataset	Comparison	Difference	95% CI
MOOCRadar	EoKT − simpleKT (published setting)	+0.0389	[+0.0367, +0.0411]
MOOCRadar	EoKT − simpleKT (validation-selected)	+0.0344	[+0.0326, +0.0363]
MOOCRadar	simpleKT (validation-selected) − simpleKT (published setting)	+0.0045	[+0.0037, +0.0052]
MOOCRadar	EoKT − IEKT	+0.0056	[+0.0048, +0.0063]
MOOCRadar	EoKT − AKT (validation-selected)	+0.0413	[+0.0389, +0.0438]
MOOCRadar	EoKT − qDKT (ID only)	+0.0112	[+0.0100, +0.0126]
MOOCRadar	EoKT − EERNN-A	+0.0066	[+0.0056, +0.0076]
MOOCRadar	EERNN-A − I3	-0.0066	[-0.0077, -0.0056]
MOOCRadar	EERNN-A − qDKT (ID only)	+0.0046	[+0.0035, +0.0059]
MOOCRadar	EoKT − LKT	+0.1508	[+0.1371, +0.1650]
MOOCRadar	LKT − qDKT (ID only)	-0.1395	[-0.1537, -0.1258]
MOOCRadar	LKT − EERNN-A	-0.1442	[-0.1582, -0.1306]
MOOCRadar	EoKT − EoKT w/o IRB	+0.0067	[+0.0059, +0.0076]
MOOCRadar	EoKT + learned KC − EoKT	-0.0010	[-0.0014, -0.0006]
MOOCRadar	EoKT + pretrained KC − EoKT	0.0000	[-0.0003, +0.0002]
MOOCRadar	EoKT w/o IRB + learned KC − EoKT w/o IRB	+0.0001	[0.0000, +0.0002]
MOOCRadar	EoKT w/o IRB + pretrained KC − EoKT w/o IRB	-0.0001	[-0.0002, 0.0000]

Adding KC inputs to EoKT without the IRB moves pooled warm AUC by at most 0.0002 (Table 14). Table 14 lists the validation-selected AUC (five-fold mean ± sample SD across folds) of the three conditions moved here from Table 1. EoKT without the IRB keeps the projection and the GRU and drops only the block. The paired differences of these conditions (learned KC +0.0001 and −0.0001, pretrained KC −0.0001 and −0.0002) are in Tables 12 and 13.

The control suite fixes everything except what enters the question side. Every cell trains with the same GRU, the same fusion point, and the same optimization as EoKT, and changes only the question-side input. I3 sends a question-ID table learned from scratch through the same IRB, and I0 uses the same table without the IRB. MLP3 replaces the IRB with an MLP of the same parameter scale, and K0 uses KC vectors alone as the question representation. MLP3 nearly matches EoKT on the warm-dominated evaluation (Table 15). This supports only a small advantage for the tested IRB configuration; the equal-size MLP under artificial question holdout is compared in Table 19 of Appendix C. F3T trains a table jointly with KT from the same pretrained embedding that EoKT keeps fixed, and F3KCPreCat concatenates pretrained KC vectors before the IRB. F3perm keeps the set of embeddings and reassigns them to questions by a fixed permutation. In Table 15 under normal training, F3perm was run outside the control suite, so its AUC is the mean ± SD over ten runs (five folds × two seeds) and it has no paired bootstrap; Appendix C reports its paired interval under artificial question holdout. No paired difference against EoKT was computed for I0. EoKT uses the runs of Table 1. EoKT and the KC conditions are selected from candidates including seed 2026, the other controls from seeds 42/3407, with failed runs excluded. AKTemb replaces the question-difficulty table of AKT with a fixed text embedding and a linear layer. It is omitted from the table because its architecture differs. On MOOCRadar, neither seed 42 nor 3407 had a valid stored run for fold 4, leaving no candidate for validation-based selection. The archived records do not preserve a diagnostic cause. We do not impute the missing fold: the reported value averages the other four folds and is excluded from five-fold paired comparisons. That four-fold mean pooled warm AUC is

Table 13: Paired differences in pooled warm AUC on XES3G5M and NIPS34

Dataset	Comparison	Difference	95% CI
XES3G5M	EoKT − simpleKT	+0.0130	[+0.0125, +0.0136]
XES3G5M	EoKT − IEKT	+0.0046	[+0.0042, +0.0050]
XES3G5M	EoKT − AKT (validation-selected)	+0.0110	[+0.0105, +0.0115]
XES3G5M	EoKT − qDKT (ID only)	+0.0085	[+0.0080, +0.0090]
XES3G5M	EoKT − EERNN-A	+0.0122	[+0.0117, +0.0127]
XES3G5M	EERNN-A − qDKT (ID only)	-0.0037	[-0.0043, -0.0031]
XES3G5M	EoKT − LKT	+0.0390	[+0.0382, +0.0398]
XES3G5M	LKT − qDKT (ID only)	-0.0305	[-0.0312, -0.0298]
XES3G5M	LKT − EERNN-A	-0.0268	[-0.0277, -0.0260]
XES3G5M	EoKT − EoKT w/o IRB	+0.0148	[+0.0143, +0.0153]
XES3G5M	EoKT + learned KC − EoKT	-0.0037	[-0.0040, -0.0035]
XES3G5M	EoKT + pretrained KC − EoKT	-0.0001	[-0.0003, 0.0000]
XES3G5M	EoKT w/o IRB + learned KC − EoKT w/o IRB	-0.0001	[-0.0002, -0.0001]
XES3G5M	EoKT w/o IRB + pretrained KC − EoKT w/o IRB	-0.0002	[-0.0002, -0.0001]
NIPS34	EoKT − simpleKT	+0.0034	[+0.0027, +0.0042]
NIPS34	EoKT − AKT (validation-selected)	+0.0053	[+0.0045, +0.0061]
NIPS34	EoKT − qDKT (ID only)	+0.0081	[+0.0071, +0.0091]
NIPS34	EoKT − EoKT w/o IRB	+0.0038	[+0.0033, +0.0043]
NIPS34	EoKT + learned KC − EoKT	-0.0016	[-0.0021, -0.0012]
NIPS34	EoKT + pretrained KC − EoKT	+0.0002	[0.0000, +0.0004]
NIPS34	EoKT w/o IRB + learned KC − EoKT w/o IRB	+0.0001	[+0.0001, +0.0002]
NIPS34	EoKT w/o IRB + pretrained KC − EoKT w/o IRB	-0.0001	[-0.0002, 0.0000]

Table 14: Validation-selected pooled warm AUC of EoKT without the IRB and its KC additions

Condition	MOOCRadar	XES3G5M
EoKT w/o IRB	0.8947 ± 0.0002	0.8140 ± 0.0001
w/o IRB + learned KC	0.8948 ± 0.0001	0.8138 ± 0.0004
w/o IRB + pretrained KC	0.8946 ± 0.0002	0.8138 ± 0.0003

0.8495, below the 0.8546 of AKT with its published settings on the same four folds. AKTemb has no question-holdout runs, so it enters no cold comparison.

Table 15: Question-side controls on MOOCRadar (pooled warm AUC)

Question-side input (MOOCRadar)	AUC	Delta from EoKT [95% CI]
EoKT: fixed text embedding + IRB (F3)	0.9015 ± 0.0004	reference
Question-ID table + IRB (I3)	0.9015 ± 0.0004	0.0000 [-0.0005, +0.0005]
IRB replaced by an equal-size MLP (MLP3)	0.9010 ± 0.0002	-0.0005 [-0.0008, -0.0003]
KC tags only, no question text (K0)	0.8979 ± 0.0004	-0.0036 [-0.0048, -0.0026]
Fixed text embedding, no IRB (E0)	0.8947 ± 0.0002	-0.0067 [-0.0076, -0.0059]
Question-ID table, no IRB (I0)	0.9008 ± 0.0002	—
Permuted text embeddings + IRB (F3perm)	0.8993 ± 0.0006	—
Text table trained with KT (F3T)	0.9018 ± 0.0005	+0.0003 [-0.0001, +0.0007]
Pretrained KC before the IRB (F3KCPreCat)	0.9009 ± 0.0007	-0.0005 [-0.0008, -0.0002]

C UNSEEN-QUESTION RESULTS

The holdout is applied to the question vocabulary, and the test set is untouched. A tenth of the MOOCRadar question vocabulary is drawn at random, and responses to those questions are removed from training and validation. The test set is left as it is, so the withheld questions remain in scoring as questions the model never saw. The withheld set is identical in every run, and the sha256 of its question list is recorded. Each fold has 18,624 responses to withheld questions, of which 1,014 learners answer both correctly and incorrectly. The within-learner cold AUC is the mean of the AUCs of those 1,014 learners, and warm refers to responses to the questions that were not withheld. In Table 16, pooled all-response AUC mixes the warm and cold responses of these holdout-trained models, and pooled warm AUC uses only the non-withheld responses, so neither equals the all-response AUC after normal training. A withheld question is a zero row of the ID table. Without the IRB the inner product of the readout is zero, every cold prediction of a learner is the same value, and the within-learner cold AUC is exactly 0.5000. With the IRB the zero row maps to one constant vector and the cold prediction is a one-dimensional projection of the learner state. That projection cannot distinguish cold-question identities at a fixed learner state, although the state changes over time. Each cell has ten runs (five folds by two seeds); the five folds share one test set, so $\pm$ expresses the variation of the split and the seed. The learner's own running accuracy with no model gives a within-learner cold AUC of 0.4410, which is lower than configurations preserving the text-to-question mapping. Within-learner cold AUC also depends on the history state at each time and does not isolate question difficulty.

Table 16: MOOCRadar results of models trained with a tenth of the questions withheld. E0 = text embedding without the IRB, F3 = EoKT, F3KCPretrained = EoKT with pretrained KCs, F3perm = embeddings reassigned by a fixed permutation, I0/I3 = question-ID table without/with the IRB.

Cell	Pooled all-response AUC	Pooled cold AUC	Within-learner cold AUC	Pooled warm AUC
E0	0.8823 ± 0.0007	0.7975 ± 0.0049	0.5887 ± 0.0097	0.8932 ± 0.0006
F3 (EoKT)	0.8929 ± 0.0005	0.8381 ± 0.0037	0.6041 ± 0.0118	0.8997 ± 0.0005
EERNN-A	0.8859 ± 0.0011	0.8268 ± 0.0090	0.5950 ± 0.0257	0.8930 ± 0.0005
F3KCPretrained	0.8932 ± 0.0006	0.8361 ± 0.0041	0.6123 ± 0.0080	0.8998 ± 0.0004
F3perm	0.8869 ± 0.0013	0.8014 ± 0.0098	0.5193 ± 0.0123	0.8969 ± 0.0005
I0	0.8629 ± 0.0025	0.5000 ± 0.0000	0.5000 ± 0.0000	0.8995 ± 0.0004
I3	0.8949 ± 0.0003	0.8393 ± 0.0009	0.5047 ± 0.0067	0.9007 ± 0.0004
Running accuracy	0.7919	0.8088	0.4410	0.7897

The 95% interval [+0.0625, +0.1069] of the paired difference +0.0848 between text and permuted text embeddings is the percentile interval of 10,000 paired bootstrap draws over original learners. Each draw resamples with replacement the 2,214 learners who answered withheld questions, and the 1,014 of them with both outcomes enter the within-learner mean. A draw is shared across the ten runs of both conditions and yields the difference of the ten-run means. This learner bootstrap shares seed 20260911 with the intervals of Appendix B but draws from that smaller population, so it is a separate draw sequence. Per training seed, the difference is +0.0932 [+0.0708, +0.1159] for seed 42 and +0.0763 [+0.0539, +0.0985] for seed 3407. The paired difference in pooled cold AUC is +0.0367 [+0.0305, +0.0429]. The interval conditions on the fixed holdout, permutation, and trained runs, and it excludes training-seed and holdout-set variability. EERNN-A has no paired interval. The point estimates (0.6041, 0.5193, and +0.0848) coincide with the ten-run means of Table 16.

We evaluate public input replacements on two datasets, three holdouts, and both training seeds. Fixed 1024-dimensional BGE-M3 (Chen et al., 2024) and multilingual-e5-large (e5) (Wang et al., 2024) inputs are used in EoKT. The main 160-run comparison covers two datasets, normal/original-holdout training, four inputs including 3072-dimensional gemini-embedding-2 and I3, five folds, and two seeds. We trained 120 additional runs: two added holdouts $\times$ two datasets $\times$ BGE/e5/I3 $\times$ five folds $\times$ seeds 42/3407, auditing all 180 runs including 60 original-holdout runs. gemini-embedding-2 comparisons use the original holdout only. BGE/e5 follow prior investigations rather than independent selection. Three IRB loops, GRU, learning rate 0.0003, batch 64, 200 epochs, patience 10, and improvement 0.001 are fixed; there is no extra search, seed selection, or imputa-

tion. Time-gap input is retained; interaction with the time-gap ablation of Appendix A is untested. NIPS34 is excluded.

Cold responses under the artificial holdout are scored on withheld sets fixed before training. Cold sets were fixed before training with seeds 20260912 (original), 20260915, and 20260916: 10% each (251/2,512 MOOCRadar; 765/7,651 XES3G5M), removed from training and validation before windowing. Independent draws may overlap; IDs, SHA hashes, and overlaps are recorded in the preregistered manifest. Pairwise intersections as fractions of each holdout contain 5.6–11.6% of MOOCRadar questions and 9.3–10.7% of XES3G5M questions. The three holdouts share test learners and folds, so they are not independent replications. Test histories and prior encounters remain allowed, so this is not first-encounter-only evaluation. This models new questions before model retraining, including repeat practice. The shared zero vector prevents I3 from identifying which unseen question has recurred. All rows, including unscored rows, were aligned across the 180 runs by learner ID, position, question ID, label, mask, and history length; no runs or alignments were missing or inconsistent, and saved AUCs were reproduced. The original 160-run audit is retained.

Paired within-learner cold AUC intervals are reported by holdout and training seed. Table 17 gives sixteen original-holdout gemini-embedding-2/I3 comparisons and sixteen I3 comparisons across the two added holdouts.

Table 17: Paired differences in within-learner cold AUC by holdout. MR = MOOCRadar, XES = XES3G5M; h0912/0915/0916 = holdout seeds 20260912/20260915/20260916.

Comparison	Seed 42: Δ [95% interval]	Seed 3407: Δ [95% interval]
MR / h0912		
BGE-M3 − gemini-embedding-2	-0.013679 [-0.029789, +0.002742]	-0.006880 [-0.022590, +0.009209]
e5 − gemini-embedding-2	-0.008338 [-0.025160, +0.008986]	-0.004005 [-0.019195, +0.011642]
BGE-M3 − I3	+0.092293 [+0.073186, +0.111095]	+0.085943 [+0.065655, +0.106014]
e5 − I3	+0.097635 [+0.076740, +0.118433]	+0.088818 [+0.068743, +0.109055]
XES / h0912		
BGE-M3 − gemini-embedding-2	+0.002207 [-0.002313, +0.006742]	-0.001631 [-0.005718, +0.002539]
e5 − gemini-embedding-2	-0.000206 [-0.004357, +0.004112]	-0.005199 [-0.009412, -0.000851]
BGE-M3 − I3	+0.058269 [+0.052934, +0.063601]	+0.060990 [+0.055502, +0.066555]
e5 − I3	+0.055857 [+0.050252, +0.061586]	+0.057422 [+0.051917, +0.062882]
MR / h0915		
BGE-M3 − I3	+0.097239 [+0.078313, +0.116304]	+0.112622 [+0.093878, +0.131383]
e5 − I3	+0.056613 [+0.037496, +0.075414]	+0.064385 [+0.046298, +0.082433]
XES / h0915		
BGE-M3 − I3	+0.043979 [+0.038308, +0.049848]	+0.038200 [+0.032651, +0.043962]
e5 − I3	+0.064923 [+0.059441, +0.070377]	+0.058863 [+0.053252, +0.064507]
MR / h0916		
BGE-M3 − I3	+0.084212 [+0.064215, +0.103895]	+0.095762 [+0.076514, +0.115327]
e5 − I3	+0.052065 [+0.030131, +0.073782]	+0.060725 [+0.038752, +0.082943]
XES / h0916		
BGE-M3 − I3	+0.039086 [+0.033340, +0.044702]	+0.046264 [+0.040378, +0.052024]
e5 − I3	+0.032223 [+0.026422, +0.037840]	+0.048516 [+0.042766, +0.054021]

The control with permuted text embeddings was also run on the XES3G5M holdout, with the same result as on MOOCRadar. Ten runs (five folds $\times$ training seeds 42 and 3407) train EoKT with the gemini-embedding-2 embeddings reassigned to other questions by a fixed permutation (seed 20260912). The holdout is holdout 1 of XES3G5M (765 of 7,651 questions, seed 20260912), and the code, settings, and holdout are those of the gemini-embedding-2 reference runs in Table 4. Within-learner cold AUC is 0.6489 $\pm$ 0.0073 for text and 0.5656 $\pm$ 0.0099 for permuted text embeddings (ten-run mean $\pm$ SD). The paired difference is +0.0833 [+0.0772, +0.0895]; the interval comes from 10,000 learner-bootstrap draws shared across the ten runs of both conditions. Each draw resamples the 3,613 learners with cold responses, and the 3,388 with both outcomes enter the within-learner mean (103,134 cold responses). Per training seed the difference is +0.0785 [+0.0721, +0.0850] and +0.0881 [+0.0820, +0.0944], and the paired difference in pooled cold AUC is +0.0552 [+0.0518, +0.0587]. As above, the interval conditions on the fixed holdout, permutation, and runs. Question-ID inputs reach 0.5896 on the same holdout (Table 4), so on XES3G5M the permuted text embeddings stay below text and below the ID table.

Pooled cold AUC differences are interpreted separately from within-learner cold AUC. Table 18 gives pooled cold AUC differences by holdout/training seed. Public-input-minus-I3 intervals for all three holdouts use the cold-set learner bootstrap described below; gemini-embedding-2-related intervals use draws shared between warm and cold responses.

Table 18: Paired differences in pooled cold AUC by holdout. MR = MOOCRadar, XES = XES3G5M; h0912/0915/0916 = holdout seeds 20260912/20260915/20260916.

Comparison	Seed 42: Δ [95% interval]	Seed 3407: Δ [95% interval]
MR / h0912		
BGE-M3 − gemini-embedding-2	-0.004091 [-0.008056, -0.000174]	-0.001535 [-0.005414, +0.002256]
e5 − gemini-embedding-2	-0.013627 [-0.017923, -0.009291]	-0.007701 [-0.011546, -0.003891]
BGE-M3 − I3	-0.004004 [-0.007820, -0.000081]	-0.004040 [-0.007883, -0.000006]
e5 − I3	-0.013540 [-0.018186, -0.008666]	-0.010206 [-0.014651, -0.005538]
XES / h0912		
BGE-M3 − gemini-embedding-2	+0.003330 [+0.000980, +0.005706]	+0.000944 [-0.001229, +0.003154]
e5 − gemini-embedding-2	-0.000288 [-0.002494, +0.001980]	-0.001793 [-0.004038, +0.000445]
BGE-M3 − I3	+0.017439 [+0.014809, +0.020143]	+0.018627 [+0.015934, +0.021417]
e5 − I3	+0.013821 [+0.011038, +0.016628]	+0.015891 [+0.013132, +0.018660]
MR / h0915		
BGE-M3 − I3	+0.015370 [+0.009933, +0.020906]	+0.019094 [+0.013334, +0.025016]
e5 − I3	+0.004821 [-0.000927, +0.010814]	+0.009484 [+0.003956, +0.015399]
XES / h0915		
BGE-M3 − I3	+0.004356 [+0.001357, +0.007364]	+0.001080 [-0.001866, +0.004071]
e5 − I3	+0.021008 [+0.017906, +0.024146]	+0.015607 [+0.012520, +0.018785]
MR / h0916		
BGE-M3 − I3	+0.014810 [+0.008828, +0.021122]	+0.016301 [+0.010469, +0.022440]
e5 − I3	-0.003098 [-0.009081, +0.003043]	-0.007326 [-0.013559, -0.001006]
XES / h0916		
BGE-M3 − I3	+0.001125 [-0.001757, +0.003932]	+0.004369 [+0.001461, +0.007241]
e5 − I3	-0.003235 [-0.006037, -0.000442]	+0.006393 [+0.003613, +0.009188]

Paired learner bootstrap intervals condition on the trained runs. MOOCRadar h20260912: 18624 cold responses (15268 correct/3356 incorrect), macro 1014 learners/11116 responses; MOOCRadar h20260915: 17273 cold responses (13758 correct/3515 incorrect), macro 1018 learners/11286 responses; MOOCRadar h20260916: 14988 cold responses (12488 correct/2500 incorrect), macro 981 learners/8568 responses; XES3G5M h20260912: 103134 cold responses (83068 correct/20066 incorrect), macro 3388 learners/97453 responses; XES3G5M h20260915: 112431 cold responses (90609 correct/21822 incorrect), macro 3432 learners/107336 responses; XES3G5M h20260916:

105540 cold responses (85303 correct/20237 incorrect), macro 3472 learners/101902 responses. Learners without both labels are excluded from macro. The public-input bootstrap resamples original scored learners before cold masking, sharing 10,000 valid draws per dataset (RNG seed 20260911; rejected: MOOCRadar 0, XES3G5M 0) across all three holdouts, models, folds, and both seeds. Matched differences are averaged over five folds per seed with linear-percentile individual 95% intervals. The original-holdout gemini-embedding-2-related rows in Tables 17 and 18 use intervals from draws shared between warm and cold responses; public-input-minus-I3 intervals for all three holdouts use this bootstrap. Points are observed differences, not bootstrap means. Intervals condition on fixed runs/holdouts and exclude training-seed/holdout variation. The 24 public-input-minus-I3 primary and 24 supplementary comparisons are not multiplicity-adjusted and do not provide simultaneous 95% coverage. Intervals including zero do not establish equivalence.

Public-input gains depend on the measure and comparator. Within-learner cold AUC: The 24 observed differences range from +0.032223 to +0.112622 (24 positive/0 negative/0 zero). Individual 95% CIs: 24 positive/0 negative/0 including zero. Sign reversals across holdouts within matched dataset/input/training-seed conditions were not observed. Pooled cold AUC: The 24 observed differences range from -0.013540 to +0.021008 (17 positive/7 negative/0 zero). Individual 95% CIs: 14 positive/6 negative/4 including zero. Sign reversals across holdouts within matched dataset/input/training-seed conditions occurred. These counts are not success criteria for replication. gemini-embedding-2-versus-public comparisons apply only to the original holdout. EoKT with gemini-embedding-2 was not trained on the added holdouts; its relative performance is not extrapolated. Between-learner and within-learner ranking differ, and macro remains sensitive to history states. Within-learner cold AUC equally weights learners with both correct and incorrect cold responses. Individual 95% intervals condition on fixed runs and holdouts, exclude training-seed and holdout-set variability, and are not adjusted for multiple comparisons. Table 16 gives EERNN-A's matched ten-run results without paired holdout intervals. The original MOOCRadar KC increase in within-learner cold AUC from 0.6041 to 0.6123 (Table 16) is a separate comparison; KC benefits on cold questions remain possible.

This practical replacement comparison retains provenance and capacity limitations. Public inputs have 1024 dimensions versus gemini-embedding-2's 3072, changing the projection to 512 dimensions. Trainable parameter counts are 3,333,026 for public-input F3, 4,383,650 for gemini-embedding-2 F3, and 12,100,514/27,887,522 for MOOCRadar/XES3G5M I3; this is not an equal-capacity causal comparison. No public-input shuffle control is added, so semantic and question-identification contributions are not separated. Existing public NPZ assets have approximately unit L2 norms and fixed SHA hashes, but lack complete generation-revision and full-input-text hash metadata. Five normally trained MOOCRadar gemini-embedding-2 seed-42 runs lack recorded semantic SHA; they are included based on existing configuration/result fingerprints, manifest/source compatibility audits, and full-response alignment, without imputing the missing hashes. Configurations, per-run metrics, aggregates, paired intervals, and audits are included in the open-embedding comparison record of the supplement. This is not an ESKT experiment; predictive advantages over EKT/ESKT remain untested. The additional-holdout settings, the per-run metrics of all 180 runs, the 48 individual comparisons, the 180-run aggregates, and the missing-run audit are included in the same record; the bootstrap draws are retained in the experiment records and are not distributed.

We extend the cold-question comparison on the same artificial holdout to methods that read KC annotations. The holdout is the MOOCRadar holdout of seed 20260912 used above (251 of 2,512 questions), and each cell is ten runs (five folds by training seeds 42/3407; Table 19). On the KC-expanded path (AKT and simpleKT) the KC rows of the withheld questions are removed from the training and validation sequences before the sequences are rebuilt, and the test set is scored with the same scoring key as every other condition. On MOOCRadar, runs on that path carry 17 cold responses per run that the reference condition lacks, so we report no pooled cold AUC and compare only the within-learner cold AUC after aligning every condition to a shared set of learner-position pairs. Paired differences to EoKT match fold and training seed and are bracketed by 10,000 paired learner-bootstrap draws (seed 20260911). EERNN-A on MOOCRadar and the question-ID table have no paired interval. These intervals are conditional on the fixed runs and do not include fold or training-seed variation. The preregistration limited this bootstrap to checking the permutation control; we applied the same draws uniformly to every condition without selecting cells. Replacing the IRB with an equal-size MLP raises within-learner cold AUC by a mean of 0.0050; the learner-only conditional interval [+0.0012, +0.0089] excludes zero, but resampling questions as well gives [−0.0058, +0.0153], which includes it. The last column of Table 19 resamples learners and questions independently (two-way bootstrap, 10,000 draws, seed 20260925). The widening comes almost entirely from question-side variation; 220 and 693 questions have cold responses on MOOCRadar and XES3G5M. Under this bootstrap the difference to the ID table is [+0.0257, +0.1483] and [+0.0346, +0.0871]. The SD of paired differences across runs is 0.0135; it is not an equivalence test. This result provides no support for an IRB-specific advantage. The interaction between the time-gap input and the cold condition, and a control that matches the encoders, are not carried out in this paper. For EERNN-A on XES3G5M, ten runs on the same holdout (seed 20260912) give a within-learner cold AUC of 0.6132 ± 0.0100 and a paired difference to EoKT of -0.0357 [-0.0404, -0.0311], a smaller gap to EoKT than the -0.0833 of permuted text embeddings. Each run was trained under patience 5, an improvement threshold of 0.001, and at most 60 epochs, and all ten stopped by the rule (patience 5) after 22 to 40 epochs, with selected checkpoints at epochs 17 to 35. This rule stops earlier than that of EoKT (patience 10, at most 200 epochs); Appendix G gives the stopping points and a comparison with runs capped at 33 epochs. The three methods reading KC annotations on XES3G5M are also ten runs on the same holdout (765 of 7,651 questions); AKT and simpleKT use the same published settings as on MOOCRadar, and the simpleKT runs are re-scored from runs trained for another comparison. On XES3G5M the KC-expanded runs carry no cold responses that the reference condition lacks. The differences to EoKT are -0.0259 [-0.0316, -0.0204] for the KC-only input, -0.0404 [-0.0457, -0.0351] for simpleKT, and -0.0423 [-0.0474, -0.0371] for AKT.

Table 19: Within-learner cold AUC on the same artificial holdout. Differences are the cell minus EoKT, the opposite sign of the main text.

Condition	Within-learner cold AUC	Difference vs EoKT	95% CI	Two-way 95% CI
MOOCRadar (ten runs)				
IRB replaced by an MLP	0.6091 ± 0.0071	+0.0050	[+0.0012, +0.0089]	[-0.0058, +0.0153]
EoKT	0.6041 ± 0.0118	—	—	—
EERNN-A	0.5950 ± 0.0257	—	—	—
simpleKT (reads KCs)	0.5535 ± 0.0096	-0.0505	[-0.0719, -0.0290]	[-0.1253, +0.0198]
KC tags only (reads KCs)	0.5415 ± 0.0110	-0.0626	[-0.0854, -0.0393]	[-0.1533, -0.00001]
AKT (reads KCs)	0.5286 ± 0.0109	-0.0754	[-0.0951, -0.0557]	[-0.1427, -0.0107]
Permuted text embeddings	0.5193 ± 0.0123	-0.0848	[-0.1069, -0.0625]	[-0.1696, -0.0050]
Question-ID table (I3)	0.5047 ± 0.0067	—	—	—
XES3G5M (ten runs)				
EoKT	0.6489 ± 0.0073	—	—	—
KC tags only (reads KCs)	0.6230 ± 0.0032	-0.0259	[-0.0316, -0.0204]	[-0.0513, -0.0041]
EERNN-A	0.6132 ± 0.0100	-0.0357	[-0.0404, -0.0311]	—
simpleKT (reads KCs)	0.6085 ± 0.0022	-0.0404	[-0.0457, -0.0351]	[-0.0653, -0.0184]
AKT (reads KCs)	0.6067 ± 0.0053	-0.0423	[-0.0474, -0.0371]	[-0.0669, -0.0206]
Question-ID table (I3)	0.5896 ± 0.0020	—	—	—
Permuted text embeddings	0.5656 ± 0.0099	-0.0833	[-0.0895, -0.0772]	[-0.1169, -0.0520]

With the metric held fixed for the same holdout-trained models, the EoKT–ID difference varies between warm and cold responses. Table 20 recomputes saved predictions from models trained with artificial-holdout seed 20260912; no normally trained runs enter this comparison. The pooled EoKT-minus-I3 differences are -0.0010 warm/-0.0012 cold on MOOCRadar and -0.0020 warm/$+0.0159$ cold on XES3G5M, so the cold ordering depends on the metric. The scored populations are MOOCRadar warm: 147,293 responses (119,788 correct/27,505 incorrect), 2,654 learners, macro over 2,496 learners/142,821 responses; cold: 18,624 responses (15,268 correct/3,356 incorrect), 2,214 learners, macro over 1,014 learners/11,116 responses; XES3G5M warm: 996,063 responses (791,422 correct/204,641 incorrect), 3,613 learners, macro over 3,613 learners/996,063 responses; cold: 103,134 responses (83,068 correct/20,066 incorrect), 3,613 learners, macro over 3,388 learners/97,453 responses. Counts and scoring keys match across all models, folds, and seeds within each dataset. Warm means non-held-out questions, not guaranteed train-observed questions in every fold. Within-learner AUC equally weights learners with both outcomes in each subset. Because eligible populations differ, the contrast is descriptive rather than a causal effect of making questions cold. Test histories are unchanged, so evaluation is not restricted to first encounters.

Table 20: Warm/cold $\times$ pooled/within-learner AUC of the same holdout-trained models

Dataset / Model	Pooled warm AUC	Within-learner warm AUC	Pooled cold AUC	Within-learner cold AUC
MOOCRadar / EoKT	0.8997 ± 0.0005	0.7661 ± 0.0049	0.8381 ± 0.0037	0.6041 ± 0.0118
MOOCRadar / I3	0.9007 ± 0.0004	0.7669 ± 0.0029	0.8393 ± 0.0009	0.5047 ± 0.0067
XES3G5M / EoKT	0.8271 ± 0.0006	0.7784 ± 0.0010	0.7434 ± 0.0052	0.6489 ± 0.0073
XES3G5M / I3	0.8291 ± 0.0004	0.7804 ± 0.0008	0.7275 ± 0.0009	0.5896 ± 0.0020

D Numeric Variants: Definitions and Aggregation

Numeric variants are defined on the exact question text that is embedded. The text excludes image tokens and includes the options of multiple-choice questions. After removing whitespace and replacing integers and decimals with typed placeholders, two questions whose texts then match but whose number sequences differ are numeric variants. Two questions that also share the number sequence form a separate same-text stratum. Differences in Chinese numerals are not captured. Of the 7,651 XES3G5M questions, 656 fall into 249 templates. Among the 765 held-out questions, 57–59 have a numeric variant in the fold's training sequences and 100–102 have a same-text question there. The primary analysis uses the 49–51 questions of the former group for which neither the question nor its variants carries an image, because figure differences do not appear in the text. Counting only questions with scored responses, the fold means are 59 (51 without images), 91, and 543 for questions with neither. The extraction rule and strata were fixed in the research record after inspecting only counts, before the three models' predictions were aggregated.

An LLM check of 30 random pairs found no non-variants, but about half could change the solution steps. We sampled 30 pairs at random, and an LLM agent that did not see any predictions checked them against the worked solutions; one cold question can appear in several pairs. Fourteen used the same method with only the numbers changed, 16 could change the required steps or difficulty, and none was a non-variant. The latter include counting problems in which a zero digit adds a case for the leading position and arithmetic problems in which the numbers decide whether a factoring shortcut applies. Including questions with images, EoKT − I3 is +0.0639 [+0.0429, +0.0844]. Removing one template at a time keeps the difference on the variant stratum between +0.041 and +0.084. On this stratum the history-free TEXTONLY reaches 0.6284, and EoKT − TEXTONLY is +0.0327 [+0.0099, +0.0551] ([−0.0282, +0.0836], undetermined, with questions also resampled). Split by the same learner's earlier test responses, first encounters with a template total 6,445 responses from 769 eligible learners, where EoKT reaches 0.6433, I3 0.6072, and TEXTONLY 0.6066. Responses after an earlier question of the same template (58 eligible learners) and repeats of the same question (5) fall below the pre-specified 100 learners and are not compared.

With a broader near-duplicate definition, the advantage of text tends to be larger for textually closer questions, but with questions also resampled neither the near–far gap nor the distant-third margin is determined. For the stratum with neither a numeric variant nor an identical text, we re-match against training questions after NFKC, unified punctuation and LaTeX symbols, removal of leading instruction phrases, and conversion of Chinese numerals. Questions that then match (29 on average per fold, 217 eligible learners) give EoKT − I3 of +0.1016 [+0.0494, +0.1542]. We split the rest into thirds by the character-trigram Jaccard similarity of the number-replaced text to the nearest training question, with cut points 0.342 and 0.697 fixed as tertiles of this value before reading predictions. The difference is +0.0331 [+0.0218, +0.0447] in the distant third (177 questions, 2,519 eligible learners), +0.0473 [+0.0354, +0.0590] in the middle third, and +0.0969 [+0.0857, +0.1081] in the closest third; the closest minus the distant third is +0.0638 [+0.0480, +0.0798]. A two-way bootstrap that also resamples questions within each stratum (10,000 draws, seed 20260925) gives [+0.0011, +0.1458] for EoKT − I3 on image-free variants and [+0.0261, +0.0903] on the stratum with neither a variant nor an identical text, both excluding zero. It gives [−0.0107, +0.0859] for the distant third, [−0.0172, +0.1355] for the closest minus distant third, [−0.0235, +0.0848] for identical text minus no variant, and [−0.0642, +0.0935] for variant minus no variant, all including zero.

Rewording withheld questions changes AUC only within an interval that includes zero, changing their notation lowers AUC, and both rewrites stay above I3. We rewrote only the held-out question texts, keeping numbers, answers, and options, re-embedded them with gemini-embedding-2, and re-scored the ten trained F3 models; the original input was re-scored on the same device for the comparison. For 697 questions reworded by Gemini in phrasing and word order, the median cosine to the original embedding is 0.956, the median absolute change in prediction is 0.033, the within-learner rank correlation is 0.81, and within-learner cold AUC on the original labels changes by -0.0027 (two-way interval [-0.0185, +0.0108]). A rule-based notation change that drops math delimiters and uses half-width punctuation (765 questions) gives a cosine of 0.937, similar prediction changes and rank correlation, and an AUC change of -0.0157 ([-0.0286, -0.0035]). All training-side texts use the original notation, so we interpret the notation change as moving inputs away from the training distribution. In a supplementary comparison added after seeing the results, EoKT $-$ I3 on the same questions is +0.0548 [+0.0299, +0.0803] after rewording and +0.0436 [+0.0205, +0.0689] after the notation change (two-way intervals).

E Cross-dataset Pooled EoKT

One embedding model and one architecture serve three datasets that differ in language, school stage, subject, answer format, and source of question text. Table 21 lists them; the learners run from third grade (XES3G5M) to university (MOOCRadar, 24 courses) and only which input fields exist differs.

Table 21: The three datasets (format shares weighted by responses; text length in median characters per question)

	MOOCRadar	XES3G5M	NIPS34
Language	Chinese	Chinese	English
Questions / KCs	2,512 / 2,795	7,651 / 865	948 / 57
Answer formats	single 60.8%, multiple 22.5%, judge 14.1%, fill 2.6%	fill 69.6%, single 30.4%	single (4 options) 100%
Question text	native; 77 chars	native; 66 chars	transcribed; 281 chars

The pooled model is built by concatenating the datasets with dataset-prefixed question vocabularies. We concatenate the fold-0 training sets of MOOCRadar, XES3G5M, and NIPS34 and prefix question IDs with the dataset name so that the vocabularies do not collide. The embedding model is shared across the three, and one set of question-side projection, IRB, and history-side GRU is trained. Validation pools the three datasets and is the only signal for choosing a checkpoint. The test is separated by dataset and scored with the sliding window used in the main text. Two seeds (42 and 3407) are run, and Table 7 reports their mean and half-range. Across two languages and two school stages, the single pooled model's two-seed mean stays within 0.0012 of the per-dataset model on all three datasets. Seed 42, the seed that validation selects, stays within 0.0006. The claim here is compatibility, not transfer.

F Inference Latency: Measurement Conditions

Inference latency is the time to return the correctness probability of one learner's next question, measured on a CPU close to production serving. The machine is an AWS EC2 m6i.large (Intel Xeon Platinum 8375C, 2 vCPUs, PyTorch 2.14.0 CPU build); torch is pinned to one thread and batch size 1, one measurement is the median of 30 timed calls after 5 warm-ups, and we report the median of three such measurements. Table 22 reports three independent measurement sets, A to C, taken with this procedure on the same instance type. Weights are randomly initialised (the arithmetic is identical to trained weights). We benchmark the reference implementations of simpleKT, AKT, and IEKT by recomputing a window of 200 interactions for each prediction (window-200 recomputation). This measures the chosen evaluation path rather than an inherent requirement of every model. IEKT also has a recurrent state (Table 24), so sets B and C also benchmark the incremental implementation described below. A key-value cache for causal attention is possible in principle, but the pyKT

reference implementation has none, and this section measures that implementation as it is. EoKT is measured in the same window-200 configuration and in an incremental configuration that keeps only the GRU state (one 512-dimensional vector), consumes one new interaction, and returns the next question. The incremental path is composed from the model's own submodules, and every run checks that stepping through 200 interactions one at a time reproduces the window-200 logit at the last position within 1e-4 (the observed maximum difference is 3e-7). The question-side transform (projection and IRB) is a pure function of the question, so it is precomputed once per question, at 3.2 ms per question. EoKT w/o IRB recomputes the 200-interaction window in 37.1 ms against 65.1 ms for EoKT; the difference is the three IRB loops. LKT is one forward pass over up to 512 tokens; besides the randomly initialised RoBERTa-base configuration, the table lists, measured with the same procedure on the same instance type, the pretrained backbone of the MOOCRadar accuracy runs in Table 1 (hfl/chinese-roberta-wwm-ext, vocabulary 21,128) at their mean input length of 461 tokens; the smaller vocabulary makes the masked language model (MLM) head projection lighter, giving 769 ms. The AKT, simpleKT, and LKT values also agree in order and magnitude with a separate measurement on the same instance type (32.4, 7.5, and 1,069 ms).

Measurement set B, on the same AWS m6i.large instance type, gives 1.159 ms for incremental IEKT and 0.483 ms for incremental EoKT. Set B uses the same Intel Xeon Platinum 8375C, Python 3.11, PyTorch 2.14.0 CPU build, NumPy 2.4.6, and pyKT 0.0.38 as set A. With one CPU thread and batch size 1, we take the median of three independent-process medians, each from 30 calls after five warm-ups. The Linux kernel is 6.18.44 in set A and 6.1.186 in set B. Weights and the 200-response input are random. In sets A and B, IEKT is timed in its default width-64 configuration, which at inference is the MOOCRadar setting of Table 1 (dropout is inactive at inference, and gamma and the policy-loss weight enter only the training loss); set C times the XES3G5M setting of Table 1 (width 128). IEKT retains official Categorical sampling for cognition and sensitivity, caching 64 hidden-state values, 128 previous-question representation values, one predicted probability, and the RNG state. Each timed call includes sensitivity sampling and the state update for the previous response, followed by cognition sampling and the next-question prediction, without recomputing history. RNG state save/restore is excluded. Incremental run medians range from 1.148 to 1.160 ms for IEKT and 0.476 to 0.494 ms for EoKT. IEKT probabilities at all 200 positions differ from official window inference by at most 5.96e-8; the final RNG state and synthetic-label AUC agree. This checks path equivalence, not real-data AUC for a trained IEKT. EoKT uses the incremental path of set A on the same instance type, with a maximum all-position logit difference of 1.49e-7.

Measurement set C times IEKT at the XES3G5M setting of Table 1 (width 128): 312.96 ms for window-200 recomputation and 1.482 ms incrementally. Set C follows the procedure, dependency versions, and Linux kernel 6.1.186 of set B on a separate m6i.large instance. IEKT is built with the XES3G5M setting of Table 1 (width 128, dropout 0.3, gamma 0.8) and the XES3G5M vocabulary (7,651 questions and 865 KCs), giving the 5.40M parameters of the Table 1 runs. EoKT, timed on the same instance, takes 58.69 ms for window-200 recomputation and 0.488 ms incrementally, close to the 62.63 and 0.483 ms of set B. Incremental IEKT run medians range from 1.463 to 1.488 ms, and the cached state grows to 128 hidden-state values and 256 previous-question representation values. IEKT probabilities at all 200 positions differ from official window inference by at most 5.96e-8, and the final RNG state and synthetic-label AUC agree. A reference measurement that keeps the MOOCRadar vocabulary and changes only the width to 128 gives 1.608 ms for incremental IEKT. Table 6 in Section 4.5 takes EoKT, simpleKT, and AKT from set A and IEKT (width 128) from set C; in set C, EoKT takes 58.69 ms for window-200 recomputation and 0.488 ms incrementally.

Table 22: Time to return one learner's next-question prediction (Sets A to C: three independent measurement sets on the same instance type)

Measurement	Model	Inference path	Latency (ms)	Parameters
Set A	LKT	RoBERTa-base, up to 512 tokens	1042.54	124.70M
Set A	AKT	Window 200	24.12	4.25M
Set A	simpleKT	Window 200	5.83	2.27M
Set A	IEKT (width 64)	Window 200	254.88	1.49M
Set A	EoKT	Window 200	65.13	4.38M
Set A	EoKT	Incremental	0.50	4.38M
Set B	LKT	Accuracy backbone (pretrained), 461 tokens	769.45	102.29M
Set B	LKT	RoBERTa-base, up to 512 tokens	1025.54	124.70M
Set B	IEKT (width 64)	Window 200	247.05	1.49M
Set B	IEKT (width 64)	Incremental	1.159	1.49M
Set B	EoKT	Window 200	62.63	4.38M
Set B	EoKT	Incremental	0.483	4.38M
Set C	IEKT (width 128)	Window 200	312.96	5.40M
Set C	IEKT (width 128)	Incremental	1.482	5.40M
Set C	EoKT	Window 200	58.69	4.38M
Set C	EoKT	Incremental	0.488	4.38M

Table 23: Sliding-window versus retained-state re-scoring

Dataset	Window AUC	Incremental AUC	Delta (incr - window)	Share position>=200	Window AUC pos>=200	Incremental AUC pos>=200
MOOCRadar	0.9013 ± 0.0004	0.9013 ± 0.0004	0.0000	2.4%	0.8417 ± 0.0037	0.8416 ± 0.0042
XES3G5M	0.8288 ± 0.0003	0.8288 ± 0.0003	+0.0001	34.6%	0.8417 ± 0.0003	0.8419 ± 0.0003
NIPS34	0.8067 ± 0.0003	0.8067 ± 0.0003	0.0000	41.4%	0.8028 ± 0.0004	0.8028 ± 0.0003

Re-scoring every test response through EoKT's retained-state path reproduces its sliding-window AUC. For each test learner the GRU state is initialised once, responses are fed in order, and every response from the second onward is scored with the retained state and no window truncation. The scoring keys are identical to the sliding-window evaluation, and the predictions agree to within 1e-4 for positions up to 200. Pooled warm AUC agrees to four decimals on all three datasets (differences of at most 0.0001), and on the responses beyond original position 200 (2.4% of MOOCRadar, 34.6% of XES3G5M, and 41.4% of NIPS34) the difference stays within 0.0002. The GRU is a causal recurrence, so the window enters only through the pyKT scoring convention, not through the prediction. Table 23 gives five-fold means $\pm$ sample SDs of runs selected from seeds 42/3407. Table 1's MOOCRadar candidates also include seed 2026, so the MOOCRadar runs differ from those of Table 1, whereas XES3G5M and NIPS34 use the same selected seeds as Table 1.

G Input Representations and Operational Requirements

Since DKT, standard KT methods have combined question and KC representations with a model of the response history (Piech et al., 2015). The comparators in Section 4.1 are AKT, with Rasch-inspired question representations and monotonic attention (Ghosh et al., 2020); simpleKT, which combines question-specific variations with ordinary attention (Liu et al., 2023b); IEKT, which estimates each learner's cognition and acquisition sensitivity (Long et al., 2021); and qDKT, which predicts at the question level from a question-ID table alone (Sonkar et al., 2020). All but qDKT take KC annotations as input, and pyKT restored their comparability by fixing the split, the scoring, and the KC fusion rules (Liu et al., 2022).

Text-based methods differ in whether they require KCs or relational structure. KCQRL trains question representations contrastively using LLM-generated solutions and KCs (Ozyurt et al., 2024). PICKT links question and concept IDs and text through a knowledge map (Lee et al., 2025), and SINKT uses question and concept text with a relation graph for unseen questions (Fu et al., 2024). RKT derives exercise relations from text and responses (Pandey and Srivastava, 2020); KCluster derives KCs from LLM similarity judgments (Wei et al., 2025). KCGen-KT uses LLMs for KC generation and prediction (Duan et al., 2026). DKVMN is not text-based: it uses a static key memory and a value memory updated per learner (Zhang et al., 2017).

Language-model predictors differ in their inputs and evaluation settings from this paper's question holdout. NTKT serializes the history and question text, including option texts, for a fine-tuned LLM (Norris et al., 2025). LLM-KT injects question- and concept-specific tokens and a sequence-model state into the LLM (Wang et al., 2026). LKT also uses a language-model predictor and takes KC text in its reported formulation (Lee et al., 2024). CLST fine-tunes a generative language model on natural-language histories and evaluates low-data settings including fewer than 100 students or short histories (Jung et al., 2025), a different cold-start setting from our question holdout. These configurations execute a language model at prediction time. We make no numerical comparison with NTKT, LLM-KT, or CLST because their evaluation settings differ; our LKT implementation and measured results appear below.

Table 24 summarizes the inputs each method requires for a new question or dataset. ESKT tunes Longformer on RACE before freezing representations for Knowledge Tracing with an Answer Encoder and Multiple Questions Attention Mechanism (KTAM) (Cheng and Li, 2025). It requires learners' option responses and answer keys, so we do not evaluate it on our correctness-only logs. ESKT-AM also needs a fixed option set and answer key. MOOCRadar releases answer keys for every question and option sets for its choice formats, but we did not run ESKT-AM on it; that direct comparison is outside our scope. A method taking KC annotations cannot form an input for a new question until it is tagged. A question-ID table has no trained representation for a question absent from training. Text-based methods form representations from question text. EoKT requires neither answer keys nor structured option sets; available options enter as text.

EERNN-A is an independent character-level reimplementation of the attention variant of EERNN (Su et al., 2018). The attention variant combines a bidirectional-LSTM question encoder with attention that weights past states by question similarity; EKT (Liu et al., 2021) later extended EERNN with KC inputs. It uses random character initialization, no Word2vec, and batch size changed from 32 to 64. It uses 50-dimensional character embeddings, a BiLSTM with 100 dimensions per direction and max pooling, a 100-dimensional history LSTM, signed cosine attention over strict past states, and a 50-dimensional ReLU prediction layer; dropout is 0.1 and Adam uses learning rate 0.001. Attention retains past states and question vectors within the evaluation window of 200. One configuration uses seeds 42/3407; validation selected 3407 in all five MOOCRadar folds and 3407/42/3407/42/42 on XES3G5M (the patience-5 runs of Appendix A). KCs, time gaps, and the text-availability flag are excluded.

For EERNN-A on XES3G5M, training under a patience-5 rule with at most 60 epochs gives aggregate values no higher than runs capped at 30 to 33 epochs, so a training-time cap does not explain the gap to EoKT. The 20 XES3G5M runs of EERNN-A (ten normal and ten holdout runs, with the same folds, seeds, and holdout as the other conditions) are trained under patience 5, an improvement threshold of 0.001, and at most 60 epochs. All 20 stopped by the rule (patience 5), after 22 to 35 epochs for normal training and 22 to 40 for the holdout; none stopped at 60 epochs or at a training-time budget. To assess the stopping rule, we compare them with runs of the same 20 cells capped by training time at 30 epochs (normal) and 33 epochs (holdout). None of the capped runs had reached the patience-10 stopping rule, and over their last ten epochs the best validation AUC still rose by 0.0014 to 0.0036, about 0.0003 per epoch. The per-epoch validation AUC of the two sets matches wherever they overlap. Of the 12 runs that trained beyond the cap, 11 gained no more than 0.001 in validation AUC over the added epochs and selected the same checkpoint as the capped run; the remaining holdout run selected a later checkpoint, with validation AUC +0.0011 and within-learner cold AUC +0.0001. Of the eight runs that stopped before the cap, seven selected an earlier checkpoint, with test values 0.0019 to 0.0093 lower. The capped runs give 0.8171 in pooled warm AUC and 0.6165 in within-learner cold AUC, against 0.8166 (Table 1) and 0.6132 (Table 19) under the patience-5 rule; EoKT − EERNN-A is +0.0116 for the capped runs and +0.0122 under the rule, and the within-learner cold AUC difference EERNN-A − EoKT is −0.0324 and −0.0357. Under the 0.001-threshold rule, longer training therefore did not improve EERNN-A, although gradual gains smaller than the 0.001 threshold cannot be ruled out. The patience-5 rule also stops earlier than that of EoKT (patience 10, at most 200 epochs), so the comparison is not under the same stopping rule. EERNN-A also differs from EoKT in capacity, in excluding the time gap, and in training its text representation for KT instead of keeping it fixed. We therefore do not attribute the difference to the fixed general-purpose representation alone.

LKT is measured on the five-fold partition of this paper (Table 1). The predictor is hfl/chinese-roberta-wwm-ext (102M parameters, BERT-base structure); its input concatenates [CLS] KC text [SEP] question text per interaction up to 512 tokens, so on MOOCRadar the effective history is 6.7 interactions per window on average (median 7, p10 5, p90 9). Training uses AdamW (learning rate 2e-5) with patience 2; for each fold, masked-window validation AUC selects one of seeds 42/3407 (3407/42/3407/3407/42), and test scores never enter selection. All ten MOOCRadar runs stopped early after 5 to 10 epochs and took 3.8 to 7.5 hours each, 58 GPU-hours and about USD 60 in total (L4, On-Demand). Of the +0.1508 difference to EoKT (Tables 1 and 12), little is due to the shorter history: rescoring the EoKT checkpoints with only the 6 preceding responses, the median effective history of LKT, still gives 0.8851, a drop of 0.0160 (Table 11). Table 22 in Appendix F lists, next to the randomly initialised RoBERTa-base configuration (124.7M), the pretrained backbone of these MOOCRadar accuracy runs (102.3M) at their mean input length of 461 tokens (769 ms). The original paper (Lee et al., 2024) compares seven pretrained language models, and RoBERTa and DeBERTa-v3 differ by 0.0005 on XES3G5M-T; we run one RoBERTa-family model.

On XES3G5M, LKT reaches 0.7898 ± 0.0021 (Table 1); the paired difference EoKT − LKT is +0.0390 [+0.0382, +0.0398], and LKT is below qDKT and EERNN-A (Table 13). The history-length control in Table 11 is MOOCRadar only; its contribution to the XES3G5M difference remains unquantified. One epoch takes about 4.3 hours on XES3G5M, so each run had a 36-hour training-time cap; nine of the ten runs stopped at the cap after 7 epochs and are not converged (all five values behind Table 1 come from these nine; †), and one stopped by patience after 6 epochs.

In the first attempt, five of the ten runs stayed at AUC 0.5 on both training and validation from the first epoch, a constant predictor; only these five cells (seed 42 of folds 0 and 1, seed 3407 of fold 2, both seeds of fold 4) were rerun with a linear warm-up of 2,000 steps and a gradient-norm clip of 1.0, and all five reruns trained. The other five keep the first-attempt values without warm-up or clipping. Validation selection runs across rerun and first-attempt runs; of the five folds behind Table 1, folds 0 and 4 selected the rerun. Validation selected seeds 42/3407/42/3407/3407, the effective history is 7.0 interactions per window on average (median 7, p10 5, p90 9), and the mean input length is 470 tokens. XES3G5M cost 385 GPU-hours and about USD 400 in total (L4, On-Demand), of which the five collapsed runs account for 73 hours and about USD 76. The 0.8508 that the original paper (Lee et al., 2024) reports for LKT-RoBERTa on XES3G5M-T comes from a different split and training budget, so the two values are not directly comparable.

The per-learner state kept at inference is one of a single hidden state, a value memory, the last 200 interactions, or a text history. In every method the trained weights are shared across learners, and no reference implementation updates weights per response. A public SINKT implementation (Fu et al., 2024) is available at "tubehao/sinkt". EoKT has neither KC annotation nor a question-ID table, a new question enters the same input space with one embedding pass, and the per-learner state is one 512-dimensional GRU state. The table is limited to definitions and reference-implementation facts; it claims nothing about accuracy or training effort on new questions or datasets.

Table 24: Input requirements and per-learner inference state of each method

Method	KC annotation	Question-ID table	Unseen question	Per-learner state at inference
DKT	required	no	KC-level only	one RNN state
DKVMN	required	no	KC-level only	value memory (KCs $\times$ dim)
AKT	required	yes	no trained vector	last 200 interactions
simpleKT	required	yes	no trained vector	last 200 interactions
IEKT	required	yes	no trained vector	RNN state + per-step levels
qDKT	no	yes	no trained vector	one RNN state
KCGen-KT	generated per question	yes	after an LLM call	as the base model (LLM predictor)
KCluster	generated per question	yes	after an LLM call	as the base model
LKT	KC text used in the reported formulation	no	from text	text history ($\leq$ 512 tokens)
NTKT	optional (skill tags ablated)	optional (question IDs ablated)	from text (question and option texts)	text history in the LLM context
LLM-KT	required (concept-specific tokens)	yes (question-specific tokens)	no trained vector	sequence-model state injected into the LLM
EERNN	no	no	from text	one LSTM state (Markov form) or attention over past states
EKT	required (one-hot over the K concepts)	no	from text, once the question is tagged	one state matrix with one row per KC
EERNN-A	no	no	from text	past LSTM states and question vectors (window 200)
ESKT	no	no	from text	interaction history for Transformer-based KTAM
EoKT	no	no	from text	one GRU state (512-d)

H AUC Sensitivity to Response-noise Assumptions on XES3G5M

The upper end of Bayes-optimal AUC depends on assumptions about response noise. When a response is not determined by the learner's state alone, Bayes-optimal AUC can be below 1 (Ushio et al., 2026). Below, assuming a 7.1% probability of a wrong answer despite knowing the answer gives a conditional upper end of 0.8516 (about 0.85) on XES3G5M. This is an assumption-dependent reference, not an empirical estimate of the ceiling attainable by a predictor.

The assumption-dependent upper end of 0.85 is computed in five steps from answer formats, observed accuracy, and an empirical rate of wrong answers. Step 1 groups the responses by answer format and number of options: XES3G5M has 6,142 fill-in and 1,509 single-choice questions, and 69.6% of the responses fall on fill-in questions. Step 2 confines the probability η that a response in a group is correct to an interval between the guess rate l and an upper end $u = 1 - s$. Here l is the reciprocal of the number of options for single choice and 0 for fill-in. The rate s, at which a learner who knows the answer still responds wrongly, is applied uniformly to every group. We do not split s by answer format because the retry rate used in Step 5 is an aggregate that mixes forgetting, lucky guesses, and the choice of questions to retry, and a format-specific split would keep the same mixture; the sensitivity in Table 25 covers the uncertainty in its value. Step 3 takes the observed accuracy μ of each group as the mean of η in that group and imposes nothing else on the within-group distribution. Step 4 evaluates the Bayes-optimal AUC with the functional one half plus the expected absolute difference between two independent draws of η, divided by $4\theta(1-\theta)$, where θ is the overall accuracy 0.7947 (Ushio et al., 2026). The functional is smallest when every group collapses to μ and largest when each group splits its mass between l and u while keeping the mean at μ. The former gives the lower end and the latter the upper end. Step 5 fixes s. Among the 136,613 transitions in which the same learner answered the same question again after a correct response, 9,746 were wrong, a rate of 0.0713. Because that rate includes forgetting, lucky guesses, and the choice of which questions to retry, it is an empirical anchor rather than the rate of unpredictable wrong answers itself. We place 7.1% in the main scenario and vary s from 0 to 10% in Table 25 to show the sensitivity of the upper end. At 7.1% the interval is [0.5264, 0.8516], and 0.85 is this conditional upper end rounded.

Table 25: Bayes-optimal AUC interval on XES3G5M

Rate s of wrong answers	Lower end	Upper end
0	0.5264	0.9968
2%	0.5264	0.9581
3.55%	0.5264	0.9270
5%	0.5264	0.8969
7.1% (empirical anchor)	0.5264	0.8516
10%	0.5264	0.7855

I History-free Text Control and First versus Repeat Encounters

The history-free text control (TEXTONLY) maps fixed 3072-dimensional gemini-embedding-2 embeddings through a 512-unit hidden layer, GELU, dropout 0.1, and a scalar output layer. It has 1,573,889 trainable parameters and uses no response history, learner ID, time gap, KC, or availability flag. We trained ten runs per dataset with the same question holdout as Appendix C (10%, seed 20260912), five folds, and seeds 42/3407. Training uses Adam with learning rate 0.0003, batch size 64, at most 200 epochs, a validation-AUC improvement threshold of 0.001, and patience 10. No run is selected using test results.

A first encounter is the first occurrence of a learner–question pair in the complete test sequence. We include unscored initial responses when identifying encounters, then apply the scoring mask and holdout. A first encounter can still have a history of responses to other questions. We aligned all row keys, labels, scoring masks, and held-out questions across F3, I3, and TEXTONLY. Within-learner cold AUC gives equal weight to learners with both correct and incorrect responses in that subset. Responses from learners with only one label remain in pooled cold AUC. Table 27 gives the cohort sizes.

Table 26: Within-learner cold AUC on first and repeat encounters after question-holdout training

Dataset	Model	All cold	First	Repeat
MOOCRadar	EoKT (F3)	0.6041 ± 0.0118	0.6048 ± 0.0118	0.2875 ± 0.0910
MOOCRadar	Question-ID (I3)	0.5047 ± 0.0067	0.5045 ± 0.0067	0.2917 ± 0.0278
MOOCRadar	TEXTONLY	0.5669 ± 0.0026	0.5665 ± 0.0026	0.7292 ± 0.0354
XES3G5M	EoKT (F3)	0.6489 ± 0.0073	0.6425 ± 0.0069	0.7420 ± 0.0141
XES3G5M	Question-ID (I3)	0.5896 ± 0.0020	0.5851 ± 0.0021	0.6954 ± 0.0184
XES3G5M	TEXTONLY	0.5906 ± 0.0022	0.5879 ± 0.0021	0.5766 ± 0.0054

Table 27: Scored responses, learners, and eligible learners with both labels in each subset

Dataset	Subset	Responses	Learners	Eligible
MOOCRadar	All cold	18,624	2,214	1,014
MOOCRadar	First	18,563	2,214	1,013
MOOCRadar	Repeat	61	19	4
XES3G5M	All cold	103,134	3,613	3,388
XES3G5M	First	97,096	3,613	3,384
XES3G5M	Repeat	6,038	1,964	339

Table 26 is descriptive. On first encounters, within-learner cold AUC is 0.6048 for EoKT versus 0.5665 for TEXTONLY on MOOCRadar and 0.6425 versus 0.5879 on XES3G5M. TEXTONLY gives the same prediction for the same question, an invariance checked in the audit. F3 and I3 are re-scored from saved runs; TEXTONLY was trained from a separate code bundle. F3 and TEXTONLY differ in capacity, IRB, and time-gap inputs as well as history, so their difference is not the causal effect of history alone. The four eligible MOOCRadar learners do not support a general conclusion about repeat encounters. Test data are shared across runs, so the ten runs are not independent samples. Experiment records also retain each seed's five-fold mean and sample SD, differences paired by fold and seed, and pooled cold AUC.