Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Do Option-Elimination Logs Improve Student Models? A Preregistered Negative Result on EdNet-KT4

Brahim ES-SABERY

brahim.es-sabery@usms.ma

Université Sultan Moulay Slimane

Abdelkader Moumane

Université Sultan Moulay Slimane

Mohamed Ouhda

Université Sultan Moulay Slimane

Mohamed Baslam

Université Sultan Moulay Slimane

Research Article

Keywords: student modelling, knowledge tracing, adaptive learning, educational recommender systems, interaction logs, preregistration

Posted Date: September 23rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10856004/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Do Option-Elimination Logs Improve Student Models? A Preregistered Negative Result on EdNet-KT4

Brahim Es-Sabery[1*], Abdelkader Moumane[1], Mohamed Ouhada[1], Mohamed Baslam[1]

[1*]Department of Computer Science, Université Sultan Moulay Slimane, Beni Mellal, 23000, Morocco.

*Corresponding author(s). E-mail(s): brahim.essabery@usms.ma;
Contributing authors: amoumane@usms.ma; m.ouhada@usms.ma; m.baslam@usms.ma;

Abstract

Adaptive learning systems depend on student models that transform interaction histories into estimates of learner state. This study tests whether fine-grained option-elimination logs provide useful evidence beyond conventional response history. Under a preregistered protocol, we reconstruct elimination and restoration events from the full EdNet-KT4 dataset, derive 30 features over five historical windows, and add them to an otherwise unchanged 45-feature baseline. We evaluate the incremental representation on two uses of a student model: predicting correctness on a subsequent related question and ranking lectures later consumed by the learner. On 1 984 471 held-out prediction points, the change in area under the receiver operating characteristic curve is **+0.000144**, with a learner-level bootstrap interval spanning zero; lecture ranking is marginally worse. An internal positive control shows that the same pipeline detects the removal of three response-time features, which costs **+0.000379**, so the null result is bounded by an empirically detectable signal. The highest-exposure subgroup shows a corrected positive increment, but the preregistered pooled endpoint does not. We also find that tag-aligned candidate generation makes 33.15 % of observed lecture consumption unreachable before ranking. The results argue against adding option-elimination summaries to the evaluated EdNet student models and demonstrate a transferable positive-control method for evaluating candidate user-model signals.

Keywords: student modelling, knowledge tracing, adaptive learning, educational recommender systems, interaction logs, preregistration

1 Introduction

An adaptive learning system acts on a representation of the learner. The quality of that representation depends not only on the modelling algorithm but also on what evidence is allowed to enter the student model. Most knowledge tracing systems update their estimate from an item identifier, the associated knowledge component, response correctness and, sometimes, response time. This representation is attractive because it is compact and widely available, but it reduces an attempt to its outcome. Any evidence contained in the process by which the learner reached that outcome is lost.

Fine-grained interaction logs make a richer student model possible. They can record revisions, hint requests, navigation, pauses and other actions that may reflect knowledge, uncertainty, strategy or engagement. Recent learner-model research treats behavioural and interaction data as important potential model components Böck et al. (2025), and work on response times shows how a routinely logged process variable can inform knowledge estimation and adaptation Pelánek (2024). The practical question, however, is not whether such a trace is intuitively meaningful. It is whether the trace improves the student model after the information already available to the system has been taken into account.

This paper evaluates one such trace: *option elimination*. On a multiple-choice item, a learner may cross out an answer option and may later restore it. EdNet-KT4 Choi et al. (2020) records these events as `erase_choice` and `undo_erase_choice`, identifies the affected option, and preserves their order. It is therefore possible to distinguish, for example, eliminating a distractor from eliminating the correct option, and a stable elimination from a reconsidered one. Figure 1 illustrates the resulting event-level view of an attempt.

Option elimination is a plausible input to a student model. Eliminating a distractor may indicate partial knowledge; eliminating the correct option may identify a specific error; restoring an option may indicate uncertainty or self-correction. None of these states is represented by final correctness alone. Yet the same events may instead reflect an inexpensive interface habit. If some learners routinely use the affordance and others never do, elimination can primarily identify a stable user type whose performance history is already represented by conventional features. These diagnostic and habitual readings make opposite predictions about incremental value (Figure 2).

We test whether retaining option-elimination logs improves two downstream uses of a computational student model. Task A predicts correctness on a subsequent question sharing a tag with the current question. It evaluates whether the trace improves estimation of the learner's near-term knowledge state. Task B ranks lectures that the learner subsequently consumed. It evaluates whether the same learner representation improves a personalized educational recommendation. The two tasks are deliberately separated: a trace can improve state estimation without improving resource selection, and recommendation can be limited by candidate generation even when the learner

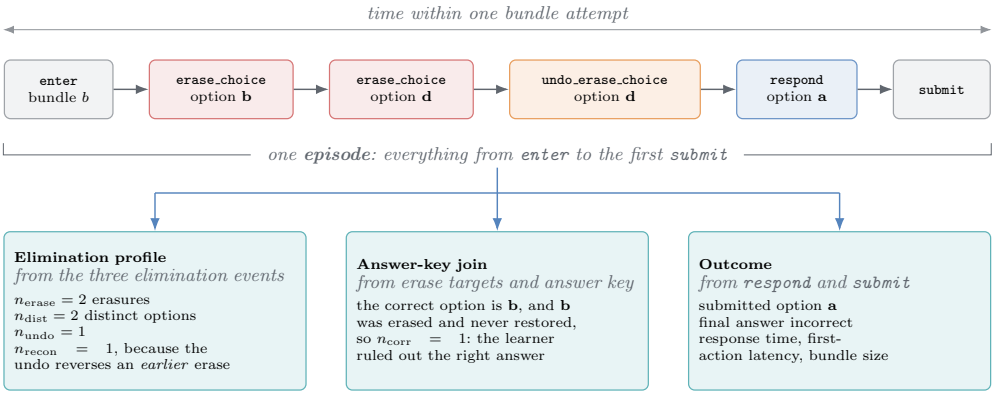


Fig. 1 One attempt in EdNet-KT4, from the raw event stream to the quantities the pipeline derives. Because `erase_choice` and `undo_erase_choice` name the option they affect, the profile can distinguish eliminating a distractor from eliminating the correct answer. A conventional knowledge-tracing representation would keep only the item and the final correctness, discarding the middle events — the question this paper asks is whether those events are worth keeping.

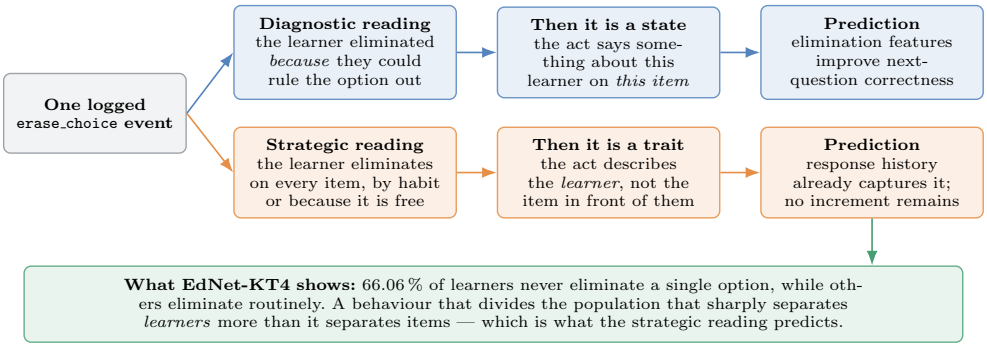


Fig. 2 Two readings of the same logged event, and the opposite predictions they make. The study was designed with both in view. The diagnostic reading is the one that motivates the hypothesis; the strategic reading predicts precisely the null we observe, and the prevalence pattern in the data is more consistent with it. We cannot adjudicate between the two from EdNet alone, and we do not try to. The distinction is included because it explains why a null was a live possibility from the outset rather than an anomaly requiring explanation.

representation is informative. This distinction follows the separation between modelling the learner and using that model for adaptation; educational recommenders commonly use performance data to estimate student state before selecting a learning activity Pelánek et al. (2024).

The study is preregistered and uses the full EdNet-KT4 release. We reconstruct question episodes from raw events, derive 30 elimination features over five historical windows, and add them to an otherwise unchanged 45-feature response-history model. The comparison is paired on the same learners, rows, split and hyperparameter configuration. We find no detectable pooled improvement on either task. On 1 984 471 held-out Task A prediction points, the ROC-AUC increment is +0.000144 and its

learner-level bootstrap interval spans zero. On Task B, the elimination-aware ranker is marginally worse. One preregistered subgroup qualifies the pooled result: the highest-exposure learners show a positive increment that survives correction for the complete subgroup family. A within-cohort permutation removes that increment, indicating that it depends on row-level elimination information rather than merely identifying learners who often use the affordance. We treat this as a specific follow-up hypothesis, not as evidence for the pooled claim.

A negative result about a candidate user-model signal is useful only if the evaluation could have detected a meaningful increment. We therefore include an internal positive control. Three response-side timing features are removed from the same model and evaluated through the same pipeline. Their removal costs +0.000379 ROC-AUC, more than twice the contribution of the complete elimination block and just above the study's estimated detection resolution. The control turns an otherwise ambiguous null into a bounded statement: for the evaluated representation and model families, any pooled elimination benefit is smaller than a modest, known component of the existing student model (Figure 3).

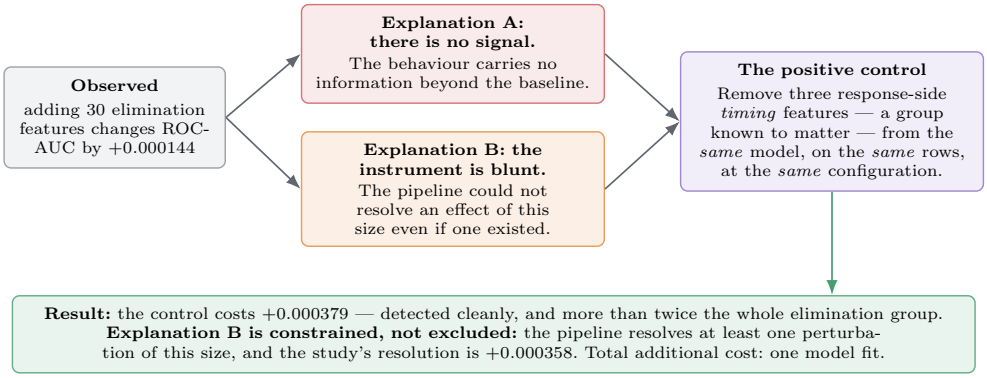


Fig. 3 The logic of the internal positive control, which is the load-bearing element of this study's design. A null result always admits two explanations, and only one of them is about the world. The control discriminates between them by pushing a feature group of known importance through the identical pipeline. If that pipeline registers the removal of three timing features but not the addition of thirty elimination features, then the pipeline works and the elimination features do not. We report this control beside every headline number in the paper rather than in a robustness appendix, because without it the central claim would be uninterpretable.

The paper addresses four research questions:

1. **RQ1: Student-model accuracy.** After controlling for item context, correctness, timing and response history, do option-elimination features improve next-question correctness prediction on EdNet-KT4?
2. **RQ2: Personalized recommendation.** Do the same features improve retrospective ranking of lectures that a learner subsequently consumed?
3. **RQ3: Sensitivity of the evaluation.** If the pooled increment is not detected, can the same pipeline detect the removal of a small feature group known to contain predictive information?

4. **RQ4: Sequential representation.** Is a sequential architecture that reads event order stable enough, at the available seed budget, to determine whether sequence information changes the conclusion?

The contribution is consequently not a new predictive architecture. It is an evaluation of what evidence should enter a student model and how a negative answer can be made informative. First, it provides a preregistered estimate and empirical upper bound for the incremental value of option-elimination logs in two adaptive-learning tasks. Second, it contributes a transferable positive-control design for evaluating candidate user-model signals: ablate a small established signal under the identical comparison and report both increments. Third, it separates representation quality from adaptation constraints by measuring the recall ceiling imposed by Task B's tag-aligned candidate generator: 33.15 % of observed lecture consumption is unreachable before any personalized ranker is applied. Finally, it reports the seed instability of the sequential comparison instead of using a single draw to support a conclusion.

The remainder of the paper positions this evaluation within student modelling, process-trace modelling and educational recommendation (Section 2); defines the behaviour and the logic of the positive control (Section 3); describes the dataset, models and protocol (Sections 4–7); and reports the empirical results (Section 8). Sections 9–12 interpret the findings for user modelling and adaptive systems, state the validity boundaries and identify the follow-up design implied by the high-exposure subgroup. Section 13 concludes.

2 Student Models, Process Traces and Adaptive Recommendation

Student modelling research must decide both how learner state is represented and which observations update that representation. Recent reviews distinguish raw learner data from characteristics inferred from it and show that behavioural and interaction data are common, but far from universal, components of learner models Böck et al. (2025). This study focuses on that acquisition boundary. It applies a corrective-evaluation lens to a proposed *input signal* rather than introducing another architecture: a behaviour that the platform logs and most student models discard.

The work connects five literatures. Knowledge tracing supplies the computational student model; educational recommendation supplies the second use of that model; assessment research motivates option elimination as a possible knowledge indicator; interaction-log research motivates retaining process evidence; and research on negative results and stochastic evaluation supplies the design that makes a small or null increment interpretable. We review each in turn and then state the unresolved user-modelling question.

2.1 Student modelling through knowledge tracing

Knowledge tracing estimates a changing mastery state for a learner from a sequence of graded responses and is therefore a central family of computational student models Pelánek (2017). Corbett and Anderson (1994) introduced Bayesian knowledge

tracing, which models mastery per skill as a hidden binary variable with four interpretable parameters — prior knowledge, learning rate, slip and guess. Its transparency is the reason it survives in deployed tutoring systems long after more accurate models appeared.

Piech et al. (2015) replaced that formulation with deep knowledge tracing, a recurrent network over the response sequence that learns skill structure implicitly rather than requiring it. Subsequent variants added memory and attention mechanisms — dynamic key–value memory Zhang et al. (2017), self-attention Pandey and Karypis (2019), monotonic attention with forgetting decay and Rasch item embeddings Ghosh et al. (2020), and a transformer with separated encoder–decoder streams evaluated on EdNet itself Choi et al. (2020). All achieve strong benchmark results, and all take the same input representation: an ordered sequence of (item, correctness) pairs, sometimes augmented with response time or skill tags.

The field has also produced a valuable corrective literature, and our study sits in that tradition. Xiong et al. (2016) replicated deep knowledge tracing and found that a portion of the original gain came from duplicated records in the benchmark rather than from the model. Wilson et al. (2016) showed that well-specified item-response models match or beat neural networks for proficiency estimation. Gervet et al. (2020) compared model families across several datasets and identified the conditions under which deep models actually help — essentially, large datasets with rich sequential structure — and the conditions under which they do not. The common lesson is that reported gains in this field require careful control of what is being compared, which is the principle our matched-configuration design applies.

What these approaches share is an outcome-centred representation: an ordered sequence of (item, correctness) pairs, sometimes augmented with response time or skill tags. Response time is a useful comparison because it is also a process measure, yet it has established roles in estimating knowledge, identifying behavioural states and driving adaptation Pelánek (2024). Actions taken *within* an item are less often represented because most public datasets do not record them. The closest prior work is therefore the EdNet-based models themselves Choi et al. (2020); Gervet et al. (2020). None of the EdNet evaluations we surveyed uses the KT4 erase and undo events. The nearest neighbours of this study differ from it precisely at the model-input boundary being tested here.

2.2 Educational recommender systems

Educational recommenders suggest courses, exercises, videos or lectures. Manouselis et al. (2011) survey the area across resource types and platforms and note the recurring difficulties: sparse interaction data, pedagogical constraints a general-purpose recommender does not represent, and evaluation protocols that vary enough between studies to make results hard to compare. Tag metadata is attractive in this setting because it is available, interpretable and stable, and because it grounds a candidate set without requiring text processing. Its weakness, which our Task B measures directly, is that a tag-aligned candidate rule silently discards everything the tags do not connect.

Within an adaptive learning system, recommendation is downstream from student modelling: performance observations are converted into an estimate of learner state,

and that state is combined with a domain or activity model to select what should be offered next. Pelánek et al. Pelánek et al. (2024) make this separation explicit in a practical framework for personalized learning-activity recommendation. It matters here because a weak Task B result can arise from either side. Elimination may fail to improve the student representation, or an improved representation may be unable to help when the candidate generator excludes the relevant resource. We therefore evaluate student-model features and candidate reachability separately.

Two methodological findings from the wider recommender literature bear directly on our design. Dacrema et al. (2019) reproduced eighteen neural recommendation papers and found that most were beaten by properly tuned non-neural baselines, concluding that reported progress in the area was substantially overstated. Rendle et al. (2020) revisited a specific widely cited comparison and showed that a correctly configured matrix factorisation outperformed the neural model that had displaced it. Neither paper argues that neural methods cannot work; both argue that a comparison is only as good as the baseline it beats. This is why our response-only baseline receives exactly the same search grid, the same rows and the same early stopping as the model under test.

Cremonesi et al. (2010) established that popularity is a strong baseline for top-N recommendation whenever consumption is concentrated, which is the regime educational catalogues typically occupy. Abdollahpouri et al. Abdollahpouri et al. (2019) analysed the resulting popularity bias and its distributional consequences. Both findings are visible in our Task B: popularity ranks lectures well, and the tag-overlap score that grounds candidate generation turns out to be the *worst* of the four rankers we evaluate. We therefore report popularity as a first-class comparator rather than as a formality.

2.3 Elimination as test-taking behaviour

Narrowing the option set before committing raises the probability of a correct guess, and test-preparation instruction commonly teaches learners to do it. The platform underlying EdNet provides an explicit interface affordance for elimination, which is why the events exist in the log at all.

Haladyna et al. (2002) review multiple-choice item-writing guidelines and treat distractors and the way learners process the option set as central to item quality. This is one reason to expect that how a learner treats the options might be informative: if distractors are functional, then which ones a learner rules out should reflect what they know. We take this as motivation for the hypothesis rather than as established psychometric theory, and we make no claim that EdNet erase and undo events are validated indicators of any construct. Adjudicating that would require instruments this dataset does not contain.

The consequences of this literature for our hypothesis are the two competing readings set out in Section 1 and drawn in Figure 2. Under the diagnostic reading the act signals partial knowledge about the item; under the strategic reading it signals a stable disposition of the learner, which a response-history model already captures. The literature does not settle which applies to a low-cost interface affordance on a commercial test-preparation platform, and neither can we.

2.4 Interaction traces as student-model evidence

The broader motivation for this study comes from work showing that intermediate behavioural traces can carry information that final outcomes do not. Kizilcec et al. Kizilcec et al. (2013) identified distinct engagement trajectories in MOOC clickstreams — completing, auditing, disengaging, sampling — that a completion indicator cannot express, and Anderson et al. (2014) related engagement styles to outcomes at scale Siemens (2013); Baker and Inventado (2014). This literature motivates the present hypothesis; it does not establish it. Whether a *particular* fine-grained action carries usable signal is an empirical question that has to be answered per action, target and dataset: the answer for video-watching patterns in a MOOC does not transfer to option elimination in a test-preparation app, which is precisely why the question is worth testing rather than assuming, and why a null is worth reporting when it arrives.

This caution also appears in work on sensor-free state detection. Interaction features can support models of affect or cognitive state, but detectors often depend on the learning environment and may transfer poorly when the available traces change Rahman et al. (2024). Our study asks a narrower question: does one trace improve two concrete downstream functions after conventional student-history variables have already been supplied?

We interpret elimination behaviourally throughout this paper. `erase_choice` records an interface event; whether it reflects uncertainty, deliberate test-taking strategy, exploration or incidental interaction cannot be determined from the fields EdNet provides, and we make no misconception claims anywhere in the manuscript.

2.5 Negative results, preregistration and seed variance

Publication practice under-reports null findings, which biases the literature toward effects that later fail to replicate Ioannidis (2005). Preregistration — fixing hypotheses and analysis choices before seeing outcomes — addresses the flexibility that makes weak effects appear strong Nosek et al. (2018), and Simmons et al. (2011) show experimentally how ordinary analytic freedom manufactures significance from noise. Pineau et al. (2021) report the reproducibility programme run at a major machine-learning conference and the checklist that came out of it, the closest analogue in our own field.

Gelman and Loken (2013) make the point that matters most for a study like ours. The relevant failure mode is not fraud; it is analytic freedom exercised in good faith. With enough defensible choices about windows, subgroups, thresholds and metrics, a null can be converted into a finding without any single step feeling dishonest. Our response is procedural: we fixed hypotheses, cohort thresholds, primary metrics and the hyperparameter grid before training anything, logged every subsequent amendment with the empirical observation that motivated it, and label every non-preregistered analysis as exploratory at the point where it appears.

A closely related strand concerns reporting practice for stochastic training: single-run comparisons of sequence-tagging models routinely support conclusions that score distributions overturn Reimers and Gurevych (2017), seed variance in deep reinforcement learning can exceed the difference between published methods Henderson et al. (2018), and carefully tuned baselines match architectures reported to improve on them

Melis et al. (2018). Bouthillier et al. (2021) formalise the accounting of how many runs are needed before a difference is meaningful, and Dodge et al. (2019) argue for reporting the computational budget alongside the result. Section 8.8 reports exactly this failure mode in our own sequential experiment and converts it into a seed budget.

2.6 Research gap and positioning

EdNet has been used predominantly as a knowledge-tracing benchmark, with models consuming question–response sequences Choi et al. (2020); Gervet et al. (2020). Its KT4 interaction level, which alone records the within-item actions, is used far less often than the aggregated levels. We are not aware of a preregistered evaluation of whether erase and undo events add incremental predictive value after response-history controls, or whether the same increment survives when the student model is used for personalized resource ranking. We state that as the limit of our literature search rather than as a priority claim.

Table 1 positions this work against the five literatures.

The gap this paper addresses can therefore be stated in one sentence:

> whether answer-option elimination provides reproducible incremental value to a student model after controlling for correctness, timing, tags, sequential history and historical resource consumption — and whether any such value improves a downstream personalized recommendation.

The positive control and candidate-recall analysis distinguish this evaluation from a report of a failed feature experiment: the former calibrates sensitivity to a small user-model signal, while the latter separates the quality of the student representation from a structural limit of the adaptive decision.

3 Key Notions

Four ideas carry the rest of the paper. The first two define the evidence and state representation of the student model. The other two define how we evaluate that representation and how we separate it from a downstream recommendation constraint.

3.1 Option elimination as a logged behaviour

An *option* is one of the answer choices attached to a multiple-choice item. *Option elimination* is the act of crossing one out, and in EdNet-KT4 it produces an `erase_choice` event carrying both the item identifier and the option affected. Restoring an option produces an `undo_erase_choice` event carrying the same two fields.

Three properties of this logging scheme determine what can and cannot be measured from it, and each has a direct consequence for the study.

The option is named. Because the event identifies which option was crossed out, and the content metadata gives the correct option for each question, eliminating a distractor and eliminating the correct answer become distinguishable — which is what makes the diagnostic reading testable at all. Without this field the hypothesis could only be tested in its weakest form.

Table 1 Where this study sits. Each row is a body of work the paper builds on; the right-hand column states what it leaves open that the present study addresses.

Literature	What it establishes	What it leaves open
Student and knowledge modelling Corbett and Anderson (1994); Pelánek (2017); Piech et al. (2015); Zhang et al. (2017); Pandey and Karypis (2019); Ghosh et al. (2020); Choi et al. (2020)	Latent mastery can be estimated from graded response sequences, with successive architectures improving accuracy.	All operate on (item, correctness) pairs. Whether within-item behaviour would add anything is untested.
Corrective replication Xiong et al. (2016); Wilson et al. (2016); Gervet et al. (2020)	Reported gains often shrink or vanish under matched baselines and clean data handling.	The corrective lens has not been applied to a proposed *new input signal*, only to new architectures.
Educational recommenders Manouselis et al. (2011); Pelánek et al. (2024); Cremonesi et al. (2010); Abdollahpouri et al. (2019); Dacrema et al. (2019); Rendle et al. (2020)	Simple baselines are strong; popularity dominates where consumption is concentrated; candidate generation drives offline results.	The share of real consumption a metadata candidate rule makes unreachable is not routinely quantified.
Assessment research Haladyna et al. (2002)	Distractors and option processing are central to item quality.	Whether a logged, low-cost elimination affordance indicates anything about learner state is an empirical question.
Interaction-log modelling Kizilcec et al. (2013); Anderson et al. (2014); Siemens (2013); Baker and Inventado (2014); Rahman et al. (2024)	Intermediate traces reveal structure that outcomes hide.	Transfers per action and per dataset; establishes plausibility, not presence.
Negative results and seed variance Ioannidis (2005); Nosek et al. (2018); Simmons et al. (2011); Gelman and Loken (2013); Pineau et al. (2021); Reimers and Gurevych (2017); Henderson et al. (2018); Melis et al. (2018); Bouthillier et al. (2021); Dodge et al. (2019)	Nulls are under-reported; analytic freedom manufactures effects; single-seed neural comparisons are unreliable.	Few applied education studies carry an internal control that makes a null interpretable.

The act is cheap. Crossing out an option on this platform costs one tap and carries no penalty, so learners may perform it without deliberation, which is what makes the strategic reading plausible. On a platform where elimination required more effort the same behaviour might be considerably more selective; we return to this in Section 10.

The act is not annotated. The log records that an option was crossed out, not why. No field distinguishes ruling out a known-wrong option from tidying the screen or tapping by accident. Every claim in this paper is therefore about the logged event, not about an inferred mental state, and we use the word "behaviour" rather than "uncertainty" or "misconception" throughout.

3.2 Student-model state and the prediction point

A *student-model state* summarises the information represented about a learner at a *prediction point* — in this study, always the moment a completed episode ends. Its defining property is that it is non-anticipatory: it may aggregate the episode that has just finished and every earlier episode by the same learner, and nothing afterwards. This is the single easiest thing to get wrong in a pipeline of this kind, because rolling-window operations over a sorted table will happily read forward across a learner boundary if the grouping is specified loosely; Section 7.7 reports the brute-force check we run against that specific failure.

We compute learner state over five windows: the last one, three, five and ten completed episodes, and all activity in the previous thirty minutes. The five-episode window is the preregistered primary; the others are sensitivity analyses, and reporting all five is deliberate, because a single window choice is exactly the kind of researcher degree of freedom that Gelman and Loken (2013) describe.

3.3 What makes a null result informative: the positive control

This notion is borrowed from experimental biology, where it is routine, and it is uncommon enough in evaluations of user-model inputs to be worth stating carefully.

A *positive control* is a manipulation whose effect is known in advance, carried through the identical measurement procedure as the manipulation under test. Its purpose is not to test a hypothesis. Its purpose is to test the apparatus. If the control produces nothing, the experiment has failed and no conclusion about the world can be drawn from it.

The converse needs stating carefully, because it is easy to overclaim. A control that registers does not prove that *any* effect of similar size would have registered: the control and the manipulation under test draw on different parts of the feature space, with different sparsity and different correlation structure, and detectability is not a property of effect size alone. What a successful control establishes is narrower and still useful — that the pipeline resolves at least one perturbation of this magnitude, end to end, rather than being blunt everywhere. We therefore use it as an internal sensitivity benchmark, and pair it with an explicit minimum detectable effect (Section 7.4), which is the quantity that does generalise.

In this study the control is a group of three response-side timing features: the time since the learner's last activity, the time to their first action within the episode, and the time to their first response. These features are present in *both* models — they belong to the shared response side, not to the elimination group — and the control consists of removing them from the elimination-aware model at its selected configuration, on the same rows, and measuring the drop. Because they sit on the shared side, this

is a control on the common portion of the feature space rather than an elimination ablation, and we label it as such everywhere it appears.

Three design requirements make such a control informative, and all three are easy to violate:

- **Same rows.** The control and the test must be evaluated on the identical prediction points. Evaluating on different subsets introduces a composition difference that neither quantity can be separated from.
- **Same configuration.** The control must be run at the hyperparameters selected for the model under test, not re-tuned. Re-tuning turns the measurement into a mixture of a feature effect and a capacity effect.
- **Comparable magnitude.** A control whose effect is enormous proves only that the pipeline can detect enormous effects. Ours costs +0.000379 ROC-AUC, which is the same order of magnitude as the effect under test — larger, but by a factor of about two and a half rather than by a factor of a hundred. That is what makes the comparison informative.

3.4 Candidate generation and the recall ceiling

A recommender does not score every item in the catalogue. It first generates a *candidate set* by some cheap rule, then ranks within it. In a metadata-grounded system the rule is typically a metadata match: for lecture recommendation on EdNet, a candidate lecture must share at least one tag with the question bundle the learner has just attempted.

The consequence is easy to state and easy to overlook. If an item shares no tag with any question, it can never enter any candidate set, and if a learner consumes it, that consumption cannot be retrieved by any model operating under the rule — regardless of how good the ranker is. The *recall ceiling* is the share of real consumption that this makes unreachable, and it is a property of the dataset and the rule, not of a model.

Three things follow, and they are the reason we treat this as a first-class result rather than a caveat:

1. **Absolute recall becomes uninterpretable without it.** A reported Recall@3 of 0.6 means one thing if the ceiling is 1.0 and something quite different if the ceiling is 0.67.
2. **Model comparisons are unaffected.** All rankers operate over identical candidate sets, so the ceiling cancels in any difference between them. Our Task B comparison is therefore valid despite the ceiling.
3. **Deployment is affected.** A system that cannot surface a third of what learners actually watch is not merely scoring poorly on a benchmark; it is structurally unable to serve those requests.

We measure this quantity for EdNet in Section 4.4 and recommend in Section 11 that comparable studies report the analogue for their own candidate rule.

4 Dataset and Data Audit

Every claim in this paper rests on properties of EdNet-KT4 that we verified before modelling rather than assumed. This section reports that audit. It is placed before the problem definition because two of its findings — the absence of a reliable question-level delimiter and the tag-reachability ceiling — determined how the problem had to be defined.

4.1 EdNet-KT4

EdNet-KT4 is the most granular level of the EdNet release Choi et al. (2020), recording every user-interface action rather than only graded responses. The archive contains 131 441 538 events from 297 915 learner logs, distributed across the action types shown in Table 2.

Table 2 Action-type composition of EdNet-KT4 after removing commerce and malformed events. Elimination actions are the two rows in bold. Shares are of all events and may not sum to 100 due to rounding and to action types omitted from this table.

Action type	Events	Share
`enter`	32 943 087	25.06 %
`respond`	23 384 480	17.79 %
`submit`	16 488 061	12.54 %
`quit`	16 455 026	12.52 %
`erase_choice`	**4 714 534**	**3.59 %**
`undo_erase_choice`	**142 092**	**0.11 %**
`play_video`	1 928 356	1.47 %
`play_audio`	16 580 464	12.61 %
All events	131 441 538	100.00 %

Two action types carry elimination. `erase_choice` is emitted when a learner crosses out an option, and `undo_erase_choice` when they restore one. On both, the item identifier names the question and the answer field names the option affected, which is what makes it possible to distinguish eliminating a distractor from eliminating the correct answer.

We never extract the archive. KT4 is distributed as 297 915 individual CSV files, and unpacking them costs substantially more than reading members directly from the compressed archive into a columnar store. The raw data therefore stay immutable throughout the study, which also removes a class of accidental modification from the reproducibility argument.

4.2 Timestamp integrity

The reconsideration construct depends on establishing that an undo followed its own erase, so we verified three timestamp properties before use. Timestamps are 64-bit epoch milliseconds at true millisecond resolution (exactly 10.00 % of values are divisible by 10 ms, which is what chance produces, so there is no coarser underlying clock). No non-monotonic consecutive pair occurs within any learner's stream. Within-learner ties occur in 2.90 % of consecutive event pairs, and because no pair is out of order, the original file order serves as a deterministic tie-breaker; ordering is therefore reproducible rather than dependent on a sort algorithm's stability.

4.3 Content metadata

After validation, 13 169 questions in 9 534 bundles and 1 021 lectures remain, with no duplicate identifiers. 797 questions (6.05 % of them) and 437 lectures (42.80 %) carry no usable tag; the asymmetry matters, because a lecture without a tag can never enter a tag-aligned candidate set, and it is one of the sources of the ceiling quantified in Section 4.4. Lecture tag frequency has a median of 2 and a maximum of 8, and no tag reaches the 20 % near-universal threshold at which a tag-overlap score would degenerate into a constant.

4.4 The question–lecture tag graph and its ceiling

The bipartite question–lecture tag graph has 52 821 edges at a density of 0.0039. Critically, 583 of 1021 lectures share no tag with *any* question in the dataset. Under a tag-aligned candidate rule these lectures can never enter a candidate set, and therefore 33.15 % of all observed lecture-consumption events are unreachable by any model, regardless of ranking quality. Figure 4 shows the split.

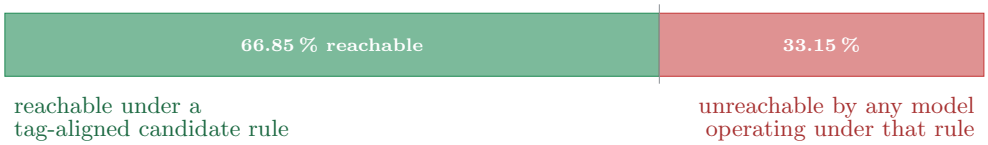


Fig. 4 The empirical recall ceiling of tag-grounded candidate generation. 583 of the 1021 EdNet lectures share no tag with any question in the dataset, so no candidate set built by tag alignment can ever contain them. 33.15 % of observed lecture consumption is therefore unreachable by any model operating under that rule, whatever its ranking quality. The figure is a property of the dataset and the candidate rule, not of a model, and it bounds how far conclusions from offline tag-grounded recommendation transfer to deployment.

We report this as a property of the dataset and the candidate rule, not of any model, and we are careful about what it does and does not affect. It does not affect the model comparisons in Section 8, because all rankers operate over identical candidate sets and the unreachable share cancels in any difference between them. It does bound how far conclusions about tag-grounded recommendation transfer to deployment, where the third of consumption that is tag-unreachable still happens and still has to be served.

5 Student-Model Evaluation Tasks

This section defines the objects the rest of the paper manipulates: the episode, the elimination profile, the learner state, and the two prediction tasks. Each design decision is stated explicitly, because the audit above showed that protocol-level choices in this dataset have larger consequences than the differences between competing models. Figure 5 gives the shape of the whole pipeline before we describe its parts.

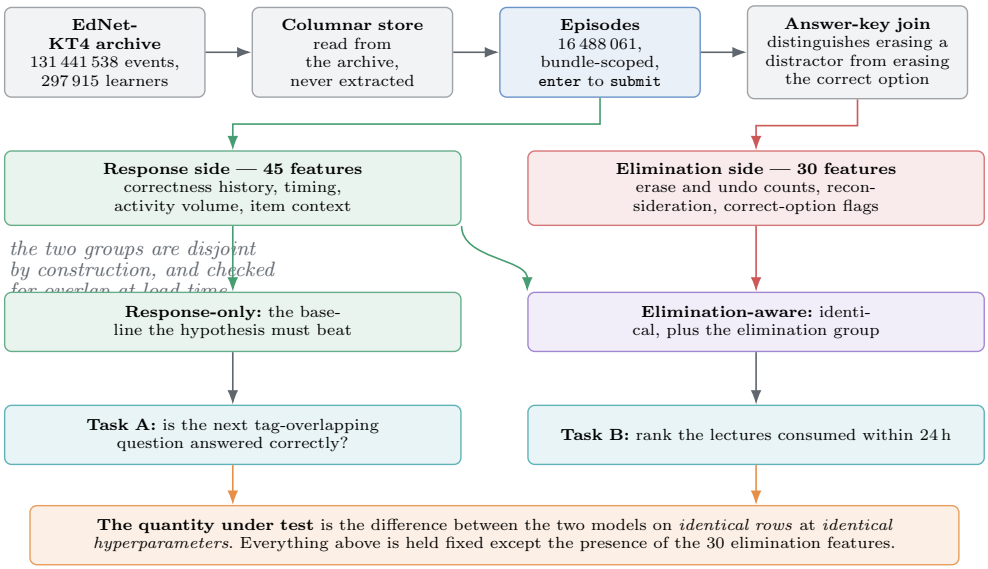


Fig. 5 The pipeline, from the raw archive to the two evaluation tasks. The design constraint that governs everything downstream is the one drawn at the bottom: the response-only and elimination-aware models must differ by exactly one thing. They see the same episodes, the same splits, the same search grid and the same rows at evaluation time, and the response and elimination feature groups are disjoint by construction rather than by convention. Any difference between them is therefore attributable to the elimination features and to nothing else.

5.1 Notation

Let $u \in \mathcal{U}$ index learners, and let learner u's event stream be $\mathcal{E}_u = (e_1, e_2, \ldots)$, ordered by timestamp with ties broken by file order. Each event e carries

- an action type $a(e)$, such as `enter`, `erase_choice` or `submit`;
- an item identifier $i(e)$, naming a question, bundle or lecture;
- a timestamp $t(e)$, in epoch milliseconds;
- and, for response and elimination actions, an option $o(e) \in \{\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{d}\}$.

We write $\kappa(q)$ for the correct option of question q, and $\tau(\cdot)$ for the tag set of a question, bundle or lecture. Table 3 collects the notation used from here on.

Table 3 Notation used throughout the paper.

Symbol	Meaning
$u \in \mathcal{U}$	learner index
ε	a reconstructed episode
$\kappa(q)$	correct option of question q
$\tau(\cdot)$	tag set of a question, bundle or lecture
$\mathrm{w}N_x$	window feature: mean of quantity x over the last N episodes

5.2 Episode reconstruction

Definition of episode. An *episode* ε is a maximal contiguous run of events that opens at $a(e) = \texttt{enter}$ with a bundle identifier and closes at the first subsequent $a(e) = \texttt{submit}$. Events occurring after that submit are discarded.

The unit is the bundle rather than the question, which was the study's most consequential amendment from the original protocol. The KT4 stream provides no reliable question-level delimiter — `enter` and `submit` are emitted at bundle granularity only — and inferring question boundaries would itself depend on the elimination events, contaminating the very construct under test. We therefore segment at bundle scope and report the structural single-question stratum as a sensitivity analysis instead. Algorithm 1 states the procedure together with the invariants asserted at construction time.

Algorithm 1 Episode reconstruction and elimination profile

Require: ordered events $\mathcal{E}_u$ for every learner, answer key κ
Ensure: episodes, each carrying an elimination profile

drop commerce and malformed events
$s \leftarrow \textsc{CumSum}(\mathbb{1}[a(e)=\texttt{enter},\ \text{bundle}])$ ▷ segment identifier
restrict each segment to events of its own learner
for all segments s **do**
  assert exactly one `submit` in s
  truncate s at that `submit`
  $E \leftarrow \{e \in s : a(e) = \texttt{erase_choice}\}$
  $U \leftarrow \{e \in s : a(e) = \texttt{undo_erase_choice}\}$
  $n_{\text{erase}} \leftarrow |E|; \quad n_{\text{undo}} \leftarrow |U|$
  $n_{\text{corr}} \leftarrow |\{e \in E : o(e) = \kappa(i(e))\}|$
  $n_{\text{dist}} \leftarrow |\{(i(e), o(e)) : e \in E\}|$
  ▷ an undo counts only if it reverses an *earlier* erase
  $R \leftarrow \{v \in U : \exists\, e \in E,\ (i(e), o(e)) = (i(v), o(v)),\ \mathrm{seq}(e) < \mathrm{seq}(v)\}$
  $n_{\text{recon}} \leftarrow |R|$
  assert $n_{\text{recon}} \le n_{\text{undo}},\ n_{\text{dist}} \le n_{\text{erase}},\ n_{\text{corr}} \le n_{\text{erase}}$
end for

Reconstruction yields 16 488 061 episodes across 297 443 learners, with a mean of 1.287 questions per bundle; 88.27 % of episodes are structurally single-question.

5.3 The elimination profile

Definition of the profile. Each episode carries nine elimination quantities, computed as Algorithm 1 specifies. Erase events are joined to the answer key so that eliminating the correct option is distinguishable from eliminating a distractor. Repeated erasures of the same option are deduplicated on the (question, option) pair, so a learner who taps the same option twice does not register as having eliminated two things.

Definition of reconsideration. An undo counts as a reconsideration only when the same (question, option) pair was erased *earlier in event order within the same episode*. The construct is therefore "the learner changed their mind" rather than "the learner pressed undo". This is a strictly narrower definition than counting undo events, and we adopted it because the wider version would have conflated genuine reversal with interface noise.

Four invariants are asserted at construction time and hold on every episode: reconsiderations never exceed undos, distinct erasures never exceed total erasures, correct-option erasures never exceed total erasures, and questions with elimination never exceed bundle size. A violation would indicate a join error rather than a surprising learner, which is why they are assertions rather than logged warnings.

Table 4 and Figure 6 report how common each behaviour is.

Table 4 Elimination prevalence over 16 488 061 reconstructed episodes. The behaviour is common enough to model and far from universal, so the design is not limited by prevalence.

Behaviour	Episodes	Share
Any elimination (`erase_choice`)	2 388 160	14.48 %
Correct option eliminated	531 663	3.22 %
Any undo	103 652	0.63 %
Erase→undo reconsideration	103 440	0.63 %

The prevalence figures matter for the design: elimination is neither rare enough to be dismissed as noise nor universal enough to be uninformative, and 66.06 % of learners never eliminate at all. That share forced the cohort amendment recorded in Section 7 and provides the study's cleanest internal control, as Section 8.7 explains.

5.4 Student-model representation

The prediction point is the instant a completed episode ends. Let $\varepsilon_{u,1}, \ldots, \varepsilon_{u,n_u}$ be learner u's episodes in time order. The state at prediction point j aggregates $\{\varepsilon_{u,k}\}$ for $k \leq j$ only, so nothing occurring after the prediction point can enter it.

For an episode-count window of size N and a per-episode quantity $x(\cdot)$, the window feature is the mean over that window,

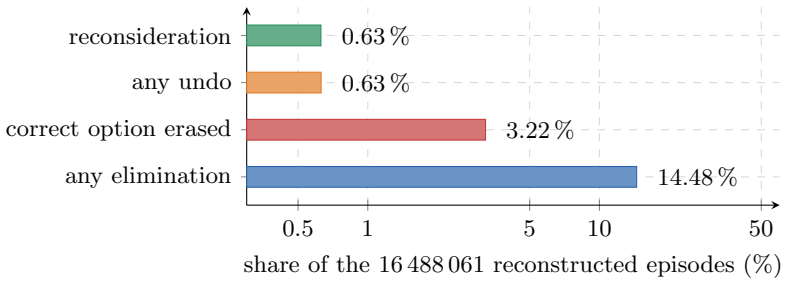


Fig. 6 How common the behaviour is. Elimination appears in 14.48 % of episodes, which is the property that makes the hypothesis worth testing: frequent enough that a real effect would be measurable, far from universal, so the feature is not a constant. The correct option is erased in 22.26 % of eliminating episodes. Undo and reconsideration are an order of magnitude rarer, which is the first hint that the reconsideration features will have little to contribute. The axis is logarithmic.

$$\mathrm{w}N_x = \frac{1}{|W|} \sum_{k \in W} x(\varepsilon_{u,k}), \tag{1}$$
$$W = \{\ell, \ldots, j\}, \qquad \ell = \max(1,\, j - N + 1).$$

Where $N \in \{1, 3, 5, 10\}$ is the window length in completed episodes, j is the index of the episode that has just finished (always included, no future episode ever is), $\ell = \max(1, j - N + 1)$ clamps the left edge at the learner's first episode so windows never run backwards across a learner boundary, and $x(\cdot)$ is any per-episode quantity — an elimination count, a correctness indicator, a duration.

For the thirty-minute window, W is instead defined by time rather than by count,

$$W = \{k \le j : t_{\mathrm{start}}(\varepsilon_{u,k}) \ge t_{\mathrm{end}}(\varepsilon_{u,j}) - \Delta\}, \qquad \Delta = 30 \text{ min}, \tag{2}$$

clamped to contain at least the current episode, so that a learner returning after a long absence still has a defined state. $N = 5$ is the preregistered primary and the others are sensitivity analyses.

The elimination group instantiates (1) with six quantities per window — the indicator of any elimination, the indicator of correct-option elimination, and the counts of erasures, undos and reconsiderations, plus the unnormalised count of episodes containing elimination — giving $6 \times 5 = 30$ features.

Rolling sums are computed with a prefix-sum and a per-learner clamp,

$$\sum_{k \in W} x_k = P_{j+1} - P_\ell, \qquad P_m = \sum_{k < m} x_k, \tag{3}$$

rather than with a grouped rolling operation. This is roughly two orders of magnitude faster on 16 488 061 rows, and because a hand-rolled optimisation of this kind is exactly where a subtle leakage bug would hide, it is verified equal to the naive grouped implementation by a built-in self-test on sampled rows (Section 7.7).

Table 5 inventories the resulting features. The response and elimination groups are disjoint by construction: no response feature may derive from an erase or undo event, and this is enforced by the configuration file that defines the groups and checked for overlap at load time.

Table 5 Feature inventory. The response and elimination groups are disjoint by construction: no response feature derives from an erase or undo event. All window features aggregate the current and earlier episodes only. Online Resource 1 lists every column.

Group	Count
Response state, 5 windows	35
Point-in-time context	9
Target-item attributes	3
response-only total	*45*
Elimination, 6 quantities × 5 windows	30
elimination-aware total	*75*

5.5 Task A: near-term knowledge-state prediction

Definition. Given the learner state at a prediction point, predict whether the learner will answer correctly at the next episode sharing at least one tag with the current bundle.

Of 16 488 061 episodes, 74.82 % have a scorable tag-overlapping successor, giving 12 336 027 prediction points overall and 1 984 471 in the test split. The tag-overlap requirement is what makes the target a *related* question rather than merely the next one, which is the sense in which the task measures difficulty modelling rather than general engagement.

One property of this target deserves emphasis because it is easy to misinterpret the results without it: the interval between a prediction point and its target is extremely dispersed (median 191 s, 95th percentile 358 h; only 43.1 % of targets are the immediately following episode), as Figure 7 shows.

The preregistered target therefore mixes short-range and very long-range prediction in one number, which is why we report the same-session stratum alongside the primary result in Section 8.10 rather than relegating it to Online Resource 1.

5.6 Task B: personalized lecture ranking

Definition. Given the same learner state, rank candidate lectures by whether the learner consumed them within 24 hours. Candidates are lectures that share a tag with the question bundle and that the learner has not previously consumed.

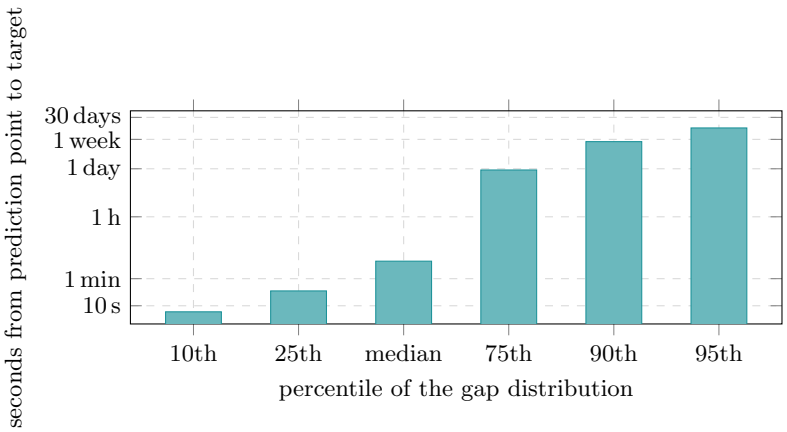


Fig. 7 The preregistered Task A target is not a short-horizon target. The median gap between a prediction point and the next tag-overlapping episode is 191 s, but the 75th percentile is 22.2 h and the 95th is 358 h; only 43.1 % of targets are the immediately following episode. The axis is logarithmic because the distribution spans six orders of magnitude. This is why the same-session stratum is reported beside the primary result rather than buried in Online Resource 1.

Of 5 395 129 episodes followed by any lecture view in the window, 538 474 qualify as prediction points; Figure 8 shows where the data go. Three exclusions account for the remainder exactly, and the generator that produces these numbers asserts the residual is zero rather than printing whatever subset it finds: 4 256 120 (78.9 %) are dropped because the lecture the learner actually consumed shares no tag with the bundle — the ceiling of Section 4.4 reappearing as attrition rather than as a metric; 600 436 because fewer than two candidates remain, so no ranking decision exists to evaluate; and 99 because the bundle has no candidate lecture at all.

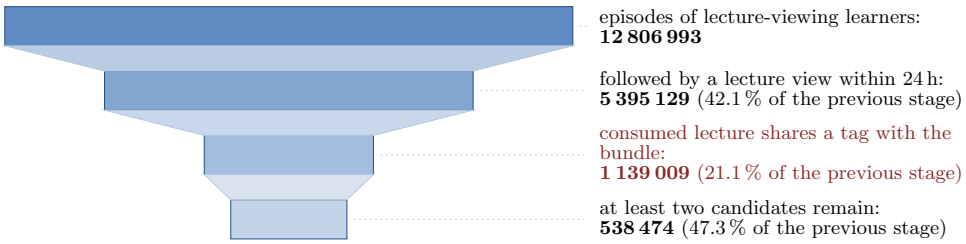


Fig. 8 Where Task B loses its data, drawn as a funnel (widths are proportional to the square root of the counts). The decisive step is the third: of the 5 395 129 episodes followed by a lecture view, 4 256 120 are discarded because the lecture the learner actually watched shares no tag with the bundle they had just attempted. This is the recall ceiling of Section 4.4 appearing as attrition rather than as a metric, and it is a property of the candidate rule, not of any ranker. 600 436 further episodes are dropped for having fewer than two candidates and 99 for having none, leaving 538 474 prediction points.

Mean candidate-set size is 8.26 with 1.13 positives, and evaluation covers 70 565 test prediction points.

This is an observational target. A lecture consumed after an episode may have been chosen by the learner, required by a curriculum, or surfaced by an unobserved platform policy. We estimate historical ranking association, and we do not claim to estimate the effect of recommending anything. No causal interpretation of Task B is warranted, and we make none.

6 Student Models and Recommendation Models

Six model families appear in this study. For Task A, the response-history and elimination-aware predictors are operational student models: each maps the learner's observed history and the next item context to a probability of a correct response. Their paired difference evaluates whether elimination should be represented in that model. For Task B, the same state variables become user features in a learning-to-rank model for adaptation. Two deliberately trivial baselines calibrate the tasks, and a sequential family asks whether representing event order changes the result. All models are evaluated on identical rows.

Figure 9 shows the only structural difference that matters in the whole paper.

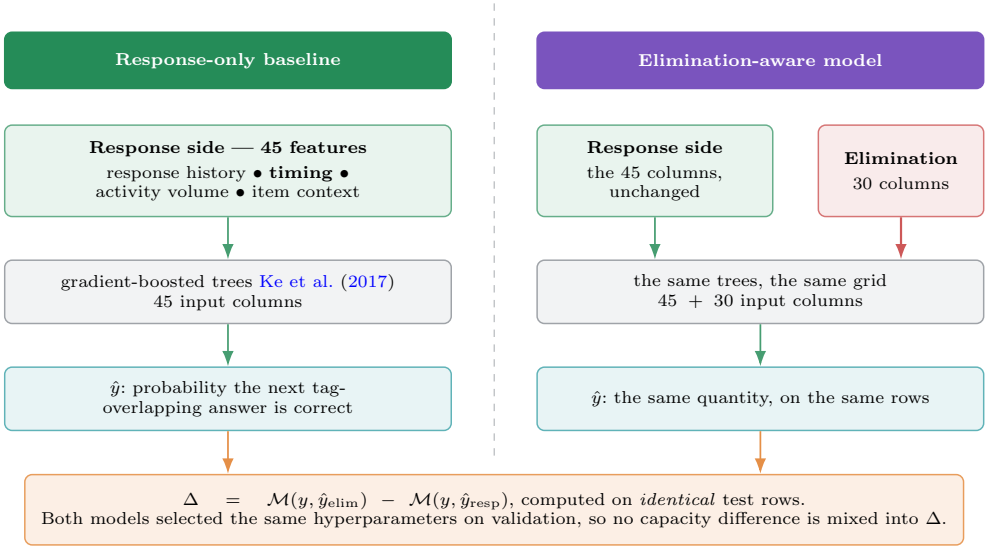


Fig. 9 The only structural difference in the study. The two models share the algorithm, the hyperparameter grid, the splits, the early-stopping rule and the evaluation rows. The elimination-aware model receives 30 additional columns, drawn in red, and nothing else changes. This is what makes the resulting difference interpretable as a feature effect: with one degree of freedom there is only one thing the difference can be attributed to. The three timing features inside the shared green block are the positive control of Section 3.3 — they are present in *both* models, which is why removing them is a control on the common feature space and not an elimination ablation.

6.1 Popularity

On Task A, the popularity baseline predicts the constant training base rate for every row, ignoring the learner entirely. On Task B, it ranks lectures by their consumption frequency in the training split alone.

A constant predictor cannot rank Task A rows, so its ROC-AUC is 0.5 by construction and it appears in the tables as a floor rather than as a competitor. On Task B, by contrast, popularity is a serious baseline, for the reason Cremonesi et al. (2010) identified: when consumption is concentrated on a small part of the catalogue, ranking by frequency is hard to beat. We include it as a first-class comparator for that reason.

6.2 Tag-only

The tag-only baseline predicts the smoothed per-tag correctness rate of the target bundle on Task A, fitted on training rows only, and ranks by tag-overlap score on Task B.

Its role is to isolate how much of the achievable performance comes from item difficulty alone, with no learner-specific information whatsoever. The gap between tag-only and response-only is the quantity we later call the *response-history gain*, and it is one of the two internal yardsticks against which the elimination increment is measured.

6.3 Response-history student model

The response-only model is a gradient-boosted decision tree ensemble Ke et al. (2017) over 45 features spanning four groups: response correctness history, timing, activity volume, and item context for the upcoming bundle. Table 5 lists them and Online Resource 1 gives the complete column-level inventory.

Gradient boosting is the right family for this comparison for three reasons. It handles heterogeneous tabular features without scaling or encoding work; it is not restricted to additive effects, since every split conditions on the path above it, so an interaction between elimination and response state is exactly the kind of structure it can build; and, configured as we configure it, it fits deterministically, which removes training-seed variance from the tabular comparison entirely.

This model is the baseline the hypothesis must beat. Following the warning that Dacrema et al. (2019) and Rendle et al. (2020) issue about under-tuned baselines, it receives exactly the same treatment as the model under test: the same search grid, the same early stopping, the same rows, the same splits.

6.4 Elimination-aware student model

The elimination-aware model is identical to response-only with the 30 elimination features appended. Same algorithm, same rows, same splits, same search grid, same early stopping. The only difference between the two models is the feature set.

This is the entire experimental manipulation, and its narrowness is the point. When two models differ in architecture, in tuning and in features at once, an observed difference cannot be attributed to any of them. Here there is one degree of freedom, so the difference means one thing.

6.5 Sequential models

The tabular models see summaries of windows. A sequential model sees the order of events, and could in principle extract structure that a window mean destroys — for example, that elimination early in a session predicts differently from elimination late in one. We therefore run a single-layer GRU (hidden size 128, dropout 0.2) over learner episode sequences, with and without the elimination group appended to each timestep.

Two caveats govern how these results may be read, and both are consequences of logged amendments rather than of choices made after seeing outcomes.

They are not comparable to the tabular models. A compute amendment restricts the sequential family to a 15 000-learner subsample with sequences truncated to 100 steps, so its absolute values sit well below the tabular models'. The comparison the sequential experiment supports is internal only: elimination-aware GRU against response-only GRU, on the same subsample.

The first version had a confound. The elimination features were initially added to a much poorer response representation than the tabular model receives. A gain measured that way might reflect nothing more than headroom: a weak baseline leaves room that any additional feature can fill. A further amendment therefore repeats the comparison with the response side raised to comparable richness. Both variants are run at three seeds, and Section 8.8 reports what happened.

6.6 Rankers for Task B

Task B uses LambdaRank Burges (2010) over the learner-state features plus five candidate-side features: tag overlap, tag score, lecture popularity, video length and part match. The objective optimises a ranking metric directly rather than a pointwise loss, which is appropriate when the evaluation is NDCG.

Note that Task B contains no upcoming-*question* features, because there is no upcoming question in the task definition — the model ranks lectures given the state, not given a next item. The no-target sensitivity analysis we run on Task A therefore has no counterpart here, and we flag this so that the absence of a row in Table 16 is not read as an omission.

7 Experimental Protocol

This section states how the data were split, how models were selected, how each metric is defined and used, how uncertainty was quantified, what was fixed in advance and what was amended, and what the leakage audit checked.

7.1 Splits and cohorts

Splits are user-chronological. Each learner's episodes are ordered in time and partitioned so that later activity is predicted by a model fitted on earlier activity, as Figure 10 shows. Accuracy and elimination prevalence are stable across the three splits, which Table 6 reports.

A user-chronological split is the right choice here because the alternative — a random split over rows — would place a learner's later episodes in training and their

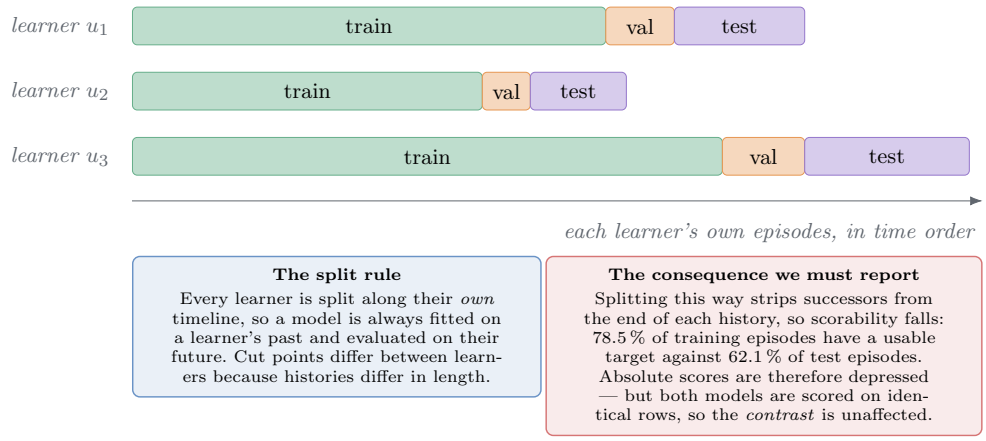


Fig. 10 User-chronological splitting. The alternative — a random split over rows — would place a learner's later episodes in training and their earlier episodes in test, letting the model learn from a future that a deployed system would not have. The cost of doing it correctly is a harder test set, which is why the absolute ROC-AUC values in this paper sit lower than figures reported under random splits on the same dataset. Only differences between models on identical rows are compared anywhere in the paper.

Table 6 User-chronological splits. Accuracy and elimination prevalence are stable across splits, so the temporal partition does not induce a distributional shift in the label or the behaviour under study.

Split	Episodes	Share	Accuracy	Elim. rate
Train	11 680 725	0.708	0.6192	0.1407
Val	1 612 988	0.098	0.6285	0.1560
Test	3 194 348	0.194	0.6261	0.1544

earlier ones in test, allowing the model to learn from a future the deployed system would not have. It has one consequence that must be stated, because it depresses the absolute numbers: splitting this way strips successors from the end of each learner's history, so scorability falls from 78.5 % of training episodes to 62.1 % of test episodes. The test set is enriched for episodes whose target lies further ahead, the regime where every model performs worst; the *contrast* between the two models is unaffected, since both are scored on identical rows.

Cohorts. Learners are stratified by elimination exposure, computed from training episodes only and frozen to an immutable artifact before any model was trained. The preregistered quartile thresholds proved degenerate: with 66.06 % of learners never eliminating, the first and second quartiles of the elimination rate are both exactly zero, so two of the four cohorts would have been identical. An amendment replaced them with zero, low and high, split at the median rate among eliminating learners. The replacement was frozen before any subgroup analysis ran.

7.2 Model selection

A four-point grid over the number of leaves and the learning rate is applied identically to both models, with selection on validation ROC-AUC and early stopping on the validation split. Table 7 reports every grid point for both models, not only the selected one.

Table 7 Validation ROC-AUC at every grid point. The elimination model is *worse* than the baseline at two of four configurations. A feature group carrying signal does not change sign with tree capacity; the configuration that was selected happens to be a favourable one.

Leaves	LR	Response-only	Elim.-aware	Δ
31	0.05	0.66333	0.66325	−0.00007
31	0.1	0.66435	0.66442	+0.00007
63	0.05	0.66447	0.66451	+0.00004
63	0.1	0.66442	0.66423	−0.00019

The grid is small by design: selection-induced variance scales with the size of the search, and that variance is not visible in any bootstrap interval computed at a fixed pair of fitted models. A larger search would have widened the noise floor that the effect must exceed while making the effect no easier to detect; fixing a small grid in advance keeps that floor low and honest.

7.3 Evaluation metrics

Task A is a binary classification problem and Task B is a ranking problem. The metric definitions are standard, and we cite them rather than restate them Järvelin and Kekäläinen (2002); we define only the two metrics that carry the primary comparisons, and we spell out the one non-standard property of cut-off-based metrics that several results in Section 8 depend on.

7.3.1 ROC-AUC

The area under the receiver operating characteristic curve is the primary Task A metric. Equation (4) gives it as the probability that a randomly chosen positive row receives a higher score than a randomly chosen negative row:

$$\text{ROC-AUC} = \frac{1}{|\mathcal{P}||\mathcal{N}|} \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{N}} \Big(\mathbb{1}[\hat{p}_i > \hat{p}_j] + \tfrac{1}{2} \mathbb{1}[\hat{p}_i = \hat{p}_j] \Big). \tag{4}$$

Where $\mathcal{P}$ and $\mathcal{N}$ are the sets of positive and negative rows, $\hat{p}_i$ is the model's predicted probability for row i, and the $\frac{1}{2}$ term splits credit for ties — which matters because the popularity baseline predicts a constant and is therefore entirely ties. ROC-AUC is preregistered as primary because it is invariant to the base rate, which differs

across the subgroups we report, and because it depends only on the ordering of scores rather than on their calibration.

7.3.2 Recall@K

For Task B, Recall@K measures how much of a learner's relevant set the top K positions recover:

$$\text{Recall@}K = \frac{\sum_{i=1}^{K} \text{rel}_i}{|\mathcal{R}|}. \tag{5}$$

Where $\text{rel}_i \in \{0, 1\}$ indicates whether the lecture at rank i was in fact consumed within the 24-hour window, $|\mathcal{R}|$ is the number of relevant lectures for that prediction point (mean 1.13), and $K \in \{1, 3, 5\}$, with $K = 3$ preregistered as primary.

7.3.3 Other metrics, and the roles they play

PR-AUC, log loss, Brier score, expected calibration error (ECE), NDCG@K and MRR appear in the tables as secondary summaries, defined and cited as standard. Two of them carry a specific role in this study. Log loss is sensitive to calibration and not only to ordering, which is why per-learner log-loss differences are the statistic of the permutation test in Section 7.4. ECE is reported because a predicted probability of 0.7 should be right about 70 % of the time; the sequential family's ECE gap is discussed in Section 8. Our NDCG implementation is verified against a reference implementation in the test suite, because a hand-written ranking metric is a classic source of silent error.

7.3.4 A property of cut-offs that the results depend on

A metric at K is well posed only where the candidate set holds at least K items. We therefore score each cut-off only on the prediction points that qualify, which means the number of scored points differs by K: 70 565 points at $K = 1$, 64 855 at $K = 3$, and 40 994 at $K = 5$.

This has a visible consequence in Section 8.6. NDCG appears to *fall* from $K = 3$ to $K = 5$ for every ranker including the baselines, which would be impossible if the same points were being scored. The comparison that is valid is down a column, between models at one cut-off. The comparison that is not valid is across columns, and we report the scored count beside every cut-off so that this cannot be mistaken.

7.4 Statistical inference

Rows within a learner are not independent. A learner who answers ten thousand questions contributes ten thousand correlated rows, and resampling rows would treat them as ten thousand independent observations, producing an interval far too narrow. Learners are the unit that varies, so learners are what we resample.

Paired user-level bootstrap. Uncertainty on the primary Task A contrast comes from 500 replicates, each resampling learners with replacement and recomputing the

metric difference on the resulting row set, as Equation (6) states:

$$\widehat{\Delta}^{(r)} = \mathcal{M}\big(y^{(r)}, p^{(r)}_{\text{elim}}\big) - \mathcal{M}\big(y^{(r)}, p^{(r)}_{\text{resp}}\big). \tag{6}$$

Where r indexes the 500 replicates, $y^{(r)}$ are the labels of the resampled row set, $\mathcal{M}$ is the metric (here ROC-AUC), and the two models' predictions are evaluated on *the same* resampled rows, which is what makes the comparison paired.

The model-fitting step is held fixed across replicates. This is a deliberate limitation and we return to it in Section 9: the interval describes test-set sampling variation at a fixed pair of fitted models, and it does not describe the variation that model selection introduces.

Intervals are percentile intervals over those replicates; we do not apply a bias or acceleration correction, and we report the standard error alongside so that the interval can be read either way.

Monte Carlo stability. 500 replicates is modest by current convention, so we repeated the bootstrap with 5 000, changing nothing else — same resampling scheme, same fitted models, same saved predictions, same percentile construction. The interval endpoints moved in the seventh decimal place and the minimum detectable effect with them; Online Resource 1 tabulates both. We report the preregistered 500-replicate analysis as primary and the 5 000-replicate run as a sensitivity check, rather than substituting one for the other, because the replication count was fixed in advance and the useful result is that it did not matter. This shows that Monte Carlo error at 500 was not affecting the conclusion; it does not establish that 500 replicates would suffice for a different estimand.

Paired permutation tests. Both are sign-flip tests under exchangeability of the two models within a learner, and both are two-sided.

The first targets the primary endpoint. For each of 2 000 draws, each learner's pair of prediction vectors is exchanged with probability 1/2 and the *pooled ROC-AUC difference* is recomputed exactly as the point estimate is. This matters because ROC-AUC is not a mean of per-learner quantities — it is not decomposable into per-learner contributions at all — so a test built on per-learner differences cannot have it as its statistic. An earlier version of this protocol reported only the second test below, which left the preregistered primary metric without an inferential procedure of its own.

The second, retained as a secondary endpoint, flips the sign of each learner's mean log-loss difference over 2000 permutations and recomputes the mean. Log loss is decomposable per learner and is sensitive to calibration rather than to ranking alone, so the two tests answer different questions and we report both.

What the design can resolve. An interval that spans zero bounds an effect; it does not establish absence, and the distinction turns on what the design could have seen. We therefore report a minimum detectable effect, taken from the learner-level bootstrap standard error $\widehat{\text{SE}}$ under a normal approximation:

$$\text{MDE} = \big(z_{1-\alpha/2} + z_{1-\beta}\big)\,\widehat{\text{SE}} \approx 2.802\,\widehat{\text{SE}}, \tag{7}$$

at $\alpha = 0.05$ two-sided and power $1 - \beta = 0.80$. It is the smallest true difference this design would detect on 80 % of samples. Two caveats belong with it: the normal approximation is doing work at this scale, and $\widehat{\mathrm{SE}}$ is itself conditional on the selected pair of fitted models, so the MDE inherits that conditioning. We compare the observed increment and the positive control against it in Section 8.

Effect size and share improved. We report paired Cohen's d and the share of learners for whom the elimination-aware model is better. Both are included because on a test set of this size a difference far below any practically relevant threshold will still reach conventional significance thresholds, so a p-value alone would be misleading in the direction that flatters the hypothesis.

7.5 Preregistration and amendments

Hypotheses, cohort definitions, primary metrics, the target definition and the search grid were fixed before any model was trained. Thirteen amendments were subsequently logged, each recorded with the empirical observation that motivated it. The first ten were made before the affected models were trained. Three were added at revision: cohort intervals (11) and the inference procedures of Section 7.4 (13), neither of which refits anything, and the within-cohort permutation control (12), the only model in the study trained after the results were known. The three most consequential are: (1) the episode scope changed from question to bundle level, because the KT4 stream provides no reliable question-level delimiter; (2) the cohort thresholds changed from quartiles to zero/low/high, because 66.06 % of learners never eliminate and the first two quartiles degenerate to zero; and (3) the sequential comparison was restricted to a 15 000-learner subsample, keeping the comparison internal only. The full log, with the remaining seven amendments, is in Online Resource 1.

Analyses that are not named in the protocol — the temporal gap strata and the per-part breakdown — are labelled exploratory wherever they appear, including in figure captions, so that a reader skimming the figures cannot mistake them for confirmatory results.

7.6 AI-assisted manuscript preparation

OpenAI Codex was used after completion of the analyses to assist with language editing and restructuring of the authors' existing manuscript for journal submission. It was not used to collect data, execute the reported analyses or generate empirical results. The authors reviewed and verified the complete manuscript and take responsibility for its content.

7.7 Leakage and correctness audit

Four families of automated check run over the full data, and all pass; the detailed results are in Online Resource 1. In brief: no (learner, episode) pair appears in more than one split, and every learner's splits are chronologically ordered. Window features recomputed by brute force on 20 000 sampled rows match the pipeline's prefix-sum output exactly, with no window extending beyond the current episode. Per-tag difficulty and lecture popularity equal their train-only recomputations exactly, while a

measurable difference from all-rows fits rules out a fit on everything. And a model trained on shuffled labels does not beat chance on test, which rules out the leakage pattern that a correctness signal baked into the features would produce.

8 Results

This section reports the empirical results in the order the protocol specifies. We first give the headline Task A comparison and its statistics, then examine the stability of that estimate across the hyperparameter grid, then place it against the internal yardsticks. We then report the ablations and the feature importances, which locate — or fail to locate — the signal inside the model. Task B follows, then the preregistered cohorts and the exploratory strata, then the sequential comparison, calibration, and the sensitivity analyses.

8.1 Task A: the elimination increment

Table 8 reports every model on both tasks, and Figure 11 shows the Task A column.

Table 8 Task A prediction and Task B ranking. Task B columns are the primary cut-off K=3. Response-only and elimination-aware selected identical hyperparameters, so their difference isolates the feature set.

	Task A (1 984 471 test prediction points)				Task B (70 565 test points)	
Model	ROC-AUC	PR-AUC	Log loss	ECE	NDCG@3	Recall@3
Popularity	0.5000	0.6542	0.6453	0.0142	0.6913	0.7749
Tag-only	0.5792	0.7175	0.6375	0.0286	0.4554	0.5911
Response-only	0.6700	0.7806	0.5973	0.0025	0.7440	0.8342
Elimination-aware	0.6702	0.7808	0.5973	0.0033	0.7435	0.8337

The first thing to record is a design outcome rather than a result. The response-only and elimination-aware models selected *identical* hyperparameters on validation, so the headline comparison isolates the feature set with no capacity difference mixed into it.

The addition of 30 elimination features changes test ROC-AUC by +0.000144. The supporting statistics are collected in Table 9 and each one points the same way.

- The paired user-level bootstrap interval is $[-0.000119, +0.000387]$, which **spans zero**.
- A paired sign-flip permutation test on the *primary* endpoint — 2 000 draws, flipping each learner's model pair, statistic the pooled ROC-AUC difference — gives $p = 0.311$. The same test on per-learner log-loss differences gives $p = 0.151$. Neither is significant at any conventional threshold.
- What the design can resolve is +0.000358 ROC-AUC: the smallest true effect it would detect on 80 % of samples at $\alpha = 0.05$, from the bootstrap standard error by Equation (7). The observed increment is 0.40 times that, so the correct reading is not that the effect is zero but that it is smaller than this design can resolve.

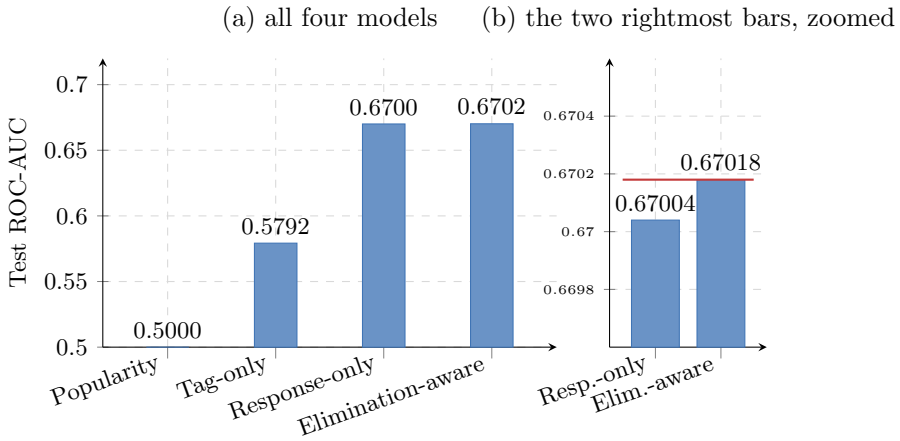


Fig. 11 Task A next-question correctness on 1 984 471 held-out prediction points. Panel (a): the step from a tag-only baseline to a response-history model is large and plainly visible, while the step from response-only to elimination-aware is +0.000144 — the two right-hand bars are the same height to four decimal places. Panel (b) zooms those two bars; the horizontal line marks the elimination-aware height. The whole study concerns the difference between them.

Table 9 Statistical and practical significance of the elimination increment on Task A. The effect is compared against yardsticks measured inside the same study rather than against an external convention alone.

Quantity	Value
Absolute ROC-AUC difference	+0.000144
95 % bootstrap CI (percentile)	[−0.000119, +0.000387]
Primary: permutation p (ROC-AUC)	**0.311**
Secondary: permutation p (log loss)	0.151
Min. detectable effect (80 %, α=0.05)	+0.000360
Paired Cohen's d	−0.0056
Learners improved	49.9 %
Share of the response-history gain	0.16 %
Share of the timing positive control	38 %
Hyperparameter grid span (validation)	0.00114

- Paired Cohen's d is −0.0056, which is not merely small but *negative*, meaning the per-learner log-loss differences slightly favour the baseline.
- The elimination-aware model is better for 49.9 % of learners — fewer than half.

The near-even split does not indicate a broadly distributed improvement across learners. It is compatible with a concentrated effect in a minority — which is what Section 8.7 goes on to examine — so we record it here and interpret it there.

8.2 The sign is unstable across the grid

Table 7, given in Section 7, reports the validation delta at every configuration in the search grid, for both models. The elimination-aware model is *worse* than the baseline at two of the four configurations and better at two. We report the instability as an observation rather than as a rule: sign reversal across capacity weakens confidence in a stable incremental effect, but it is not a theorem that an informative feature group must preserve its sign under retuning. The configuration that happened to be selected is one of the two favourable ones: selection on validation picks the draw that favours the richer feature set, and the test estimate inherits that selection without any interval reflecting it. The grid span, 0.00114 validation ROC-AUC, is roughly eight times the size of the effect under test.

8.3 The effect measured against internal yardsticks

External conventions for effect size are of limited use at this scale. A Cohen's d threshold designed for psychological experiments says little about whether a 1.4×10^{-4} ROC-AUC difference matters to a recommender. We therefore prefer yardsticks that the study itself produces, on the same rows, with the same model. Figure 12 places the increment against two of them.

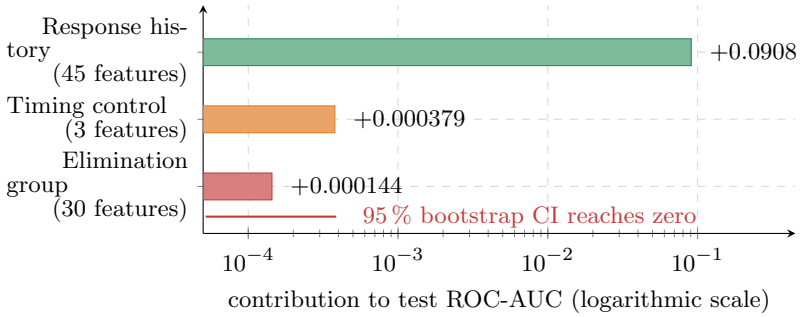


Fig. 12 The quantity under test, placed beside two yardsticks measured inside the same study, on the same rows, with the same model. Removing three response-side timing features costs +0.000379. The entire thirty-feature elimination group is worth +0.000144, and its confidence interval reaches zero. Response history is worth +0.0908 over a tag-only baseline. The axis is logarithmic because the three quantities span three orders of magnitude.

Yardstick one: the response-history gain. Moving from a tag-only baseline, which knows item difficulty and nothing about the learner, to the response-only model is worth +0.0908 ROC-AUC. This is what learner modelling is worth on this dataset for this task. The elimination group adds 0.16 % of it.

Yardstick two: the timing positive control. Removing three response-side timing features from the elimination-aware model, at its selected configuration, on the same rows, costs +0.000379. The thirty-feature elimination group is worth 38 % of what those three features contribute.

The second comparison is the one that answers RQ3, and it is worth stating precisely, because the obvious phrasing — *the instrument works, so there is nothing to*

measure — claims more than a single control can carry. What the control establishes is that this pipeline resolves at least one feature-group perturbation of this magnitude, end to end, under the identical procedure. It does not establish that any effect of that size would have been detected, because the elimination features are sparse, mutually correlated and mostly zero, and a control drawn from the dense response side need not have the same detectability.

The quantity that does generalise is the resolution itself. The design detects effects of +0.000358 and larger on 80 % of samples; the timing control, at +0.000379, sits just above that threshold, which is why it registers, and the elimination increment sits at 0.40 of it, which is why it does not. So the two yardsticks agree on a bound rather than on an absence: whatever the elimination group contributes on this task, it is smaller than the smallest effect this design was built to see. To put the increment in perspective for readers outside educational data mining, it is smaller than the fourth decimal place of a ROC-AUC score: the two models in Table 8 agree to four decimals and differ only in the fifth. We know of no accepted numeric threshold for practical relevance in this literature, and we do not propose one; the internal yardsticks above are used precisely because they do not require one.

8.4 Ablations: where the signal is not

Table 10 and Figure 13 report the feature-group ablations. Every one reuses the configuration selected for the full elimination-aware model, so each row isolates one feature group rather than a re-tuning difference.

Table 10 Feature-group ablations, all at the configuration selected for the full elimination-aware model. Positive contribution means the model is worse without the group. Two of the three elimination sub-groups are negative.

Variant	Test ROC-AUC	Contribution
Full elimination-aware model	0.67018	—
− correct-option flags	0.67007	+0.000112
− erase / undo counts	0.67034	−0.000160
− reconsideration features	0.67026	−0.000081
− timing (response-side control)	0.66980	+0.000379
− entire elimination group	0.67004	+0.000144
Tag-only	0.57923	+0.090949
Popularity	0.50000	+0.170184

Three observations follow, in increasing order of how much they constrain the hypothesis.

First, two of the three elimination sub-groups have a negative contribution. The model scores higher *without* the erase and undo counts, and higher without the reconsideration features; removing the erase/undo counts produces the highest-scoring model in the entire study. A feature group that carries information

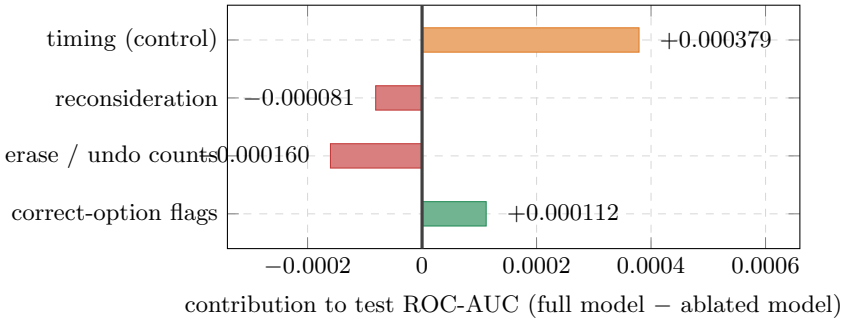


Fig. 13 Feature-group ablations, every one of them run at the configuration selected for the full elimination-aware model, so each bar isolates one group rather than a re-tuning difference. Bars right of the zero line are groups the model needs; bars left of it are groups the model scores higher without. The orange bar is the response-side timing control, present in both models, and it is the only bar of consequential length. Two of the three elimination sub-groups fall on the negative side.

does not behave this way; a group of uninformative features competing for splits with informative ones does.

Second, the sub-group contributions do not sum to the whole-group difference. Their sum is negative while the group total is positive. Exact additivity is not expected in a tree ensemble, since removing one group changes which splits the others receive, but a group carrying real signal would not typically produce parts whose sum has the opposite sign to the whole — this is what redundancy among uninformative features looks like.

Third, the timing row behaves completely differently. It is a control rather than an elimination ablation — it drops `time_since_last_activity`, `time_to_first_action` and `time_to_first_response`, which are present in *both* models — and it is the only bar in Figure 13 of consequential length, calibrating what a genuine effect of this kind looks like inside this pipeline.

8.5 Feature importance: the model does not use them

Table 11 lists the top features by split gain in the full elimination-aware model.

No elimination feature appears among the top 25; Table 11 shows the top ten and Online Resource 1 the complete list. The model's gain concentrates on target tag difficulty, bundle size, recent correctness rate and platform context. This is not a threshold artifact produced by cutting the list at an arbitrary rank: the gap between the last listed feature and the best-performing elimination feature is large, and the elimination group as a whole accounts for a negligible share of total split gain.

The importance result is worth stating in the strongest form the data support, because it answers an objection we return to in Section 9: the ensemble is not restricted to additive effects, it was free to split on any elimination feature in any combination at any depth, and it did not find one worth splitting on.

8.6 Task B: retrospective lecture ranking

Table 12 reports every ranker at every cut-off and Figure 14 shows the recall columns.

Table 11 Top 10 features of the elimination-aware model by split gain, as a share of total gain. **No elimination feature appears in this list**, although the model was given all 30 of them. The complete list is in Online Resource 1.

#	Feature	Gain
1	`target_tag_difficulty`	34.6 %
2	`target_n_questions`	12.3 %
3	`w10_correctness_rate`	10.4 %
4	`platform`	9.4 %
5	`t30_correctness_rate`	8.3 %
6	`source`	7.8 %
7	`target_part`	2.7 %
8	`prior_episode_count`	2.0 %
9	`w10_mean_duration`	1.5 %
10	`time_to_first_response`	1.3 %

Table 12 Retrospective lecture ranking, all rankers and cut-offs. Cut-offs are scored on different point counts (70 565 / 64 855 / 40 994 at $K = 1, 3, 5$), so values may be compared down a column but not across columns.

Ranker	R@1	NDCG@1	R@3	NDCG@3	R@5	NDCG@5	MRR
Popularity	0.5788	0.6027	0.7749	0.6913	0.7837	0.6136	0.7277
Tag-only	0.2664	0.2744	0.5911	0.4554	0.5364	0.3232	0.4960
Response-only	0.6201	0.6438	0.8342	0.7440	0.8620	0.6785	0.7660
Elimination-aware	0.6197	0.6433	0.8337	0.7435	0.8621	0.6781	0.7657
Δ elimination	−0.00035	−0.00052	−0.00047	−0.00042	+0.00006	−0.00041	−0.00030

The elimination-aware ranker is worse than the response-only ranker at five of six recall and NDCG cut-offs and at MRR, including the primary metric (NDCG@3 −0.000418). The single positive value, Recall@5, is the smallest magnitude in the table. Given all thirty elimination features and free choice over them, LambdaRank used 2 of them for any split, and the model it built with those two ranks slightly worse than the model built without any.

Two structural properties of the task bear on this result.

The tag-overlap score is the worst ranker of the four. The very quantity that defines the candidate set ranks within that set worse than popularity, worse than the response-only model, and worse than the elimination-aware one. Tag alignment does useful work as a filter and almost none as a score.

Popularity is strong. Consistent with the recommender-systems findings reviewed in Section 2 Cremonesi et al. (2010); Abdollahpouri et al. (2019), a learner-state model buys roughly 5.9 recall points at $K = 3$ over ranking by popularity alone, and it is inside that narrow interval that the elimination question was posed.

Finally, a reminder about the shape of Table 12 that Section 7.3 anticipated. Cut-offs are scored on different numbers of prediction points, because a metric at K is well

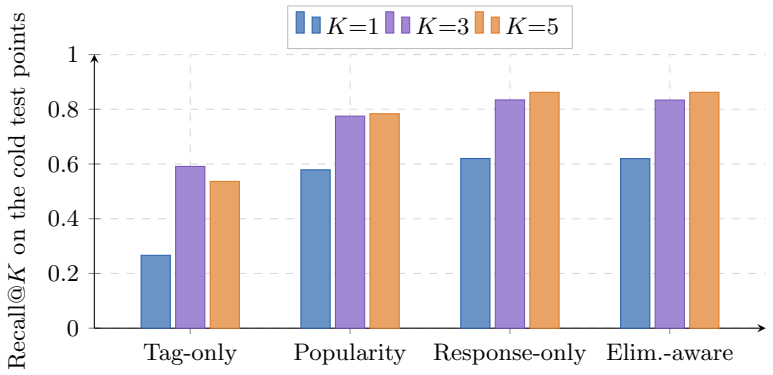


Fig. 14 Retrospective lecture ranking over 70 565 test prediction points. Two structural facts dominate the task. Popularity is a strong ranker, and the tag-overlap score that defines the candidate set is the weakest of the four — tag alignment does useful work as a filter and almost none as a score. The response-only and elimination-aware bars are indistinguishable at this resolution; numerically the elimination-aware ranker is worse at five of the six recall and NDCG cut-offs. Cut-offs are scored on different numbers of points, so values may be compared within a colour and not across colours.

posed only where the candidate set holds at least K items. This alone produces the apparent decline in NDCG from $K = 3$ to $K = 5$ for every ranker, baselines included. Comparisons down a column are valid; comparisons across columns are not.

8.7 Subgroups: the one result pointing the other way

Table 13 reports every stratum and Figure 15 shows the two that matter.

Table 13 Subgroup deltas. Preregistered strata are the three cohorts, single-question bundles and same-session targets; the gap strata are exploratory and carry no multiplicity correction. Intervals are paired user-level bootstrap 95 % CIs in units of 10^{-3}, given for the three preregistered cohorts only, because Section 9.6 argues from their ordering; the exploratory strata deliberately carry none.

Subgroup	Rows	Learners	Δ ROC-AUC	95 % CI
Cohort: zero elimination	295 206	29 322	−0.000165	[−0.42, +0.07]
Cohort: low	1 020 212	27 391	−0.000033	[−0.48, +0.29]
Cohort: high	669 053	13 953	+0.000537	[+0.18, +1.00]
Same session (≤30 min)	1 409 683	69 057	+0.000295	—
Cross session (>30 min)	574 788	31 817	−0.000332	—
gap <1 min	833 480	61 689	+0.000339	—
gap 1–30 min	576 203	46 808	+0.000235	—
gap 30 min–24 h	239 227	22 836	−0.000207	—
gap >24 h	335 561	24 681	−0.000411	—
Single-question bundles	1 583 252	66 763	+0.000189	—
Multi-question bundles	401 219	21 942	−0.000002	—

The preregistered cohorts are ordered monotonically with elimination exposure, and the zero-exposure cohort sits at −0.000165. That stratum is the design's internal

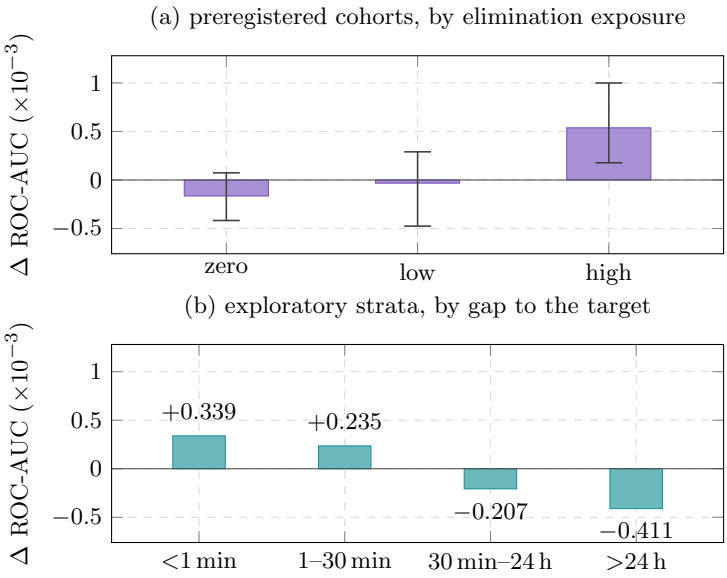


Fig. 15 Subgroup structure, both panels in units of 10^{-3} ROC-AUC. (a) The three preregistered cohorts are ordered as a dose–response would be. The zero-exposure cohort is the design's internal control: every elimination feature entry is zero for 99.47 % of those learners' rows, so almost no elimination information can reach the model, and the −0.000165 that appears there is the floor the fitting procedure produces on its own. Whiskers are paired user-level bootstrap 95 % intervals over 500 replicates, resampling learners rather than rows; Table 13 above gives every value and interval numerically. The zero and low intervals span zero; the high-exposure interval does not. (b) The exploratory temporal strata decay monotonically and cross zero; beyond 24 hours the elimination-aware model is the worse of the two. **Panel (b) carries no error bars, and none should be inferred.** Those strata were not preregistered, and attaching an interval to a stratum chosen after seeing the data would invite exactly the reading that the multiplicity discussion warns against. The eighteen comparisons carry no correction in the table; Section 9.6 reports what the high-exposure interval does under one.

control: those 66.06 % of learners performed no elimination in any training episode, and 99.47 % of their elimination feature entries are exactly zero — not all of them, because cohorts are assigned from training episodes while the rolling windows keep running into the test period, so a learner who starts eliminating later carries a few non-zero values. At that purity whatever difference appears is still essentially the floor produced by refitting with thirty near-constant columns added. Its interval, $[-0.000417, +0.000073]$, contains zero, which is what an internal control should do: the floor is consistent with no effect, and we do not read its negative point estimate as a measured harm.

The three cohorts are the only strata carrying intervals, because they are preregistered and because Section 9.6 argues from their ordering. The high-exposure cohort is the one stratum in the study whose interval excludes zero: $\Delta = +0.000537$, 95 % CI $[+0.000177, +0.000999]$, and $[+0.000087, +0.001218]$ after Bonferroni widening over all 18 subgroup comparisons, which also excludes zero.

A significant stratum sitting beside two non-significant ones does not by itself establish that the strata *differ*, so we test the difference directly. Resampling learners

within each cohort independently and taking the difference per replicate, the high-minus-zero contrast is +0.000705, 95 % CI [+0.000237, +0.001309], which excludes zero. The heterogeneity is therefore a finding in its own right and not an artefact of reading three intervals separately.

That is the one result in the study pointing toward the hypothesis rather than away from it, and it deserves a fair hearing. It does not overturn the primary result: the preregistered endpoint is the pooled comparison, whose interval spans zero, and a stratum of a null aggregate is not a substitute for it.

The obvious objection is that the cohorts encode learner type. A block of features that is near-constant within a learner can identify which cohort a row came from without saying anything about elimination, and a model can profit from that alone. We tested it rather than argued it (Table 14). Shuffling the elimination block among rows that share a cohort and a split leaves cohort membership perfectly recoverable — the block's distribution still differs by cohort, and a single split isolates high-exposure rows — while destroying the link between a row's own elimination behaviour and its outcome. Membership alone buys nothing. The high-exposure advantage does not merely shrink under the shuffle, it inverts to −0.000200, onto the same floor the zero cohort occupies, and pooled the permuted model is −0.000080: thirty shuffled columns cost a little, as they should. Whatever produces the high-cohort value needs the row-level signal intact.

Table 14 The within-cohort permutation control. The elimination block is shuffled among rows sharing a cohort and a split, so the features still say which cohort a row belongs to while saying nothing about that row's own elimination behaviour. Same 75 columns, same configuration, same rows. Deltas are against the identical response-only baseline, in units of 10^{-3} ROC-AUC, with paired user-level bootstrap 95 % intervals. If the high-exposure advantage were cohort membership, it would survive the shuffle. It does not.

	Elimination features		Shuffled within cohort	
Cohort	Δ	95 % CI	Δ	95 % CI
Zero elimination	−0.165	[−0.42, +0.07]	−0.166	[−0.32, −0.01]
Low	−0.033	[−0.48, +0.29]	+0.033	[−0.13, +0.22]
High	+0.537	[+0.18, +1.00]	−0.200	[−0.39, −0.02]

The honest statement is therefore narrow but firmer than resampling alone would make it: *if* an elimination effect exists it is concentrated in high-exposure learners, it survives resampling, multiplicity correction and a within-cohort permutation, and it is not detected in the zero- or low-exposure cohorts, whose intervals both contain zero. Section 9 weighs what remains: not learner type, which this control excludes, but the possibility that these features track a momentary condition rather than elimination reasoning as such.

The exploratory temporal strata decay monotonically and cross zero. Beyond 24 hours the elimination-aware model is worse than the baseline. This is what a short-lived state signal would look like, and it is consistent with elimination reflecting a momentary condition — hesitation, narrowing, uncertainty about a specific item — rather than a durable trait. We offer it as a hypothesis for a future study rather than as a result of this one. These strata were not preregistered, no multiplicity correction is applied, and the entire gradient spans less than the hyperparameter grid.

8.8 Seed variance exceeds the elimination effect in sequential models

The sequential experiment produced a finding, and it is an instability finding rather than a point estimate. Across three seeds — 42, 101 and 202, fixed in advance and not chosen after seeing results — (Table 15, Figure 16), the per-seed elimination delta ranges over an order of magnitude, and the across-seed standard deviation exceeds the across-seed mean for both the poor ($+0.001665 \pm 0.001846$) and the rich ($+0.000256 \pm 0.000272$) response baseline. Neither comparison is distinguishable from zero. Any single seed in Figure 16, reported alone, would have supported a confident conclusion in whichever direction that seed happened to fall; the first seed we ran would have supported a positive one.

Table 15 Sequential GRU models, three seeds each. Absolute values are not comparable to the tabular models: these use a 15 000-learner subsample under the compute amendment. The comparison is internal only, and it does not resolve.

Variant	ROC-AUC	SD over seeds	ECE
Response-only (poor)	0.62753	0.001231	0.0234
Elimination-aware (poor)	0.62919	0.002306	0.0206
Response-only (rich)	0.63287	0.000794	0.0442
Elimination-aware (rich)	0.63313	0.000764	0.0375
Elimination delta, mean ± SD over seeds			
vs. poor baseline		$+0.001665 \pm 0.001846$	
vs. rich baseline		$+0.000256 \pm 0.000272$	

The preregistered fair-baseline question — whether a sequential elimination gain reflects headroom or sequence structure — is therefore **not answered** by this study. We state the limit of what three seeds support precisely: under this architecture and this subsample, the observed seed-to-seed variation exceeds the observed mean delta, so the design cannot resolve whether a small sequential effect exists. We deliberately do not phrase this as a bound on the effect, because with three seeds the standard deviation is itself estimated from two degrees of freedom and is not known precisely.

Planning recommendation. The shortfall can be made concrete, as a planning estimate rather than a guarantee. *If* the observed seed means and standard deviations are representative, and seeds are independent draws with the same mean and variance,

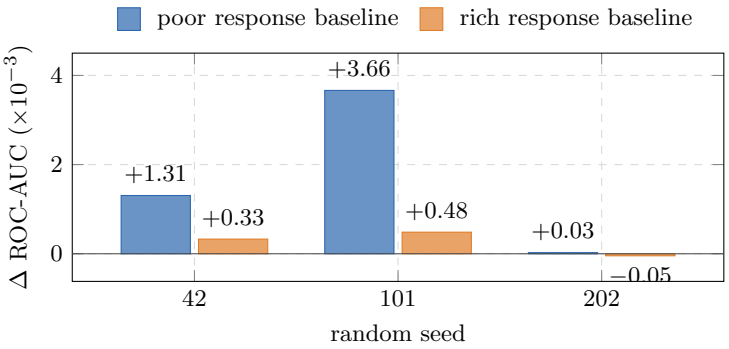


Fig. 16 Why the sequential comparison is reported as unresolved. Each pair of bars is one random seed, and the per-seed difference ranges over an order of magnitude while the across-seed standard deviation exceeds the mean for both response baselines. Any single bar, reported alone, would have supported a confident conclusion in whichever direction its seed happened to fall.

then the two-sided 95 % t-interval on the rich-baseline delta is $[-0.000421, +0.000933]$ at three seeds, and would first exclude zero at seven seeds; the poor baseline would require eight. Both figures inherit the imprecision of a variance estimated from two degrees of freedom, so they indicate an order of magnitude for the required budget rather than a threshold Dodge et al. (2019). The procedure that produced the figure — run three seeds, compute the t-interval, solve for the n at which it excludes zero — is general, and it turns "more seeds would be needed" into a number that can be budgeted.

We regard this instability finding as more valuable to the community than a single-seed point estimate would have been Bouthillier et al. (2021); Reimers and Gurevych (2017); Henderson et al. (2018). A point estimate would have been one more small effect in a literature that reports many of them; the measurement that a standard practice in the field produces different conclusions from different draws of the same computation is a warning that transfers to every sequential-model comparison on this dataset, not only to the elimination question.

The two model families have opposite reproducibility profiles. Refitting the boosted models under different seeds reproduces results exactly, because with bagging and feature fractions both at 1.0 and deterministic mode enabled the fit is a function of the data and the hyperparameters alone. Three seeds produced bit-identical results. We stress that this is exact reproducibility rather than robustness: it means the training seed is not a source of variance for the tabular comparison, not that the tabular comparison is stable under perturbation. The variation that bears on interpretation there is selection-induced, and it is reported in Table 7.

8.9 Calibration

The ECE column of Table 8 is comparable within the tabular family but not across families. Sequential models range from 0.021 to 0.044 against 0.0025–0.0033 for the boosted models, because the GRU is trained on log loss with no calibration step, while the boosted fits on this objective come out close to calibrated here — a property we

measure rather than one guaranteed by the algorithm. The gap is a property of the training procedures and no comparison in this study depends on it. We note without drawing an inference that within the sequential family the better-ranking rich baseline is the *worse*-calibrated one.

8.10 Sensitivity analyses

Table 16 reports four variants of the primary comparison. The purpose is to establish that the null is not an artifact of one particular protocol choice.

Table 16 Amendment 6 sensitivity: the harder variant in which no attribute of the upcoming question is available at prediction time. Both models lose substantially; the elimination increment does not change character.

Target features	Resp.-only	Elim.-aware	Δ
Present	0.67004	0.67018	+0.000144
Removed	0.64537	0.64607	+0.000698
Cost of removal	−0.024669	−0.024116	

Removing target-item features. Amendment 6 defines a harder variant in which no attribute of the upcoming question — its part, its size, its tag difficulty — is available at prediction time, the setting a deployed system faces when it must act before knowing what the learner will attempt next. Both models lose roughly 0.025 ROC-AUC, confirming that target-item attributes carry much of the total signal; the elimination increment remains of the same order and the same practical irrelevance.

Same-session targets. Restricting to targets within 30 minutes raises both models substantially (response-only 0.68118 against 0.67004 overall), consistent with short-horizon prediction being easier, and leaves the increment at +0.000295.

Structural single-question bundles. On the 1 583 252 test rows whose bundle contains exactly one question — which removes any confound from multi-question aggregation — the increment is +0.000189.

Per-part breakdown. The exploratory breakdown by TOEIC part is reported in Online Resource 1. The spread across parts is wider than the pooled effect. This is expected under a null with eighteen comparisons, and it is the reason we attach no evidential weight to individual strata. A reader who selected the most favourable part and reported it alone could construct a positive finding from these data, which is exactly the practice that preregistration is designed to prevent.

9 Discussion

The results are consistent across two tasks, four ablations, eighteen subgroups and four sensitivity variants. This section explains what we think they mean, what they do not mean, and what evidence would have led us to the opposite conclusion.

9.1 What the result means for student modelling and adaptation

The central result concerns model content, not model architecture. Adding thirty summaries of option-elimination history does not materially change the pooled student-state estimate once correctness, response timing, activity volume and item context are represented. This is evidence against treating the two logged actions as a generally useful additional component of an EdNet-KT4 student model under the evaluated representation. It is not evidence that process data as a category are uninformative: the positive control shows that another process signal, response timing, is detectably useful in the same pipeline.

The distinction between student modelling and adaptation is equally important. Task A tests the representation directly through its near-term predictive consequence. Task B tests one use of that representation, but personalized ranking also depends on the content model and the candidate generator. Its negative result therefore cannot be attributed to the learner representation alone. The measured recall ceiling demonstrates the separation empirically: 33.15 % of observed lecture consumption is unavailable to every ranker, including a hypothetical ranker with a perfect student model. For adaptive-system evaluation, representation quality and action-set coverage must be reported as different quantities.

9.2 The effect is smaller than the procedure that measured it

The single most useful way to understand the size of the elimination increment is to compare it not with an external threshold but with the sources of variation inside our own estimation procedure.

The four-point hyperparameter grid spans 0.00114 validation ROC-AUC, which is roughly eight times the effect. The bootstrap interval includes zero. The zero-exposure cohort, where no elimination information can possibly reach the model, shows a difference larger in absolute value than the pooled effect. Only the training seed is not a source of variation, and only because the boosted fits are deterministic in this configuration — which is a property of the learner, not evidence of robustness.

An effect smaller than every measurable source of variation in the procedure that produced it is not an effect a practitioner can act on, whatever a significance test might say about it. And here the significance test does not say it either: $p = 0.311$ on the primary endpoint over 1 984 471 rows.

One thing we should not claim is that the sample size settles it. Large n does not by itself confer power on a clustered, paired comparison: the unit of variation is the learner, not the row, and the learner-level standard error is what sets the resolution. That resolution is +0.000358, and the observed increment is 0.40 of it. So the study was *not* powered to detect an effect of the size it observed, and the interval spanning zero should be read as a bound rather than as a demonstration of absence. What the design does support is the negative claim at a stated scale: an effect as large as the timing control would have registered, and nothing of that size is present.

9.3 Why the two tasks fail differently

It would be a mistake to read the two nulls as one finding repeated. They have different causes, and separating them is what the two-task design bought.

Task A shows no detectable incremental signal. The tested response-history representation already accounts for most of the predictive performance these models reach on next-question correctness. The move from tag-only to response-only is worth +0.0908, and elimination adds 0.16 % on top of a gain that is itself modest in absolute terms. Little headroom remains within the evaluated model families and feature representation, and elimination does not occupy it. We state it that way deliberately: what is exhausted is what these features and these model families can extract, not the information content of KT4 itself.

A natural objection is that headroom arguments assume additivity. Perhaps the elimination signal is strongly non-linear, visible only in interaction with particular response-history states, so that a headroom calculation would miss it. The ablations answer this directly, and the answer is worth spelling out because it is the strongest objection the study faces. A gradient-boosted ensemble is not restricted to additive effects: every split conditions on the path above it, so an interaction between elimination and response state is exactly the structure the model is free to build, within the expressive capacity and search budget of the ensemble. Given that freedom and all 30 features, *no elimination feature reaches the top twenty-five by split gain* (Table 11 and the complete list in Online Resource 1), and two of three sub-groups contribute negatively (Table 10). The ensemble had the capacity to represent such interactions, and the fitted model did not assign substantial split gain to the elimination features. That is weaker than saying no conditional effect exists — correlated features can share or suppress one another's gain, and the same caveat we apply to Task B applies here — but it does mean the objection is not supported by anything we can see in the fitted model. Headroom and split gain are two views of one observation, not two independent arguments.

Task B fails for a partly different reason. Even a perfect learner-state representation would have limited room there, because the task is dominated by candidate-side properties — popularity above all — and because the tag-aligned candidate rule discards a third of real consumption before ranking begins. Elimination is not merely unhelpful in this setting; it is marginally harmful, and the ranker assigns split gain to only 2 of its 30 available elimination features — a descriptive fact about this fit, not evidence that the other 28 carry no information, since correlated features can share or suppress one another's gain. The difference between the two failure modes is that Task A's null is about the behaviour and Task B's is substantially about the task construction.

9.4 What the assessment-research literature would have predicted

The assessment literature treats distractors as functional: Haladyna et al. Haladyna et al. (2002) review item-writing guidelines on the premise that how learners process the option set reflects what they know, and under that premise a learner who eliminates

the correct option has committed a specific, identifiable error that a model seeing the event should exploit. Our null must therefore be read carefully. It does not contradict that premise about elimination as a construct; it bounds elimination as a *logged, zero-cost affordance* on a commercial test-preparation platform. The assessment literature's claims concern thoughtful option processing under consequential stakes. The events EdNet logs include taps that cost nothing and may carry no deliberation at all. A null on the second is compatible with a real effect of the first, and a study that instruments deliberate elimination would test the construct rather than the affordance. This is the single most likely way a future study improves on ours without overturning it, and it is the boundary condition stated in Section 10.

9.5 What would have changed our conclusion

A null claim is only as strong as the alternatives it excludes, so it is worth stating explicitly what evidence would have led us to the opposite finding. We fixed these criteria as part of the analysis plan, not after seeing the results.

1. **An increment comparable to the timing control** — roughly 4×10^{-4} ROC-AUC or more — with a bootstrap interval excluding zero and a majority of learners improved, would have supported H1. We observed a quarter of that magnitude, an interval spanning zero, and fewer than half of learners improved.
2. **A stable sign across the hyperparameter grid** would have supported the effect even at small magnitude, because a consistent direction under varying capacity is difficult to produce by chance. We observed sign reversal at half the grid points.
3. **At least one elimination sub-group contributing positively and appearing among the model's high-gain features** would have located the signal even if the group total were small. We observed two of three sub-groups contributing negatively and no elimination feature in the top twenty-five.
4. **On Task B, a consistent positive change across cut-offs** — even a small one — would have supported H2. We observed the elimination-aware ranker losing at five of six cut-offs, and the ranker declining to use 28 of its 30 available elimination features.

Any one of these could have appeared under a true null through ordinary sampling variation. All four failing in the same direction, in a design that resolves effects of $+0.000358$ and larger, is what a bounded absence looks like: not proof that the effect is zero, but consistent evidence that nothing of the size we set out to detect is present.

9.6 On the monotone cohort ordering

The one result pointing toward the hypothesis deserves a fair hearing rather than dismissal. The cohorts are ordered as a dose–response would be, with the zero-exposure stratum lowest and the high-exposure stratum highest. If we had preregistered only the cohort analysis and reported it alone, it would read as a positive finding, and it would have been a defensible paper.

We do not report it that way. One objection we previously raised against it has since been answered, and answered in the ordering's favour; two others remain, and they are the ones that matter.

The objection that no longer stands is resampling. This revision computes the paired user-level bootstrap separately within each cohort (Table 13, Figure 15). The zero and low intervals span zero, as the pooled interval does. The high-exposure interval does not: $\Delta = +0.000537$, 95 % CI $[+0.000177, +0.000999]$. Widened by Bonferroni over all 18 subgroup comparisons in the study it becomes $[+0.000087, +0.001218]$, which still excludes zero. So the high-cohort value does survive resampling, and it survives the multiplicity correction its own family would impose. We report that plainly, because a preregistered stratum that clears both bars is not something a null paper is entitled to bury. Note also what this does to the internal control: the zero-exposure cohort's -0.000165 carries the interval $[-0.000417, +0.000073]$, which contains zero, so that stratum is best read as a noise floor consistent with zero rather than as a measured negative effect.

The second objection, the one we had considered decisive, has since been tested and did not survive. The pattern is exactly what one would expect if the elimination features acted as a coarse proxy for *learner type* rather than for the elimination process: high-exposure learners differ from low-exposure learners in ways a response-history model captures imperfectly, and a block of features near-constant within a learner can encode that membership without encoding anything about elimination itself. That explanation makes a prediction, and the prediction fails. Permuting the elimination block within cohort preserves membership exactly while removing only the row-level signal; if membership were doing the work, the advantage would persist. It disappears, and inverts (Section 8.7, Table 14). We had this argument the wrong way round, and our own control is what corrected it.

What remains is one reason and one open question.

The reason is that the quantity the ordering decomposes is not significant. The primary preregistered endpoint is the pooled comparison, not any stratum of it, and a subgroup of a null aggregate does not carry the hypothesis. On the question as posed — does adding elimination features improve learner-state modelling on this dataset — the answer remains no.

The open question is what the high-exposure signal is made of. The permutation excludes learner type. It does not separate elimination *reasoning* from a momentary condition that merely coincides with erasing: hesitation, low confidence, or simply more care taken over that particular item. Either would produce the same row-level dependence. Two things make the momentary reading the more plausible *as a hypothesis*. The temporal strata decay monotonically and reverse beyond 24 hours, which is what a short-lived state would look like and not what a durable trait would; and the association is confined to learners who eliminate often, which is where a state marker would have the most occasions to fire. Both observations are exploratory and neither identifies a mechanism. What we are entitled to report is the association itself — row-level, concentrated in high-exposure learners, robust to permutation — and to name the transient-state account as the leading candidate for a study designed to test it, rather than as something this study establishes.

Separating those two readings requires a design we did not run: one powered for the high cohort specifically, with items whose difficulty and distractor structure are known, so that elimination on a genuinely confusable item can be distinguished from elimination under time pressure. We flag it as the most promising target for follow-up work in Section 12. Two limits of the present control should be stated with it. It rests on a single permutation draw, so its interval reflects learner sampling and not the variability of the shuffle itself; and it speaks only to the cohorts, since the exploratory strata carry no intervals by design.

9.7 Implications for acquiring student-model evidence

The interaction-log literature that motivated the hypothesis shows process data revealing structure that outcomes hide Kizilcec et al. (2013); Anderson et al. (2014). Our result does not weaken that programme; it sharpens the evaluation required for each proposed signal. Fine-grained features arrive at a prediction task *after* the conventional features have been computed, and on a target where little headroom remains — next-question correctness on EdNet, for the model families we tested — the conventional features have already accounted for most of what those models achieve. That is why the elimination group contributes nothing detectable while the same machinery cleanly detects three timing features. Research should therefore move to targets where headroom has not been measured away: lecture consumption, where our Task B is dominated by candidate-side structure, or effortful behaviours on platforms where each action is deliberate. Fine-grained features are most valuable exactly where coarse features are weakest, and a headroom measurement of this kind is the instrument that locates such targets.

9.8 The positive control as a user-model evaluation standard

A methodological lesson should outlive this study. The positive control cost one extra model fit and converted an uninterpretable user-model comparison into a bounded measurement: "we found no effect" became "the effect, if any, is smaller than the contribution of three timing features measured on the same rows". We would now regard a null claim about feature value in this literature without an internal control as difficult to evaluate, because the reader cannot separate "the feature adds nothing" from "the pipeline measured nothing". Nothing in the design is specific to elimination: any study proposing a new observation, inferred attribute or context variable for a user model can push a small established feature group through the identical pipeline and report the two increments side by side. This is a low-cost way to distinguish evidence that a candidate signal adds little from evidence that an evaluation is simply insensitive.

9.9 What generalises beyond the hypothesis

Two measurements survive independently of what one concludes about elimination, and we expect them to be more useful to other researchers than the null itself.

The headroom result bounds what an additional behavioural feature can contribute to next-question correctness *for these model families on this representation.*

Response history buys +0.0908 ROC-AUC over tag-only, and the remaining headroom is small enough that a feature group would have to be substantially more informative than elimination to register at all. This is a statement about what our instrument leaves available, not about the information content of the dataset. Future work proposing a new behavioural signal on EdNet can position itself against this number rather than against an absolute score, which is a more informative comparison and a harder one to satisfy by accident.

The recall ceiling bounds tag-grounded recommendation. At 33.15 % of consumption unreachable, a tag-aligned candidate rule cannot serve roughly a third of what learners actually do, independently of ranking quality. This is a design constraint rather than a modelling failure, and it should be reported alongside absolute recall in comparable studies. Without it, a reported Recall@K cannot be interpreted, because the reader has no way to know what fraction of the achievable maximum it represents.

9.10 Why we report this at all

There is a straightforward argument that a study like this one should not be written up: it found nothing, and the field rewards findings. We disagree. The direction was attractive — a research group proposing to add elimination to a learner model would not have been doing anything eccentric — and closing an attractive direction with a bounded measurement saves that work. And the two ancillary measurements — the headroom benchmark and the recall ceiling — are useful regardless of the hypothesis. Both were obtained only because the study was designed to make the null interpretable, which is a general argument for designing null-capable studies rather than an argument about elimination.

9.11 Take-home message

Take-home message. For the evaluated EdNet-KT4 student models, representing option-elimination history does not produce a detectable pooled improvement in knowledge-state prediction or lecture ranking. Any increment is smaller than the contribution of three timing features measured through the same pipeline.

10 Threats to Validity

We organise the threats by the kind of inference each one endangers, and in each case we state what the threat would change if it were realised, rather than merely acknowledging it.

10.1 Construct validity

`erase_choice` is an interface event. We cannot distinguish uncertainty from strategy, exploration or accidental interaction, and we make no misconception claims anywhere

in this paper. If elimination in EdNet is predominantly incidental — learners tapping options without deliberation — then our null is about that population of actions and not about deliberate elimination in general. A study able to separate deliberate from incidental elimination, perhaps through response-time thresholds or through instrumented data collection, could therefore reach a different conclusion without contradicting ours.

We regard this as the most substantive limitation of the study, and it is not one that more data from the same source would fix.

10.2 Sparsity of the behaviour

On 53.3 % of test rows all thirty elimination features are exactly zero, because elimination is present in 14.48 % of episodes and the windows are short. The effective sample carrying any elimination signal is therefore much smaller than the headline row count suggests.

This is an honest property of the behaviour rather than a defect of the design — a feature that is zero most of the time is exactly what a sparse behaviour produces — but it limits the design's power for sparse-exposure learners specifically. It is also part of why we report the cohort breakdown: the high-exposure cohort is where the features are dense enough for an effect to show, and it is where the largest cohort delta appears.

10.3 Test-set composition

Scorability falls from 78.5 % of training episodes to 62.1 % of test episodes (Section 7), because user-chronological splitting strips successors from the end of each learner's history. Absolute test metrics are therefore depressed relative to figures published on EdNet under random splits; readers comparing ROC-AUC values across studies should account for this. The contrast between the two models is unaffected, since both are scored on identical rows.

10.4 Multiplicity

Eighteen subgroup comparisons are reported without correction. Under a true null the largest observed subgroup delta is expected to exceed the pooled value several times over through selection alone, so we attach no evidential weight to extreme strata individually, and we report all eighteen rather than a selection precisely so that the spread is visible. To make the size of the correction explicit: at eighteen comparisons a Bonferroni-controlled family-wise threshold would be $\alpha = 0.05/18 \approx 0.0028$. We do not apply it, because we compute no per-subgroup p-value and therefore declare no subgroup significant at any level; the number is given so that a reader who wishes to weigh an individual stratum knows what it would have to clear.

10.5 Model class

We evaluated gradient-boosted trees and a GRU. A different architecture might extract signal that neither does. We consider this unlikely given that the ablations find no identifiable contribution from any elimination sub-group and the importance analysis finds no elimination feature worth splitting on, but it is not excluded, and a transformer over the full event stream rather than over episode summaries would be a reasonable thing for a subsequent study to try. Three considerations bound this threat. First, the tabular model was not restricted to additive effects: gradient-boosted trees condition every split on the path above it, so the interaction structure a transformer might learn was already within the tabular model's reach, and it found nothing worth splitting on. Second, a transformer over the raw event stream would inherit the sequential experiment's unresolved seed variance, so it could not be evaluated more conclusively than the GRU unless the seed budget of Section 8.8 is spent first. Third, the positive-control design is architecture-agnostic: if a transformer found elimination signal, the same control comparison would validate the measurement, and we would recommend carrying it in any such attempt.

10.6 The sequential experiment does not answer its question

We separate this from the model-class threat because it is more specific. The preregistered fair-baseline question — whether a sequential elimination gain reflects headroom or sequence structure — is not answered here. Three seeds establish that the comparison is unstable and do not establish what it would converge to. A reader should treat RQ4 as open, and Section 8.8 gives the seed budget that would be needed to close it.

10.7 Single dataset, single platform

All conclusions are conditional on EdNet-KT4 and on this platform's interface, in which eliminating an option is a low-cost action. On a platform where elimination is effortful — requiring a deliberate gesture, or carrying a visible cost — the behaviour might be considerably more selective and correspondingly more informative. Cross-platform replication is the most direct test of whether our null reflects the behaviour or its interface, and we identify it in Section 12 as the second most promising direction.

11 Implications for Student-Model and Adaptive-System Design

Four recommendations follow for researchers and system designers deciding what evidence to represent in a student model and how to evaluate its use in adaptation. Each is stated with its cost because the value of a methodological recommendation depends partly on whether it can be incorporated into an ordinary modelling workflow.

11.1 Carry a positive control

A student-model feature group known to matter, ablated through the identical pipeline, on the identical rows, at the identical configuration, converts an uninterpretable null into a bounded measurement. Section 9 makes the case for it as a reporting standard; here we note only its cost, which is the argument against skipping it.

Cost: one extra model fit. This is enough to calibrate whether the evaluation can detect at least one modest representation change end to end.

11.2 Report the candidate-rule ceiling

Any adaptive retrieval or recommendation study that filters candidates by metadata should state what share of real consumption that filter makes unreachable. Absolute recall is not interpretable without it: a Recall@3 of 0.6 against a ceiling of 1.0 and the same number against a ceiling of 0.67 describe very different systems.

Cost: one join and one count, computed once per dataset and candidate rule. Ours is 33.15 %.

11.3 Move beyond strict tag alignment

Our ceiling is a statement about deployment viability, not only about metric interpretation. A system that cannot surface a third of what learners actually watch is not merely scoring poorly on a benchmark; it is structurally unable to serve those requests. The 583 orphan lectures carry no tag shared with any question, so no tag-based rule can reach them.

Future EdNet recommendation work intending to be deployable will need a candidate generator that does not depend on the tag graph alone. Three directions are available without new data: collaborative signal from co-consumption, embedding similarity over lecture metadata, and semantic tagging of the orphan catalogue.

For this task, closing the candidate-coverage gap is a more promising systems change than adding further learner-side features, because the latter can only reorder resources that the former makes reachable.

11.4 Check seed variance before believing a neural result

Our single-seed sequential comparison would have supported a confident conclusion. Three seeds showed that its standard deviation exceeds its mean. The direction of the single-seed result was an artifact of which seed we happened to run first.

Cost: two extra model fits to detect the instability, and roughly seven to eight in total to resolve a comparison of this size. That figure is specific to our architecture and subsample, but the procedure that produced it — run three seeds, compute the t-interval, solve for the n at which it excludes zero — is general, and it turns "more seeds would be needed" into a number that can be budgeted.

12 Limitations and Future Work

The threats to validity are in Section 10. This section is about what we would do next, in the order we would do it.

12.1 A design powered for the temporal gradient

The temporal gradient is the most promising direction we can identify from our own data: the effect is largest for targets within a minute, negative beyond a day, and decays monotonically in between — the shape a transient state signal would have. A study restricted to the immediately following episode, powered for high-exposure learners, and reporting per-cohort bootstrap intervals could establish whether that ordering reflects a real transient signal; our strata for it were exploratory, so we cannot answer it here. The within-cohort permutation control (Amendment 12, Table 14) has already excluded the learner-type proxy explanation, but it rests on a single permutation draw on one dataset; repeating it prospectively, over several draws and powered for the high cohort, is what would turn that exclusion into a replicated result.

12.2 Cross-platform replication

EdNet's elimination affordance is cheap to use. A platform on which elimination requires a more deliberate action would test whether our null reflects the behaviour or its interface — which is the single question most likely to overturn our conclusion, and the one our data cannot address. A positive result on such a platform would not contradict this study; it would locate the boundary of it.

12.3 Resolving the sequential comparison

RQ4 is open. Section 8.8 gives the budget: roughly seven to eight seeds per arm, against the three we ran. A study that spends that budget would resolve whether a sequential architecture extracts anything the tabular models miss, and it would do so at a cost that is entirely predictable in advance, which is the practical value of reporting the seed budget rather than simply noting the instability.

12.4 Beyond elimination

The headroom result suggests a more general question than the one we asked. If the tested response-history representation already captures most of what these model families reach on next-question correctness, then the interesting question is not which additional behavioural feature to try next, but whether the task itself is close to the ceiling of what *this class of models on these features* can do — and if so, which tasks on this dataset are not. We are careful not to state that as a property of the dataset: establishing an information-theoretic ceiling for KT4 would need a different argument than the one we have made, and a model family that reads raw event streams could well find headroom we cannot. We regard the question as a more productive framing for future work on KT4 than continued feature engineering against a target that may already be near its practical limit.

13 Conclusion

This study asked whether a fine-grained interaction trace should become part of a computational student model. Under a preregistered protocol, thirty features summarising option-elimination and restoration events were added to a strong response-history representation and evaluated on two uses of that model: near-term knowledge-state prediction and personalized lecture ranking.

The pooled evidence does not support the addition. Across 1 984 471 held-out prediction points from 70 666 learners, the Task A ROC-AUC increment is +0.000144; its learner-level bootstrap interval spans zero, the primary permutation test gives $p = 0.311$, and fewer than half of learners improve. On Task B, the elimination-aware ranker is marginally worse. The conclusion is appropriately bounded. The design would detect an increment of +0.000358 or larger with approximately 80 % power, and an internal positive control detects the smaller representation change produced by removing three timing features. Thus the evaluation is capable of registering a modest user-model signal, while the complete elimination block contributes only 38 % of the timing effect.

The high-exposure subgroup prevents an overbroad interpretation. Its positive increment survives the preregistered multiplicity correction and disappears when elimination features are permuted within cohort. This association is a specific hypothesis for replication in a design powered for frequent eliminators and short prediction horizons; it does not overturn the null pooled endpoint.

The two tasks also show why user-model and adaptive-system evaluation should not be conflated. Task A bounds the incremental value of the candidate student-model signal for the evaluated representations and model families. Task B is further limited by the content side: tag-aligned candidate generation leaves 33.15 % of observed lecture consumption unreachable before ranking begins. A better learner representation cannot repair an action set that omits the relevant resource.

The transferable contribution is therefore methodological. Candidate signals for a user model should be evaluated incrementally against an unchanged baseline, accompanied by a positive control that calibrates sensitivity, and separated from constraints imposed by the downstream adaptive mechanism. Under those conditions, a negative result becomes actionable evidence about what a model need not represent, rather than an inconclusive absence of improvement.

14 Supplementary Information

Online Resource 1: Supplementary methods and results (`ESM_1.pdf`). This file contains detailed artifacts that support the main text, generated from the same JSON artifacts and subject to the same prose-versus-data verification as this manuscript. It contains the full preregistration amendment log (all thirteen amendments with their empirical justifications); the complete column-level feature inventory; the full results of the leakage and correctness audit; the complete feature-importance list (the main text shows the top ten); and the exploratory per-part breakdown. The three most consequential amendments are summarised in Section 7, and the terminology change from “choice elimination” to “option elimination” is noted at first use in Section 1.

Acknowledgements

The authors thank the EdNet team for releasing the KT4 interaction level, whose granularity made this evaluation possible.

Statements and Declarations

Funding

No funding was received for conducting this study. The authors declare that no funds, grants or other support were received during the preparation of this manuscript, and that they have no financial interests to disclose.

Competing interests

The authors have no relevant financial or non-financial interests to disclose.

Author contributions

All authors contributed to the conception of the study, reviewed the manuscript and approved the final version. The corresponding author performed the data processing, analysis and initial manuscript preparation. All authors accept responsibility for the integrity of the work.

Ethics approval

This study is a secondary analysis of a publicly released, de-identified dataset. No new data were collected from human participants, no personally identifying information was accessed and learners are represented only by opaque identifiers. The study makes no inference about any individual learner and proposes no intervention.

Consent to participate

Not applicable. The study involved no recruitment or contact with human participants.

Consent for publication

Not applicable.

Data availability

The EdNet-KT4 dataset analysed in this study is publicly available from the dataset authors Choi et al. (2020). The preregistration, amendment log and derived analysis artifacts are available from the corresponding author on reasonable request. The dataset licence and the platform's conditions govern access to and reuse of the original event data.

Code availability

The code used for data processing, modelling, statistical analysis, manuscript table and figure generation, and consistency verification is available from the corresponding author on reasonable request.

Use of artificial intelligence-assisted tools

OpenAI Codex was used to assist with language editing and restructuring of the authors' existing manuscript for journal submission. It was not used to collect data, run the reported analyses or generate empirical results. The authors reviewed and verified the complete manuscript and take full responsibility for its content.

References

Anderson, A., Huttenlocher, D., Kleinberg, J., Leskovec, J.: Engaging with massive online courses. In: Proceedings of the International Conference on World Wide Web (WWW) (2014). https://doi.org/10.1145/2566486.2568042

Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B.: The unfairness of popularity bias in recommendation. In: Workshop on Recommendation in Multistakeholder Environments (RMSE) at RecSys (2019)

Bouthillier, X., Delaunay, P., Bronzi, M., Trofimov, A., Nichyporuk, B., Szeto, J., Sepah, N., Raff, E., Madan, K., Voleti, V., Kahou, S.E., Michalski, V., Serdyuk, D., Arbel, T., Pal, C., Varoquaux, G., Vincent, P.: Accounting for variance in machine learning benchmarks. In: Proceedings of Machine Learning and Systems (MLSys) (2021)

Baker, R.S.J.d., Inventado, P.S.: Educational data mining and learning analytics. In: Learning Analytics: From Research to Practice, pp. 61–75. Springer, New York, NY (2014). https://doi.org/10.1007/978-1-4614-3305-7_4

Böck, F., Ochs, M., Henrich, A., Landes, D., Leidner, J.L., Sedelmaier, Y.: Learner models: Design, components, structure, and modelling. User Modeling and User-Adapted Interaction **35**, 15 (2025) https://doi.org/10.1007/s11257-025-09434-4

Burges, C.J.C.: From RankNet to LambdaRank to LambdaMART: An overview. Technical Report MSR-TR-2010-82, Microsoft Research (2010)

Corbett, A.T., Anderson, J.R.: Knowledge tracing: Modeling the acquisition of procedural knowledge. User Modeling and User-Adapted Interaction **4**(4), 253–278 (1994) https://doi.org/10.1007/BF01099821

Cremonesi, P., Koren, Y., Turrin, R.: Performance of recommender algorithms on top-N recommendation tasks. In: Proceedings of the ACM Conference on Recommender Systems (RecSys) (2010). https://doi.org/10.1145/1864708.1864721

Choi, Y., Lee, Y., Cho, J., Baek, J., Kim, B., Cha, Y., Shin, D., Bae, C., Heo, J.: Towards an appropriate query, key, and value computation for knowledge tracing. In: Proceedings of the ACM Conference on Learning at Scale (L@S) (2020). https://doi.org/10.1145/3386527.3405945

Choi, Y., Lee, Y., Shin, D., Cho, J., Park, S., Lee, S., Baek, J., Bae, C., Kim, B., Heo, J.: EdNet: A large-scale hierarchical dataset in education. In: Proceedings of the International Conference on Artificial Intelligence in Education (AIED), pp. 69–73 (2020). https://doi.org/10.1007/978-3-030-52237-7_6

Dacrema, M.F., Cremonesi, P., Jannach, D.: Are we really making much progress? a worrying analysis of recent neural recommendation approaches. In: Proceedings of the ACM Conference on Recommender Systems (RecSys) (2019). https://doi.org/10.1145/3298689.3347058

Dodge, J., Gururangan, S., Card, D., Schwartz, R., Smith, N.A.: Show your work: Improved reporting of experimental results. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-IJCNLP) (2019). https://doi.org/10.18653/v1/D19-1224

Ghosh, A., Heffernan, N., Lan, A.S.: Context-aware attentive knowledge tracing. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD) (2020). https://doi.org/10.1145/3394486.3403282

Gervet, T., Koedinger, K., Schneider, J., Mitchell, T.: When is deep learning the best approach to knowledge tracing? Journal of Educational Data Mining **12**(3), 31–54 (2020)

Gelman, A., Loken, E.: The garden of forking paths: Why multiple comparisons can be a problem, even when there is no "fishing expedition". Technical report, Department of Statistics, Columbia University (2013). Available at http://www.stat.columbia.edu/~gelman/research/unpublished/p_hacking.pdf

Haladyna, T.M., Downing, S.M., Rodriguez, M.C.: A review of multiple-choice item-writing guidelines for classroom assessment. Applied Measurement in Education **15**(3), 309–333 (2002) https://doi.org/10.1207/S15324818AME1503_5

Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., Meger, D.: Deep reinforcement learning that matters. In: Proceedings of the AAAI Conference on Artificial Intelligence (2018)

Ioannidis, J.P.A.: Why most published research findings are false. PLoS Medicine **2**(8), 124 (2005) https://doi.org/10.1371/journal.pmed.0020124

Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of IR techniques. ACM Transactions on Information Systems **20**(4), 422–446 (2002) https://doi.org/10.1145/582415.582418

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.-Y.: LightGBM: A highly efficient gradient boosting decision tree. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 30 (2017)

Kizilcec, R.F., Piech, C., Schneider, E.: Deconstructing disengagement: Analyzing learner subpopulations in massive open online courses. In: Proceedings of the International Conference on Learning Analytics and Knowledge (LAK) (2013). https://doi.org/10.1145/2460296.2460330

Melis, G., Dyer, C., Blunsom, P.: On the state of the art of evaluation in neural language models. In: Proceedings of the International Conference on Learning Representations (ICLR) (2018)

Manouselis, N., Drachsler, H., Vuorikari, R., Hummel, H., Koper, R.: Recommender systems in technology enhanced learning. In: Recommender Systems Handbook, pp. 387–415. Springer, Boston, MA (2011). https://doi.org/10.1007/978-0-387-85820-3_12

Nosek, B.A., Ebersole, C.R., DeHaven, A.C., Mellor, D.T.: The preregistration revolution. Proceedings of the National Academy of Sciences **115**(11), 2600–2606 (2018) https://doi.org/10.1073/pnas.1708274114

Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L., Sohl-Dickstein, J.: Deep knowledge tracing. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 28 (2015)

Pelánek, R., Effenberger, T., Jarušek, P.: Personalized recommendations for learning activities in online environments: A modular rule-based approach. User Modeling and User-Adapted Interaction **34**, 1399–1430 (2024) https://doi.org/10.1007/s11257-024-09396-z

Pelánek, R.: Bayesian knowledge tracing, logistic models, and beyond: An overview of learner modeling techniques. User Modeling and User-Adapted Interaction **27**(3–5), 313–350 (2017) https://doi.org/10.1007/s11257-017-9193-2

Pelánek, R.: Leveraging response times in learning environments: Opportunities and challenges. User Modeling and User-Adapted Interaction **34**, 729–752 (2024) https://doi.org/10.1007/s11257-023-09386-7

Pandey, S., Karypis, G.: A self-attentive model for knowledge tracing. In: Proceedings of the International Conference on Educational Data Mining (EDM) (2019)

Pineau, J., Vincent-Lamarre, P., Sinha, K., Larivière, V., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Larochelle, H.: Improving reproducibility in machine learning research (a report from the NeurIPS 2019 reproducibility program). Journal of Machine Learning Research **22**(164), 1–20 (2021)

Reimers, N., Gurevych, I.: Reporting score distributions makes a difference: Performance study of LSTM-networks for sequence tagging. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP) (2017). https://doi.org/10.18653/v1/D17-1035

Rendle, S., Krichene, W., Zhang, L., Anderson, J.: Neural collaborative filtering vs. matrix factorization revisited. In: Proceedings of the ACM Conference on Recommender Systems (RecSys) (2020). https://doi.org/10.1145/3383313.3412488

Rahman, M.M., Ollington, R., Yeom, S., Ollington, N.: Generalisable sensor-free frustration detection in online learning environments using machine learning. User Modeling and User-Adapted Interaction **34**, 1493–1527 (2024) https://doi.org/10.1007/s11257-024-09402-4

Siemens, G.: Learning analytics: The emergence of a discipline. American Behavioral Scientist **57**(10), 1380–1400 (2013) https://doi.org/10.1177/0002764213498851

Simmons, J.P., Nelson, L.D., Simonsohn, U.: False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. Psychological Science **22**(11), 1359–1366 (2011) https://doi.org/10.1177/0956797611417632

Wilson, K.H., Karklin, Y., Han, B., Ekanadham, C.: Back to the basics: Bayesian extensions of IRT outperform neural networks for proficiency estimation. In: Proceedings of the International Conference on Educational Data Mining (EDM) (2016)

Xiong, X., Zhao, S., Van Inwegen, E., Beck, J.: Going deeper with deep knowledge tracing. In: Proceedings of the International Conference on Educational Data Mining (EDM) (2016)

Zhang, J., Shi, X., King, I., Yeung, D.-Y.: Dynamic key-value memory networks for knowledge tracing. In: Proceedings of the International Conference on World Wide Web (WWW) (2017). https://doi.org/10.1145/3038912.3052580