# RECAST: LEARNING-SCIENCE INDUCTIVE BIASES FOR KNOWLEDGE TRACING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Knowledge tracing predicts whether a learner will answer the next item correctly from their interaction history. Transformer KT models are strong sequence predictors, yet they typically treat tutoring logs as generic tokens and rely on Softmax attention to rediscover structure that learning science already identifies as central: last-attempt retrieval, local response persistence, spacing-dependent forgetting, and individualised cross-skill transfer. We encode these regularities as explicit inductive biases rather than incidental byproducts of attention. In this paper, we propose REcency- and Concept-lag Aware Skill Tracing (RECAST), a dual-stream causal Transformer with a Skill Recency Bank for last same-skill outcome, lag, and visit count; a Local Markov state for streaks and skill switches; concept-conditioned lag biases with distinct same- vs. cross-skill schedules; and a student factor $U$ that gates a mixture of low-rank transfer graphs. On five pyKT benchmarks with 5-fold evaluation, RECAST obtains the best AUC on four datasets and the best ACC on three against recurrent, memory, and Transformer baselines, with the largest gains when learners frequently revisit the same skill and competitive accuracy under heavily interleaved practice.

## 1 INTRODUCTION

Knowledge tracing (KT) predicts whether a learner will answer the next item correctly from their interaction history. It is a core inference problem behind adaptive practice in intelligent tutoring systems: estimates of what a learner currently knows drive item selection, review scheduling, and mastery decisions (Corbett & Anderson, 1994; Piech et al., 2015). Classical approaches model mastery as a latent Markov state (Corbett & Anderson, 1994; Anderson et al., 1995), while modern deep KT casts the same next-step objective as sequence prediction with recurrent networks (Piech et al., 2015; Yeung & Yeung, 2018), key–value memories (Zhang et al., 2017), and Transformers (Pandey & Karypis, 2019; Ghosh et al., 2020; Liu et al., 2023; Shen et al., 2022). Further gains have come from difficulty modelling, denoising, forgetting kernels, and mixture-of-concept structure (Liu et al., 2022; Guo et al., 2025). On public benchmarks these models are strong predictors, yet accuracy alone does not specify what the representation should track about human learning.

Successful retrieval changes the odds of later retrieval—the testing effect (Roediger III & Karpicke, 2006). Responses show local persistence and switching costs under desirable difficulties (Bjork et al., 1992). Retention depends on spacing since last practice (Cepeda et al., 2006; Ebbinghaus, 2013), and same-skill review need not share a forgetting schedule with cross-skill interference. Prior work on skill $i$ can help or hinder skill $j$, with strength that varies across individuals (Singley & Anderson, 1989). These regularities describe how people practise. A KT architecture can either encode them as explicit inductive biases, or rely on Softmax attention to rediscover them under label noise, irregular curricula, and early stopping—which is difficult to verify when the relevant quantities never appear as named components (Khajah et al., 2016).

Current attentive KT leaves three mechanisms under-specified. First, last attempt retrieval is discrete: whether the previous attempt on a skill succeeded is not interchangeable with a soft average over all past practice on that skill. Blending a recent failure into older successes attenuates the contrast that retrieval theory treats as primary; see Section 4.1. Second, spacing is concept-conditioned: same-skill review and cross-skill interference are different cognitive processes, yet a shared temporal kernel requires attention to recover this partition from content embeddings alone.



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

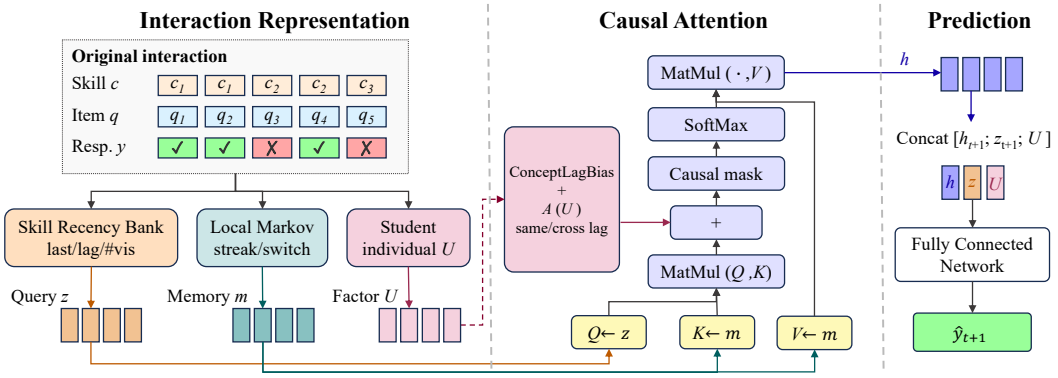



Figure 1: Overview of RECAST. *Left:* Skill Recency Bank and Local Markov construct query $z$ and memory $m$; student factor $U$ summarises the learner as a *causal prefix* pool (only tokens $\leq t$ enter $U_t$). *Middle:* causal attention with ConceptLagBias and transfer prior $A(U)$. *Right:* clean prediction of $\hat{y}_{t+1}$.

Third, transfer is individual: the same curriculum induces different cross-skill influence across learners, which content-only attention does not parameterise. Orthogonal modelling choices, generic temporal smoothing, denoising, or clock-time decay, can raise aggregate AUC while leaving these mechanisms unnamed, and therefore hard to ablate, interpret, or connect back to learning science.

We argue that human learning behaviors should drive KT architecture design, rather than define the resulting inductive biases. In this paper, we propose REcency- and Concept-lag Aware Skill Tracing (RECAST), a dual-stream causal Transformer that encodes four behaviors as named modules in Figure 1:

- a **Skill Recency Bank** that injects last same-skill outcome, lag, and visit count into the query stream to support retrieval and spacing;
- a **Local Markov** state that tracks response streaks and skill switches in memory to expose persistence;
- a **concept-conditioned lag bias** that assigns distinct temporal schedules to same- vs. cross-skill keys in attention logits for differential forgetting;
- a **student factor** $U$ that gates a mixture of low-rank transfer graphs for individualised transfer.

Our novelty is not a larger Transformer or a new aggregation of existing KT components. It is a shift in design target: last-attempt retrieval, concept-conditioned forgetting, local persistence, and individualised transfer are encoded as named inductive biases, rather than left for Softmax to rediscover from content embeddings alone. Our contributions are as follows. (1) We formulate hard last-attempt recency as an inductive bias, give a sufficiency argument under a per-skill Markov assumption, and prove that EWMA-style soft summaries attenuate the recoverable last-attempt contrast. (2) We instantiate the four behaviors above in a single dual-stream architecture that keeps a clean prediction head and makes each module independently ablatable. (3) Under the pyKT protocol (Liu et al., 2022), RECAST matches or exceeds a broad suite of recurrent, memory, and Transformer baselines on five public benchmarks, with the largest gains where learners frequently re-retrieve the same skill and competitive accuracy under interleaved practice—a boundary condition predicted by retrieval theory rather than a dataset-specific failure. Soft substitutes for hard recency do not close the gap.

## 2 Related Work

**Classical and deep knowledge tracing.** Classical KT models mastery as a latent Markov state (Corbett & Anderson, 1994; Anderson et al., 1995), with extensions for difficulty, ability, and learner-specific transitions (Pardos & Heffernan, 2011; Yudelson et al., 2013). Deep models treat

the same next-step objective as sequence prediction: DKT and DKT+ (Piech et al., 2015; Yeung & Yeung, 2018), DKVMN (Zhang et al., 2017), and Transformers such as SAKT, AKT, and simpleKT (Pandey & Karypis, 2019; Ghosh et al., 2020; Liu et al., 2023), later enriched by difficulty modelling and denoising (Shen et al., 2022; Deng et al., 2025; Guo et al., 2025). On pyKT (Liu et al., 2022), these systems are strong predictors, but they rely primarily on architectural capacity: the network must discover which aspects of the interaction stream matter for retrieval, spacing, and transfer. RECAST keeps a competitive Transformer backbone and instead names those aspects as modules that can be inspected and ablated.

**Temporal dynamics and relational structure.** Retention decays with time and benefits from spaced practice (Ebbinghaus, 2013; Cepeda et al., 2006); KT models inject time gaps or decay kernels into recurrent and attentive architectures (Nagatani et al., 2019; Ghosh et al., 2020). A shared temporal kernel over attended tokens conflates same-skill review with cross-skill interference, even though the two processes need not share a schedule. Prerequisite graphs and relation-aware attention encode skill–skill links (Chen et al., 2018; Pandey & Srivastava, 2020; Tong et al., 2020), recognising that prior work on skill $i$ can help or hinder skill $j$ (Singley & Anderson, 1989). Such graphs are typically static and shared across learners. RECAST instead uses concept-conditioned lag biases—so same- and cross-skill keys can follow different forgetting schedules—and a student-gated mixture of low-rank transfer graphs (Gong et al., 2024) so that transfer strength is individualised.

**Learning science and sequence models.** Testing effects, desirable difficulties, spacing, and transfer are foundational in learning science (Roediger III & Karpicke, 2006; Bjork et al., 1992; Cepeda et al., 2006). Flexible sequence models can fit educational interaction streams well—sometimes matching classical cognitive models (Khajah et al., 2016)—without exposing those cognitive quantities in the architecture. Soft temporal summaries such as EWMA over skill history improve smoothness but attenuate the last-attempt success/failure contrast that retrieval theory treats as primary. RECAST retains a dual-stream causal Transformer and adds named modules for retrieval, persistence, differential forgetting, and individualised transfer, so that gains can be attributed to these behavioral mechanisms rather than to unspecified capacity alone.

## 3 Problem Formulation

Each learner yields $\{(c_t, q_t, y_t)\}_{t=1}^{T}$ with skill $c_t$, optional item $q_t$, and correctness $y_t \in \{0, 1\}$. At step $t$ the model predicts $\hat{y}_t = \Pr(y_t{=}1 \mid \mathcal{H}_{<t}, c_t, q_t)$ under the next-step protocol (Liu et al., 2022). The history $\mathcal{H}_{<t}$ contains past skills, items, and outcomes. The target skill and item at $t$ are observed before the prediction, while $y_t$ is not. We evaluate ranking and thresholded accuracy of $\hat{y}_t$ using the standard pyKT tool (Liu et al., 2022).

## 4 Method

RECAST is a dual-stream causal Transformer whose components encode four learning behaviors—last-attempt retrieval and spacing (Roediger III & Karpicke, 2006; Cepeda et al., 2006), response persistence and switching (Bjork et al., 1992), and individualised transfer (Singley & Anderson, 1989), as named inductive biases rather than as byproducts of Softmax attention. We first formalise the hard-recency claim that motivates the Skill Recency Bank, then instantiate the architecture in the order of Figure 1: interaction representation, causal attention, and prediction.

### 4.1 Hard last-attempt recency

We make the hard-recency claim precise under a minimal per-skill Markov assumption familiar from classical BKT-style reasoning, then show why soft summaries are a weak substitute for hard last-attempt features.

**Assumption 1** (Per-skill last-outcome Markov)**.** For each skill $c$, when $c_t{=}c$ and a previous visit $\tau_t(c)$ exists, $y_t \mid \mathcal{H}_{<t}, c_t$ depends on history only through $(y_{\tau_t(c)}, \mathrm{lag}_t)$.

**Proposition 1** (Sufficiency of Skill Recency features)**.** *Under Assumption 1, the Skill Recency Bank inputs* $(c_t, \ell_t, \mathrm{lag}_t)$ *are sufficient for predicting* $y_t$ *among measurable functions of* $(\mathcal{H}_{<t}, c_t)$.

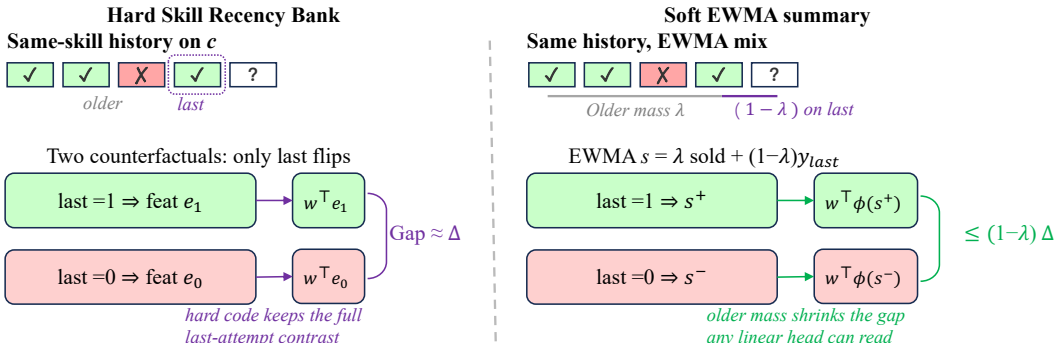



Figure 2: Hard vs. soft recency under a last-attempt flip; see Proposition 2. Both panels share the same same-skill history and differ only in the last outcome. *Left:* the bank exposes a discrete last-attempt code, so a linear readout retains gap $\approx \Delta$. *Right:* an EWMA mixes older mass $\lambda$, attenuating the readable gap to at most $(1-\lambda)\Delta$.

**Proposition 2** (Soft averaging attenuates carry-over)**.** *Fix a skill $c$ and two histories that differ only in the last same-skill outcome, inducing a Bayes score gap $\Delta > 0$ for a linear readout of that outcome. An EWMA that mixes the last outcome with older mass weighted $\lambda \in (0,1)$ shrinks the gap available to any linear readout of the EWMA to at most $(1-\lambda)\Delta$.*

**Proof sketch of Proposition 1.** Let $\tau = \tau_t(c)$ be the most recent prior time with $c_\tau = c$, or a cold-start symbol if none exists. Assumption 1 states that $\Pr(y_t{=}1 \mid \mathcal{H}_{<t}, c_t = c) = g(y_\tau, \mathrm{lag}_t)$ for some $g$. The bank encodes $y_\tau$ injectively, with an unknown atom for cold start, and represents $\mathrm{lag}_t$ by a measurable bucket map $\beta$; visit counts are a deterministic function of the same-skill subsequence. Any predictor of $y_t$ from $(\mathcal{H}_{<t}, c)$ can therefore be rewritten as a function of the bank features alone.

**Proof sketch of Proposition 2.** Write the EWMA update as $s_t = (1-\lambda)\, y_{\text{last}} + \lambda\, s_{\text{older}}$ with $\lambda \in (0,1)$ and $s_{\text{older}}$ identical under the two histories. Flipping $y_{\text{last}} \in \{0,1\}$ changes $s_t$ by at most $1-\lambda$. A linear readout of $s_t$ therefore retains at most a $(1-\lambda)$ fraction of the gap $\Delta$ available from reading $y_{\text{last}}$ directly. Recovering the full Bayes gap would require amplifying older residuals, reintroducing noise that the hard bank never mixes in.

**Remark.** If same-skill and cross-skill tokens obey different lag–logit maps, a shared temporal kernel is misspecified. Concept-conditioned biases $f_{\text{same}}$ and $f_{\text{cross}}$ restore the partition in the attention logits with $O(1)$ parameters per head.

Figure 2 illustrates the hard-vs-soft mechanism: hard bank features preserve the retrieval contrast required by testing effects, whereas EWMA-style summaries attenuate it. Concept-lag biases play a complementary role: they allow attention logits to use different forgetting schedules for same-skill review and cross-skill tokens, so Softmax need not rediscover that partition from embeddings alone.

### 4.2 Interaction Representation

RECAST separates a response-shifted memory stream from a concept/item query stream, with causal self- and cross-attention using rotary positions and SwiGLU. Figure 1 summarises the pipeline. The separation implements a division of labour that matches how tutoring logs are read: memory should carry recent interaction dynamics—what just happened, whether a streak is ongoing, whether the skill changed—while the query should carry the learner's standing on the skill being asked now, including whether the last attempt on that skill succeeded and how long ago it occurred. Without this split, a single stream must serve both roles, and Softmax attention is left to rediscover which tokens encode retrieval versus local persistence. We initialise

$$m_t^{(0)} = e_t^c + e_t^q + e_{t-1}^y + \mathrm{LocalMarkov}_t, \quad z_t^{(0)} = e_t^c + e_t^q + \mathrm{SkillBank}_t. \tag{1}$$

**Skill Recency Bank.** Testing-effect research shows that successful retrieval changes the odds of later retrieval on the same material (Roediger III & Karpicke, 2006): whether the previous attempt

on skill $c$ was correct is a first-order cue for the next $c$-attempt, not a detail that should be diluted into a long-run average. Spacing and practice amount further modulate retention (Cepeda et al., 2006; Ebbinghaus, 2013), so lag since the last visit and how often $c$ has been practised are natural companions to that last outcome. Attentive KT typically exposes only skill/item/response embeddings and relies on Softmax to recover these quantities; soft summaries such as EWMA mix a recent failure into older successes and attenuate the contrast that retrieval theory treats as primary (Proposition 2, Figure 2). We therefore maintain a causal per-skill bank that is updated *after* step $t$ but read *before* $y_t$ is observed, injecting into the query (i) last same-skill outcome $\in \{\text{unk}, \text{wrong}, \text{correct}\}$, (ii) lag since the previous visit, and (iii) visit count, projected by a small MLP. The unknown atom handles cold start when $c$ has never been seen. This design makes last-attempt retrieval an explicit inductive bias on the query stream: spacing enters without requiring wall-clock timestamps, and the discrete outcome code is never collapsed into a scalar average.

**Local Markov.** Beyond per-skill retrieval, learners exhibit short-horizon response structure: runs of correct or incorrect answers, and costs when the practice focus switches skills (Bjork et al., 1992). These effects are local in the token stream—often visible at lag 1—and are easy for a generic sequence model to under-represent when the only response signal is a single previous-outcome embedding buried among skill and item features. Classical BKT-style models treat mastery transitions as Markovian; we adopt a related but *local* inductive bias on the memory stream rather than a latent mastery HMM. Memory is therefore augmented with embeddings of $y_{t-1}$, a capped same-response streak length, and a skill-switch indicator $\mathbf{1}[c_t \neq c_{t-1}]$. Streak length is capped so that very long runs do not dominate capacity; the switch bit marks a change of practice focus independently of correctness. Together, these features make lag-1 persistence and switching directly available to attention, complementary to the Skill Recency Bank: the bank answers "what happened last time on *this* skill?", while Local Markov answers "what is happening in the immediate interaction neighbourhood?".

**Student factor $U$.** The same curriculum induces different cross-skill influence across learners (Singley & Anderson, 1989): transfer is individual, not only content-dependent. A model that conditions only on skill/item tokens must absorb learner variability into shared embeddings or into a shallow output bias, which neither parameterises transfer graphs nor supports FiLM-style modulation of temporal biases. We therefore pool valid memory tokens into a sequence-level student factor $U$, represented as a Gaussian posterior with a KL penalty toward $\mathcal{N}(0, I)$ so that $U$ remains a compact, regularised summary rather than an unconstrained per-sequence code. Pooling is strictly causal: at prediction time $t$ the factor is the *prefix* summary $U_t$ over memory tokens $\leq t$ (a running average), never over future positions $s{>}t$. $U$ is used in two places that match its semantics: FiLM modulation of intermediate features, and gating of the low-rank transfer mixture $A(U)$ in Section 4.3. Individualisation is thus a first-class parameter of how skills influence each other and how lag biases are shaped, not only of the final prediction head.

### 4.3 Causal attention

Same-skill review and cross-skill interference need not share a forgetting schedule. We therefore add concept-conditioned lag biases to the attention logits,

$$b_{t,s} = \mathbf{1}[c_t{=}c_s]\, f_{\text{same}}(\log(1+\ell_{t,s}), U) + \mathbf{1}[c_t{\neq}c_s]\, f_{\text{cross}}(\log(1+\ell_{t,s}), U), \tag{2}$$

with $\ell_{t,s} = |t-s|$ and small MLPs $f_{\text{same}}, f_{\text{cross}}$. This lets Softmax use distinct temporal schedules without rediscovering the same-/cross-skill partition from content embeddings. Complementing same-skill recency, prior work on skill $i$ can help or hinder skill $j$ with learner-dependent strength (Singley & Anderson, 1989). We add a student-gated transfer prior inspired by sparse temporal dependence (Gong et al., 2024), $A(U) = \sum_{k=1}^{K} \pi_k(U) A_k$ with low-rank factors $A_k(i \to j) = \text{softplus}(\langle s_{k,i}, t_{k,j} \rangle)$ and $\pi = \text{softmax}(W_\pi U)$, entering logits as $\text{scale} \cdot \log A_{c_s \to c_t}$. Sparse $L_1$ regularisation encourages selective transfer.

### 4.4 Prediction

A concept memory stores reliability-weighted skill slots, down-weighting likely slips/guesses during writes, but is not mixed into the prediction head via a slip channel: we keep a clean sigmoid so

Table 1: Dataset statistics and practice-regime metrics from pyKT `train_valid` splits. $\Delta_{\text{same}}$ denotes the accuracy difference after a previous same-skill correct versus incorrect response, and $f_{\text{lag}=1}$ represents the fraction of same-skill revisits at lag 1.

| | Dataset Statistics | | | | Practice-Regime Statistics | | |
|---|---|---|---|---|---|---|---|
| Dataset | Students | Skills | Questions | Interactions | Median lag | $f_{\text{lag}=1}$ | $\Delta_{\text{same}}$ |
| ASSIST2009 | 3,082 | 120 | 17,405 | 274,614 | 1 | 0.60 | 0.278 |
| ASSIST2015 | 15,275 | 100 | — | 544,339 | 1 | 0.93 | 0.218 |
| NIPS Task34 | 3,935 | 57 | 948 | 1,123,383 | 1 | 0.63 | 0.228 |
| OLI Statics | 454 | 1,353 | 1,353 | 184,737 | 16 | 0.12 | 0.138 |
| EdNet | 3,750 | 188 | 11,784 | 1,071,464 | 10 | 0.02 | 0.116 |

observational noise does not redefine the decision target,

$$\hat{y}_t = \sigma\big(f([h_t; e_t^q; U]) - b_{q_t}\big). \tag{3}$$

Training minimises BCE plus KL and graph sparsity,

$$\mathcal{L} = \text{BCE} + \lambda_{\text{KL}}\text{KL}(q(U)\|p(U)) + \lambda_{\text{sp}}\|A\|_1. \tag{4}$$

# 5 Experiments

Our experiments evaluate three claims. (Q1) Encoding retrieval, spacing, persistence, and transfer as architectural components improves next-step prediction against a broad suite of established KT models. (Q2) Hard last-attempt features are necessary; soft temporal summaries are not substitutes. (Q3) Gains concentrate in learning situations where those human behaviors are operative—not as artefacts of particular datasets.

## 5.1 Setup

We evaluate on ASSIST2009, ASSIST2015, NIPS Task 34, OLI Statics, and EdNet under the pyKT protocol (Liu et al., 2022): sequence length 200, and concept-level AUC/ACC; Table 1 summarises the logs. Baselines span recurrent, memory, and Transformer families: DKT (Piech et al., 2015), DKT+ (Yeung & Yeung, 2018), DKVMN (Zhang et al., 2017), SAKT (Pandey & Karypis, 2019), AKT (Ghosh et al., 2020), simpleKT (Liu et al., 2023), DIMKT (Shen et al., 2022), DenoiseKT (Deng et al., 2025), MoC-KT (Hou et al., 2026), FA-KT (Bai et al., 2026), and RobustKT (Guo et al., 2025). DIMKT/DenoiseKT require question ids and FA-KT requires timestamps; unsupported entries on concept-only ASSIST2015 are marked "—". Unless noted otherwise, RECAST uses width 256, 8/6 memory/query layers, 4 heads, student-factor dimension 16, $K{=}4$ transfer graphs of rank 32, streak cap 8, Adam at $10^{-4}$, batch size 24–48, dropout 0.2, and early stopping on validation AUC with patience 20. The behavioral modules in Section 4 are enabled; soft LocalContinuity, skill EWMA, neural clock kernels, and prediction-head noise channels are disabled.

## 5.2 Main results

Tables 2–3 report 5-fold mean±std against the baseline suite. RECAST obtains the best AUC on four of five benchmarks and the best ACC on three, and remains competitive wherever it is not first. Rather than attributing performance differences to dataset identity, we interpret the pattern through the practice regimes summarised in Table 1: how often learners revisit the same skill after one step, and how strongly previous same-skill success predicts the next attempt.

**Where same-skill retrieval is frequent.** ASSIST2015 and NIPS34 exhibit high immediate retrieval ($f_{\text{lag}=1}{=}0.93$ and $0.63$) with substantial same-skill carry-over ($\Delta_{\text{same}}{=}0.218$ and $0.228$). In these regimes last-attempt outcome and lag are locally informative, and RECAST's largest gains appear: $+0.088$ AUC over DKT+ on ASSIST2015 and $+0.019$ over FA-KT on NIPS34. ASSIST2009 lies between these extremes ($f_{\text{lag}=1}{=}0.60$) and yields a smaller but positive margin over RobustKT ($+0.004$ AUC). The ordering matches the inductive bias: hard Skill Recency and concept-conditioned lag are most useful when same-skill history is repeatedly queried.

Table 2: 5-fold test AUC, mean±std. Bold: best in column. — : unsupported.

| Model | ASSIST2009 | NIPS34 | OLI | ASSIST2015 | EdNet |
|---|---|---|---|---|---|
| DKT | .8233±.0020 | .7713±.0004 | .8801±.0023 | .7266±.0005 | .9253±.0006 |
| DKT+ | .8255±.0007 | .7716±.0004 | **.8819±.0005** | .7282±.0007 | .9263±.0004 |
| DKVMN | .8170±.0011 | .7690±.0006 | .8506±.0034 | .7205±.0007 | .9252±.0006 |
| SAKT | .7536±.0011 | .7472±.0006 | .8022±.0023 | .7041±.0005 | .9184±.0005 |
| AKT | .8440±.0045 | .8044±.0012 | .8781±.0017 | .7259±.0004 | .9432±.0004 |
| simpleKT | .8376±.0012 | .8035±.0006 | .8563±.0016 | .7239±.0006 | .9409±.0004 |
| DIMKT | .8319±.0016 | .8036±.0005 | .8785±.0013 | — | .9403±.0007 |
| DenoiseKT | .7799±.0043 | .7982±.0006 | .8636±.0023 | — | .7281±.0009 |
| MoC-KT | .8360±.0011 | .8032±.0004 | .8636±.0014 | .7252±.0009 | .9279±.0002 |
| FA-KT | .8434±.0005 | .8056±.0004 | .8708±.0005 | — | .9430±.0004 |
| RobustKT | .8441±.0009 | .8049±.0005 | .8780±.0015 | .7270±.0006 | .9420±.0009 |
| **RECAST** | **.8481±.0018** | **.8245±.0003** | .8750±.0017 | **.8162±.0085** | **.9481±.0002** |

Table 3: 5-fold test ACC, mean±std. Bold: best in column.

| Model | ASSIST2009 | NIPS34 | OLI | ASSIST2015 | EdNet |
|---|---|---|---|---|---|
| DKT | .7673±.0019 | .7050±.0004 | .8450±.0010 | .7506±.0005 | .8550±.0004 |
| DKT+ | .7692±.0009 | .7046±.0004 | **.8464±.0010** | .7512±.0002 | .8568±.0006 |
| DKVMN | .7621±.0009 | .7024±.0006 | .8305±.0025 | .7504±.0003 | .8538±.0010 |
| SAKT | .7229±.0013 | .6834±.0005 | .8066±.0025 | .7455±.0004 | .8491±.0008 |
| AKT | .7782±.0026 | .7327±.0009 | .8424±.0026 | .7511±.0003 | .8708±.0009 |
| simpleKT | .7735±.0007 | .7321±.0003 | .8312±.0019 | .7505±.0001 | .8669±.0024 |
| DIMKT | .7720±.0017 | .7314±.0007 | .8424±.0010 | — | .8665±.0029 |
| DenoiseKT | .7336±.0036 | .7272±.0011 | .8360±.0014 | — | .7061±.0039 |
| MoC-KT | .7721±.0014 | .7322±.0002 | .8334±.0017 | .7502±.0004 | .8576±.0003 |
| FA-KT | .7768±.0014 | .7348±.0004 | .8360±.0016 | — | .8701±.0005 |
| RobustKT | **.7813±.0011** | .7329±.0007 | .8414±.0021 | .7512±.0004 | .8690±.0018 |
| **RECAST** | .7759±.0029 | **.7480±.0003** | .8384±.0011 | **.7811±.0036** | **.8753±.0004** |

**Where practice is heavily interleaved.** OLI Statics and EdNet present the complementary regime. OLI combines a large skill vocabulary (1,353 concepts) with few learners (454), a median same-skill lag of 16, and only 12% lag-1 revisits (Table 1), so the Skill Recency Bank is often in a cold-start state. DKT+ then leads (0.8819 vs. 0.8750 AUC; 0.8464 vs. 0.8384 ACC). EdNet is likewise interleaved ($f_{\text{lag}=1}$=0.02), yet RECAST remains first by a small margin over AKT (+0.005 AUC): the modules are not harmful when recency is rare, only less decisive. Section 5.6 analyses the OLI gap in more detail; we treat it as a predicted boundary condition of retrieval-centric modelling.

### 5.3 Soft vs. hard ablations

Proposition 2 predicts that soft temporal summaries cannot substitute for hard last-attempt features. Table 4 tests this on ASSIST2009: a dual-stream backbone, soft continuity/EWMA substitutes, and the full hard-recency stack. Soft LocalContinuity and skill EWMA, as well as a continuity-only attention bias, remain in the 0.835–0.837 range and are indistinguishable from the backbone, whereas the full hard-recency model reaches 0.850. The gap is not explained by additional capacity alone: the soft substitutes add temporal summary features to the same dual-stream Transformer, yet still fail to recover the discrete last-attempt contrast. Soft smoothing therefore improves neither aggregate fit nor the retrieval signal emphasised by testing-effect accounts.

### 5.4 Module ablations

Table 5 isolates each RECAST module on ASSIST2009 under matched hyperparameters as 5-fold mean±std. Leave-one removals indicate complementarity rather than redundancy: collapsing to the backbone costs $\Delta\approx 0.016$ AUC, removing the student factor $U$ costs $\Delta\approx 0.008$, and Local Markov yields the next largest drop at $\Delta\approx 0.006$; bank, ConceptLagBias, and transfer-mixture removals are smaller but non-positive relative to the full model. The modest bank-only drop on ASSIST2009 does not contradict Table 4: soft substitutes never approach the hard stack, while leave-one starts from an already equipped model in which other modules can partially compensate.

Table 4: ASSIST2009 soft vs. hard ablations, test AUC. Soft continuity/EWMA does not recover the last-attempt contrast of hard Skill Recency.

| Variant | Test AUC |
|---|---|
| Dual-stream Transformer, no bank | $0.8371 \pm .0013$ |
| + LocalContinuity only | $0.8360 \pm .0011$ |
| + EWMA only, small init | $0.8370 \pm .0010$ |
| + LocalContinuity + EWMA | $0.8355 \pm .0010$ |
| + continuity attention bias only | $0.8369 \pm .0016$ |
| **RECAST, full hard bank + lag bias** | $\mathbf{0.8481 \pm .0018}$ |

Table 5: Leave-one-module test AUC on ASSIST2009, 5-fold mean±std.

| Variant | Test AUC |
|---|---|
| RECAST, full | $\mathbf{.8481 \pm .0018}$ |
| w/o Skill Recency Bank | $.8458 \pm .0016$ |
| w/o Local Markov | $.8407 \pm .0017$ |
| w/o ConceptLagBias | $.8458 \pm .0017$ |
| w/o transfer mixture $A(U)$ | $.8465 \pm .0017$ |
| w/o student factor $U$ | $.8387 \pm .0010$ |
| backbone, all off | $.8309 \pm .0010$ |

### 5.5 Behavioral diagnostics

Beyond aggregate AUC, Figure 3 conditions performance on the learner's *behavioral state* at prediction time rather than on dataset membership. By spacing since the last same-skill attempt, AUC is highest after short but non-immediate spacing at lags 2–7 and remains strong on cold starts near $\mathrm{AUC} \approx 0.85$, while immediate re-attempts at lag 1 are hardest near $\mathrm{AUC} \approx 0.80$. This matches response persistence: lag-1 streaks are locally noisy and remain a difficult regime even with Local Markov features. By last same-skill outcome, accuracy is higher after a previous correct attempt than after a failure, consistent with the retrieval asymmetry of testing effects, while ranking AUC remains strong in both cases. Sequence position is comparatively flat, so these patterns are unlikely to be artefacts of long histories alone. Figure 4 diagnoses the fitted ConceptLagBias on an illustrative ASSIST2009 window, complementing the performance conditioning above with a mechanism view of how spacing enters attention. Figure 4 (a) plots causal bias logits $b_{t,s}$ over query time $t$ and past key time $s$, with the upper triangle masked so only admissible past keys appear; the pattern is not uniform across lags, so temporal structure is carried by the bias rather than absorbed entirely into content similarity. Figure 4 (b) marks which pairs are same-skill versus cross-skill, i.e. which schedule $f_{\mathrm{same}}$ or $f_{\mathrm{cross}}$ applies at each $(t, s)$. Figure 4 (c) aggregates mean bias against lag $\ell{=}t{-}s$ under those two masks: the same-skill and cross-skill curves diverge, so Softmax receives distinct decay schedules for review of the queried skill and for interference from other skills.

### 5.6 When recency helps

If RECAST's modules encode retrieval and spacing, their benefit should track whether those behaviors are locally available (Roediger III & Karpicke, 2006; Cepeda et al., 2006). Comparing RECAST to the best non-RECAST baseline on each benchmark in Table 2, relative to the immediate re-retrieval rate $f_{\mathrm{lag}=1}$ in Table 1, yields a consistent ordering: large gains where re-retrieval is common—ASSIST2015 at $+0.088$ vs. DKT+ ($f{=}0.93$) and NIPS34 at $+0.019$ vs. FAKT ($f{=}0.63$)—and small or negative gains under interleaved practice—EdNet at $+0.005$ vs. AKT ($f{=}0.02$) and OLI at $-0.007$ vs. DKT+ ($f{=}0.12$). ASSIST2009 lies between these extremes at $+0.004$ vs. RobustKT ($f{=}0.60$).

The OLI gap admits a more detailed account. First, sparse same-skill revisits mean that last-attempt codes and short lags (the inputs the Skill Recency Bank is designed to expose) are infrequently observed at prediction time, so the associated parameters receive less effective supervision than on ASSIST2015. Second, a 1,353-skill transfer mixture gated by a sequence-level student factor is difficult to identify from only 454 learners; under-determined transfer can add variance without improving next-step ranking. Third, DKT+ regularises recurrent state transitions toward smooth

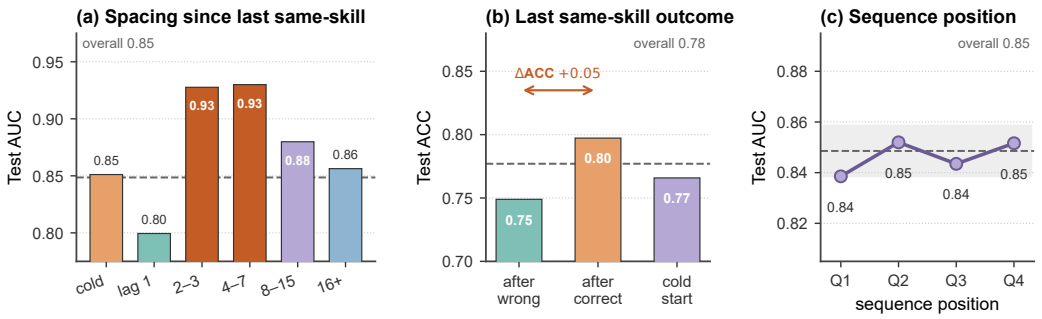



Figure 3: Window-test metrics conditioned on the learner's behavioral state on ASSIST2009. *(a)* AUC by spacing since the last same-skill attempt: medium spacing is easiest; immediate re-attempts at lag 1 remain hardest. *(b)* ACC by last same-skill outcome: higher after a previous correct than after a failure. *(c)* AUC by sequence-position quartile is comparatively flat, so the patterns in (a)–(b) are not artefacts of long histories alone. Dashed lines mark overall performance.

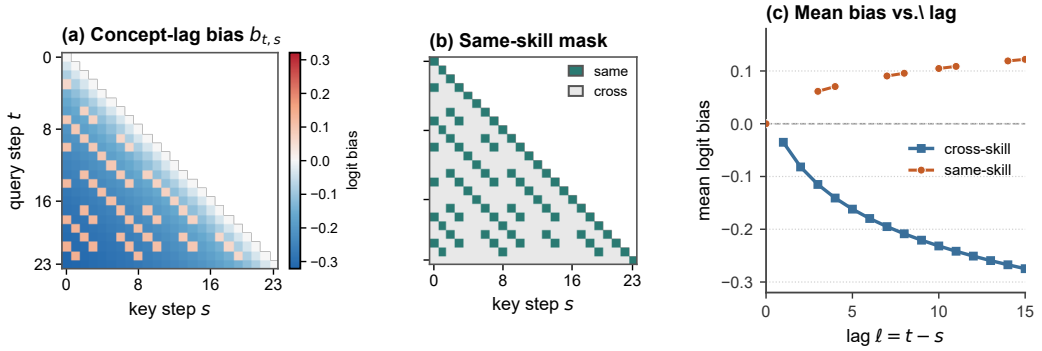



Figure 4: Concept-conditioned lag structure in a fitted RECAST model on an illustrative window. *(a)* Causal ConceptLagBias logits $b_{t,s}$ with the upper triangle masked. *(b)* Same-skill vs. cross-skill mask. *(c)* Mean bias vs. lag $\ell{=}t{-}s$, aggregated separately for same-skill and cross-skill pairs: the two schedules diverge, as required by differential forgetting. This is a mechanism diagnosis, not a model comparison.

local dynamics, which aligns with OLI's interleaved token stream even when concept-level recency is rare. Taken together, these observations explain why a retrieval-centric architecture can trail a strong recurrent regulariser on OLI while remaining competitive with attentive and memory baselines on the same log, and while leading where same-skill retrieval is the dominant cue. The ranking therefore tracks practice regime rather than dataset label.

## 6 Conclusion

We presented RECAST, a dual-stream knowledge tracing model whose components are derived from human learning behaviors, last-attempt retrieval, spacing, response persistence, and student-modulated transfer. Encoding these regularities as explicit inductive biases addresses modelling gaps left by generic attentive KT: hard retrieval contrast, concept-conditioned forgetting, and individualised transfer. On standard benchmarks, RECAST matches or exceeds a broad baseline suite, with the largest gains where learners repeatedly re-retrieve the same skill, and with a clear boundary under interleaved practice typified by OLI Statics. Soft substitutes do not close the hard-recency gap, and leave-one ablations indicate complementary module contributions. Practice regime, not merely dataset identity, determines when the prior helps.

### AI USE STATEMENT

(This section is **required** and does not count toward the page limit.)

In this work, we used generative AI tools for manuscript language polishing and assistance in identifying relevant related work. We have not used generative AI tools for generating research results, conducting experiments, analyzing data, or developing scientific claims; the remaining disclosure tasks are not applicable. We have reviewed all AI-assisted work, verified the accuracy of the relevant literature, and revised the AI-assisted text as necessary. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

### ACKNOWLEDGMENTS

Use unnumbered third level headings for the acknowledgments. All acknowledgments, including those to funding agencies, go at the end of the paper.

## REFERENCES

John R Anderson, Albert T Corbett, Kenneth R Koedinger, and Ray Pelletier. Cognitive tutors: Lessons learned. *The journal of the learning sciences*, 4(2):167–207, 1995.

Youheng Bai, Mingliang Hou, Teng Guo, Zitao Liu, and Weiqi Luo. A frequency-aware mixture of heterogeneous experts framework for knowledge tracing. In *Proceedings of the ACM Web Conference 2026*, pp. 7080–7090, 2026.

Robert A Bjork, Elizabeth L Bjork, et al. A new theory of disuse and an old theory of stimulus fluctuation. *From learning processes to cognitive processes: Essays in honor of William K. Estes*, 2:35–67, 1992.

Nicholas J Cepeda, Harold Pashler, Edward Vul, John T Wixted, and Doug Rohrer. Distributed practice in verbal recall tasks: A review and quantitative synthesis. *Psychological bulletin*, 132 (3):354, 2006.

Penghe Chen, Yu Lu, Vincent W Zheng, and Yang Pian. Prerequisite-driven deep knowledge tracing. In *2018 IEEE international conference on data mining (ICDM)*, pp. 39–48. IEEE, 2018.

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

Jiwei Deng, Youheng Bai, Mingliang Hou, Teng Guo, Zitao Liu, and Weiqi Luo. Denoised attention and question-augmented representations for knowledge tracing. In *IJCAI*, pp. 9619–9627, 2025.

Hermann Ebbinghaus. [image] memory: A contribution to experimental psychology. *Annals of neurosciences*, 20(4):155, 2013.

Aritra Ghosh, Neil Heffernan, and Andrew S Lan. Context-aware attentive knowledge tracing. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2330–2339, 2020.

Chang Gong, Chuzhe Zhang, Di Yao, Jingping Bi, Wenbin Li, and Yongjun Xu. Causal discovery from temporal data: An overview and new perspectives. *ACM Computing Surveys*, 57(4):1–38, 2024.

Teng Guo, Yu Qin, Yubin Xia, Mingliang Hou, Zitao Liu, Feng Xia, and Weiqi Luo. Enhancing knowledge tracing through decoupling cognitive pattern from error-prone data. In *Proceedings of the ACM on Web Conference 2025*, pp. 5108–5116, 2025.

Mingliang Hou, Zitao Liu, Renqiang Luo, Jiliang Tang, Weiqi Luo, and Feng Xia. Moc-kt: Mixture of convolutions for knowledge tracing. *ACM Transactions on Information Systems*, 2026.

Mohammad Khajah, Robert V. Lindsey, and Michael Mozer. How deep is knowledge tracing? In Tiffany Barnes, Min Chi, and Mingyu Feng (eds.), *Proceedings of the 9th International Conference on Educational Data Mining*, 2016.

Zitao Liu, Qiongqiong Liu, Jiahao Chen, Shuyan Huang, Jiliang Tang, and Weiqi Luo. pyKT: A Python library to benchmark deep learning based knowledge tracing models. In *Advances in Neural Information Processing Systems*, 2022.

Zitao Liu, Qiongqiong Liu, Jiahao Chen, Shuyan Huang, and Weiqi Luo. simplekt: A simple but tough-to-beat baseline for knowledge tracing. In *The Eleventh International Conference on Learning Representations*, 2023.

Koki Nagatani, Qian Zhang, Masahiro Sato, Yan-Ying Chen, Francine Chen, and Tomoko Ohkuma. Augmenting knowledge tracing by considering forgetting behavior. In *The world wide web conference*, pp. 3101–3107, 2019.

Shalini Pandey and George Karypis. A self attentive model for knowledge tracing. In *Proceedings of the 12th International Conference on Educational Data Mining*, 2019.

Shalini Pandey and Jaideep Srivastava. Rkt: relation-aware self-attention for knowledge tracing. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pp. 1205–1214, 2020.

Zachary A. Pardos and Neil T. Heffernan. KT-IDEM: introducing item difficulty to the knowledge tracing model. In Joseph A. Konstan, Ricardo Conejo, José-Luis Marzo, and Nuria Oliver (eds.), *User Modeling, Adaption and Personalization - 19th International Conference*, volume 6787, pp. 243–254, 2011.

Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J Guibas, and Jascha Sohl-Dickstein. Deep knowledge tracing. *Advances in neural information processing systems*, 28, 2015.

Henry L Roediger III and Jeffrey D Karpicke. Test-enhanced learning: Taking memory tests improves long-term retention. *Psychological science*, 17(3):249–255, 2006.

Shuanghong Shen, Zhenya Huang, Qi Liu, Yu Su, Shijin Wang, and Enhong Chen. Assessing student's dynamic knowledge state by exploring the question difficulty effect. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pp. 427–437, 2022.

Mark K Singley and John Robert Anderson. *The transfer of cognitive skill*. Number 9. Harvard University Press, 1989.

Shiwei Tong, Qi Liu, Wei Huang, Zhenya Hunag, Enhong Chen, Chuanren Liu, Haiping Ma, and Shijin Wang. Structure-based knowledge tracing: An influence propagation view. In *2020 IEEE international conference on data mining (ICDM)*, pp. 541–550. IEEE, 2020.

Chun-Kit Yeung and Dit-Yan Yeung. Addressing two problems in deep knowledge tracing via prediction-consistent regularization. In *Proceedings of the fifth annual ACM conference on learning at scale*, pp. 1–10, 2018.

Michael V Yudelson, Kenneth R Koedinger, and Geoffrey J Gordon. Individualized bayesian knowledge tracing models. In *International conference on artificial intelligence in education*, pp. 171–180. Springer, 2013.

Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on World Wide Web*, pp. 765–774, 2017.