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.

Incorporating Feedback into Elo-Based Knowledge Tracing for Iterative Problem Solving

Sébastien Rebaudo

sebastien.rebaudo@univ-grenoble-alpes.fr

Grenoble Alpes University

Cédric d'Ham

Grenoble Alpes University

Sihem Amer-Yahia

French National Centre for Scientific Research

Research Article

Keywords: Knowledge Tracing, Learner Model, Elo Rating System, Adaptive Feedback

Posted Date: July 31st, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9837888/v1

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

Additional Declarations: No competing interests reported.

Incorporating Feedback into Elo-Based Knowledge Tracing for Iterative Problem Solving

Sébastien Rebaudo[1*], Cédric d' Ham[1] and Sihem Amer-Yahia[1]

[1]LIG, Univ. Grenoble Alpes, CNRS, Grenoble INP, Grenoble, 38000, France.

*Corresponding author(s). E-mail(s): sebastien.rebaudo@univ-grenoble-alpes.fr;
Contributing authors: cedric.dham@univ-grenoble-alpes.fr; sihem.amer-yahia@univ-grenoble-alpes.fr;

Abstract

We study the problem of modeling learner knowledge in digital environments where complex tasks are solved through iterative attempts supported by feedback. In such settings, standard Elo Rating Systems (ERS) treat all successful outcomes equivalently, failing to distinguish independent success from success achieved with assistance, which may lead to biased knowledge estimates. To address this limitation, we propose two feedback-aware extensions of ERS. The New Item (NI) model represents each (item, feedback level) pair as a distinct task, while the Probability Modifier (PM) model distinguishes between assisted and unassisted attempts and models feedback as a level-dependent shift in the probability of success within a Rasch-based formulation. These approaches reflect complementary assumptions about the role of feedback. We evaluate the models on 47,224 learner–item interactions from 296 university learners engaged in a complex iterative task. Results show that both NI and PM significantly outperform feedback-agnostic ERS models across multiple complementary evaluation settings, including robustness to learner-population variations, consistency across learners, and generalization to unseen learners, while preserving meaningful knowledge discrimination and stable difficulty estimates. These findings highlight the importance of explicitly modeling feedback in iterative learning contexts.

Keywords: Knowledge Tracing, Learner Model, Elo Rating System, Adaptive Feedback

1 Introduction

Solving complex tasks in digital learning environments places substantial cognitive demands on learners. In classroom settings, learners can rely on teachers to seek help and overcome difficulties. By contrast, in contexts where the teacher is not directly present, digital environments often provide automated feedback to support task completion, foster learning, and increase engagement (Hattie and Timperley 2007). A particularly interesting class of systems is iterative learning environments, in which learners can revise their solutions multiple times and receive feedback after each attempt until a correct solution is reached. Such environments introduce challenges for modeling learner knowledge, as observed performance may reflect both underlying competence and the assistance received (see Figure 1).

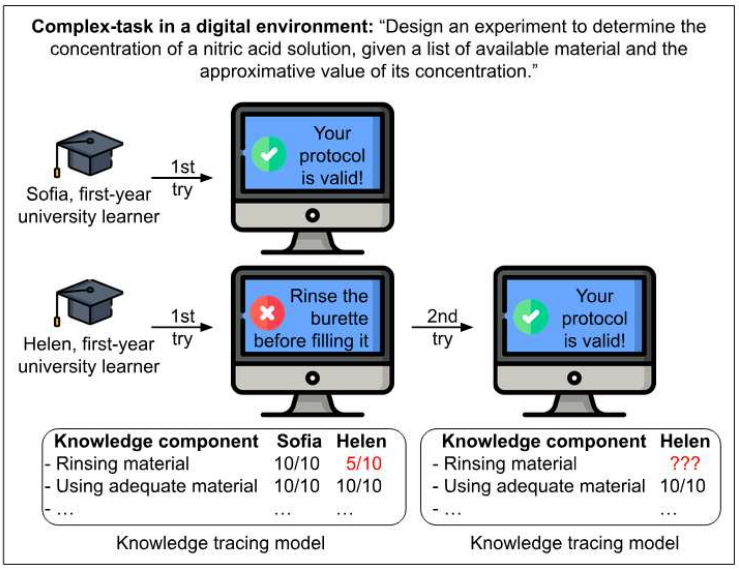


Fig. 1 Consider two learners engaged in the complex task of designing a chemistry experiment, where each attempt consists of submitting a protocol to the system. Although both learners ultimately succeed, their performances reflect different underlying knowledge states. Sofia's correct submission suggests that she likely possesses the required competence, whereas Helen's success may be attributable to the feedback she received.

This work addresses the challenge of accurately modeling learners' knowledge in contexts where automated feedback is provided during complex, iterative tasks.

Modeling learner knowledge from interaction data has been extensively studied in the literature on knowledge tracing. Traditional approaches, such as Bayesian Knowledge Tracing (BKT), explicitly represent latent knowledge states and provide mechanisms to incorporate external factors, including feedback, into knowledge estimates (Šarić Grgić et al. 2024). In parallel, the Elo Rating System (ERS) has been

adopted in educational settings as a lightweight and scalable alternative for estimating learner ability and item difficulty from performance data (Pelánek 2016). Due to its simplicity, its computational efficiency, and its intrinsic ability to model learning, ERS is particularly well-suited for real-time adaptation in interactive learning systems (Pelánek 2016; San Pedro and Baker 2021; Vermeiren et al. 2025).

Despite these advantages, standard ERS formulations treat all attempts equivalently, regardless of whether feedback was provided previously. This assumption becomes problematic in iterative learning environments, where feedback plays a central role in guiding learners toward correct solutions.

We investigate how automated feedback interventions should be represented when estimating learner knowledge in complex, iterative tasks.

A first modeling decision is to distinguish between attempts made without feedback and those made with feedback, as these correspond to qualitatively different learning conditions. Beyond this binary distinction, several perspectives can be considered regarding the role of feedback. One assumes that feedback fundamentally changes the task, such that attempts to address the same underlying problem under different feedback levels should be treated as distinct tasks; this assumption underlies the New Item (NI) model. An alternative view assumes that, once feedback is present, it does not alter the nature of the task itself but instead increases the probability of success in a graded manner depending on its level; this perspective is captured by the Probability Modifier (PM) model.

By combining a shared distinction between unassisted and assisted attempts with two different ways of modeling the effect of feedback levels, these approaches span a broad range of plausible assumptions about feedback in iterative learning.

NI and PM are two simple extensions that operationalize distinct assumptions about the role of feedback in iterative learning. Through an empirical evaluation on a real-world dataset of 47,224 learner–item interactions from 296 university learners, we compare the models using three complementary evaluation perspectives: robustness to learner-population variations, consistency across learners, and generalization to unseen learners. Results show that both models significantly improve the predictive accuracy compared to feedback-agnostic ERS-based models. Although the two approaches rely on different conceptualizations of feedback, they achieve highly comparable performance while preserving meaningful knowledge discrimination and stable difficulty estimates. These contributions provide practical and computationally efficient approaches for extending ERS to feedback-rich learning environments.

The data used in this study and the code for the proposed models are publicly available on the `code repository`[1] of Université Grenoble Alpes.

2 Related Work

Recent surveys distinguish traditional knowledge-tracing models from deep-learning approaches (Abdelrahman et al. 2023; Shen et al. 2024). While deep models often

[1] https://gricad-gitlab.univ-grenoble-alpes.fr/renaudos/ers_with_feedback

achieve higher predictive accuracy, traditional approaches—such as Bayesian Knowledge Tracing (BKT) and logistic or factor-analysis–based models—remain attractive for their simplicity, data efficiency, and interpretability. As noted by Pelánek, "deep machine learning [...] may not necessarily translate into deep human learning" (Pelánek 2025). Motivated by these considerations, we focus on lightweight models capable of tracking learner knowledge over time in practical educational settings.

A substantial body of work has addressed the integration of feedback and instructional support within BKT-based models. Using data from the ASSISTments platform, Wang et al. use the concept of partial credit (Wang and Heffernan 2013), discounting learner performance according to the amount of help received. Subsequent studies proposed alternative partial-credit formulations and reported improved prediction by incorporating hints or scaffolding items (Ostrow et al. 2015; Van Inwegen et al. 2015; Asselman et al. 2020). A recent review (Šarić Grgić et al. 2024) synthesizes these extensions, highlighting sustained efforts to model feedback and tutor interventions within the BKT framework.

In contrast, comparatively little work has explored feedback integration in other learner modeling approaches, notably the Elo Rating System (ERS), originally developed for rating chess players (Glickman and Jones 1999). While Pelánek notes that concepts such as partial credit could be applied to ERS (Pelánek 2016), we are not aware of any concrete formalization explicitly incorporating feedback into ERS-based knowledge tracing.

This gap is noteworthy because ERS possesses properties well suited for adaptive learning (San Pedro and Baker 2021; Vermeiren et al. 2025). ERS relies on few hyper-parameters, supports online updating, and is computationally efficient while achieving reasonable predictive performance (Vermeiren et al. 2025). It has been applied across diverse educational domains, including geography practice (Pelánek et al. 2017), Chinese-language learning (Zhang et al. 2023), programming exercise recommendation (Mangaroska et al. 2019), and language and mathematical practice (Vermeiren et al. 2025), though recent work has also identified potential biases and increased variance under some conditions (Bolsinova et al. 2025).

BKT and ERS differ in their modeling assumptions. BKT represents learning as probabilistic mastery transitions and, in its standard form, does not model item difficulty, whereas ERS models learner–item interactions as matches and updates both learner knowledge and item difficulty based on prediction error (Pelánek 2016). More specifically, ERS attributes learning to discrepancies between expected and observed performance. From a practical perspective, BKT requires non-trivial hyper-parameter estimation, while ERS operates with fewer hyper-parameters and minimal optimization.

Given the relevance of ERS for adaptive learning and the lack of feedback-aware extensions, this work addresses an open gap by proposing and evaluating methods for integrating feedback into ERS-based knowledge tracing.

3 Context and Data

3.1 Learning Context

To evaluate our feedback-aware ERS models, we use data that capture learning in environments characterized by three key properties: (i) the presence of complex tasks involving multiple interdependent decisions, (ii) the possibility for learners to iteratively revise and resubmit complete solutions, and (iii) the provision of feedback that can be used to improve subsequent attempts at the same task.

The LabNBook platform (d'Ham et al. 2026) meets these requirements. In this environment, first-year learners at Université Grenoble Alpes are asked to solve a complex task: designing an acid–base titration protocol. This task involves constructing a complete protocol by selecting, parameterizing, and ordering predefined actions, requiring coordination across multiple reasoning steps. For instance, a standard protocol requires selecting and ordering 12 actions and specifying 31 parameters (see Figure 2). Learners can submit their solutions multiple times and receive targeted feedback after each attempt, enabling iterative refinement of the same task.

Other existing platforms such as ASSISTments (Feng et al. 2009) share some of these characteristics but do not fully match our requirements. In ASSISTments, learners solve multi-step problems and receive feedback through scaffolding questions that break the task into simpler substeps, with the option to request hints. However, this setting does not involve the iterative resubmission of a complete solution to the same complex task, where feedback from a previous attempt directly informs subsequent revisions. Consequently, we deliberately selected the LabNBook platform for our experimental study.

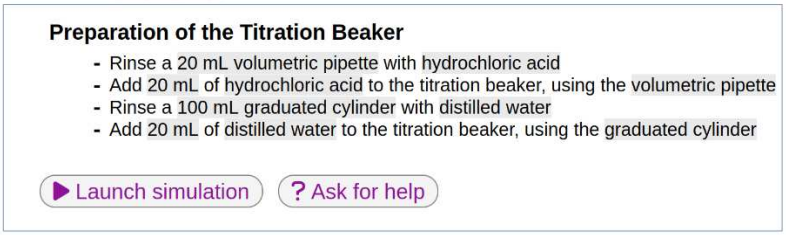


Fig. 2 An example of the beginning of a titration protocol

In LabNBook, after each submission, the protocol is automatically analyzed by the system to detect errors. If at least one error is identified, the platform provides feedback; otherwise, it returns simulated experimental results (see Figure 3). Due to the complexity of the task, most submissions contain at least one error and therefore trigger feedback. Consequently, while the first submission is typically made without feedback, subsequent submissions are generally informed by feedback provided on a previous attempt. Importantly, each feedback message targets a single error, selected according to a priority defined by domain experts.

Error detection is based on the Constraint-Based Modeling (CBM) framework (Ohlsson 1994). Each submission is evaluated against a set of 42 expert-defined boolean

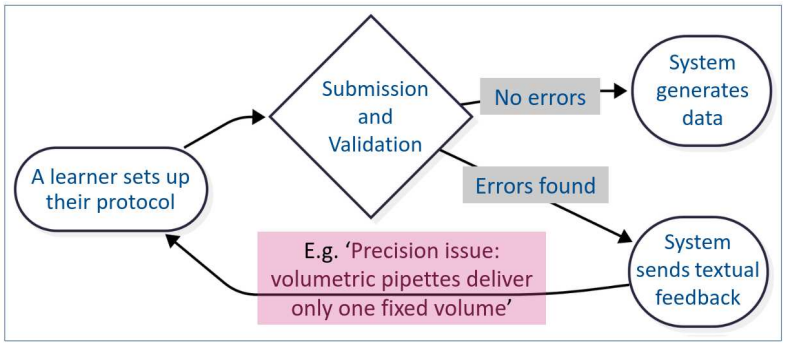


Fig. 3 Learner workflow when working on the LabNBook platform

constraints, where each constraint represents a specific requirement of the task. For example, the system verifies whether the volumetric pipette is rinsed before use in a titration protocol. A submission is considered correct only if all constraints are satisfied; otherwise, violated constraints correspond to detected errors. This approach allows for multiple valid solutions (those that satisfy all the constraints), in contrast to methods based on comparison with a fixed solution space.

In the remainder of this paper, we treat each constraint as an item. Thus, each protocol submission corresponds to a set of item-level evaluations, where items are assessed in a binary manner. At each submission, a learner interacts with multiple items, with at most one item being addressed with the support of feedback derived from the previous submission.

3.2 Data Collection and Dataset Structure

Data were collected during a real-world study conducted between October and December 2025 on the LabNBook platform, involving first-year undergraduate learners enrolled in chemistry and biology programs at Université Grenoble Alpes. Learners worked individually, without time constraints, on three exercises requiring them to design an acid–base titration protocol. Although the exercises differed in the available equipment, they relied on the same underlying set of items. We therefore consider all interactions as instances of a single complex task—protocol design—despite learners being free to switch between exercises.

The dataset comprises interactions from 296 learners with the item set. We define the unit of observation as an item-level interaction. Each interaction corresponds to a tuple (learner, item, submission). For each interaction, we record a binary outcome (success or failure) and a feedback level indicating the amount of assistance available to the learner at the moment of addressing the item. Feedback is encoded as an ordinal variable: a value of 0 indicates no feedback, while values from 1 to 5 correspond to increasing levels of support, ranging from minimal guidance to the explicit provision of the correct solution for the item. Figure 4 illustrates the structure of the collected data.

Importantly, interactions are recorded only when an item's status or the feedback level changes between two consecutive submissions. Thus, each interaction captures a

learner's active engagement with a specific item, rather than repeated observations of unchanged states.

	id_submission	timestamp	id_user	id_exercise	id_item	feedback_level	success
135	...	...	...	...	...	...	...
136	146888	1761044589	906993	1267195	4	3	False
137	146889	1761044590	906993	1267195	4	4	False
138	146891	1761044603	906993	1267195	3	0	False
139	146891	1761044603	906993	1267195	18	0	True
140	146892	1761044609	906993	1267195	3	5	False
141	...	...	...	...	...	...	...

Fig. 4 Anonymised dataset. For example, during submission 146889, learner 906993 engaged with item 4 using feedback at level 4 (high assistance), yet the attempt resulted in failure

In total, the dataset contains 47,224 learner–item interaction records. On average, each learner submitted 37 attempts and generated approximately 160 item-level interactions, with each interaction corresponding to exactly one of the 42 items. The items are further grouped into 11 task types, which we interpret as knowledge components. Each item is associated with a single task type, whereas each task type typically encompasses 1 to 6 items. All data have been anonymised and are available on the `code repository` of Université Grenoble Alpes.

Although the data were collected within the LabNBook environment, our approach is not tied to this specific platform. More generally, it applies to learning settings that satisfy three key conditions:

- (1) the presence of a complex task involving multiple interdependent decisions, together with a principled decomposition into smaller components (items), each associated with a knowledge component (e.g., via Constraint-Based Modeling),
- (2) the possibility for learners to iteratively revise and resubmit complete solutions to the same task, and
- (3) the availability of feedback messages for each item, ordered by level of assistance.

4 Modeling

We begin by introducing the standard Elo Rating System (ST) as applied to educational settings. We then present an extension of this model that addresses three common limitations. This model, denoted EN (for ENhanced standard ERS), serves as a baseline and point of comparison to our subsequent models. All proposed models involve hyper-parameters, whose estimation is described in Section 5.2.

Throughout this paper, we abstract away from the specific content of feedback and consider only its presence and level. That is, we model whether feedback was provided for a given item attempt and, if so, the corresponding level of assistance.

4.1 The Standard ERS (ST)

In the standard ERS (Pelánek 2016), the probability that learner k succeeds on item i is modeled using the standard logistic function σ:

$$P(S_{k,i} = 1) = \sigma\left(\theta_k - d_i\right) \tag{1}$$

where "$S_{k,i} = 1$" denotes the event "learner k solved item i", θ_k represents the learner's knowledge, d_i denotes the item difficulty and σ is defined by:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

This formulation corresponds to the Rasch model.

After each interaction, ERS updates both the learner knowledge and the item difficulty as follows:

$$\theta_k \leftarrow \theta_k + \alpha\left(s_{k,i} - P(S_{k,i} = 1)\right) \tag{2}$$

$$d_i \leftarrow d_i + \beta\left(P(S_{k,i} = 1) - s_{k,i}\right) \tag{3}$$

where $s_{k,i} \in \{0, 1\}$ denotes the observed outcome, α is the knowledge update rate, and β is the difficulty update rate. Learner knowledge and item difficulty are initialized to zero. To avoid infinite divergence, the values of knowledge and difficulty are both limited to the interval $[-3, 3]$, similar to (Antal 2013).

For illustration, consider a learner (ID 74) submitting a protocol. The submission is evaluated by the system against an ordered set of constraints, each of which is treated as an item in our framework. The protocol is checked sequentially against these constraints. When evaluating item 51 ("*Distilled water is poured into the beaker using a measuring cylinder*"), the standard ERS models the interaction between learner 74 and item 51 as a match.

The outcome of this interaction determines how the model parameters are updated. Suppose that, prior to this interaction:

- the update rates are set to $\alpha = \beta = 0.4$,
- the estimated knowledge of learner 74 is $\theta_{74} = 1.5$, and
- the difficulty of item 51 is $d_{51} = 1.2$.

The predicted probability of success is therefore $P(S_{74,51}) = \sigma(\theta_{74} - d_{51}) \approx 0.57$.

Assume that the submitted protocol violates constraint 51, so that $s_{74,51} = 0$. The resulting updates are:

- the learner's knowledge decreases from 1.5 to $1.5 + 0.4 \times (0 - 0.57) \approx 1.27$
- the item's difficulty increases from 1.2 to $1.2 + 0.4 \times (0.57 - 0) \approx 1.43$

In this paper, we distinguish between parameters and hyper-parameters. We define parameters as quantities that are updated online during the learning process. In the standard ERS setting, these include the learner knowledge parameters $(\theta_k)_{k=1}^{K}$ where K denotes the number of learners, and the item difficulty parameters $(d_i)_{i=1}^{I}$ where I denotes the number of items. The model thus maintains $K + I$ parameters that evolve over time as new observations are incorporated. These parameters are not necessarily

expected to converge, particularly the knowledge parameters, since they represent the learner's current knowledge state and are expected to evolve as learning occurs.

In contrast, hyper-parameters are not updated during online learning but are determined through a separate estimation procedure, described in Section 5. In the standard ERS model, the hyper-parameters are α, the knowledge update rate, and β, the difficulty update rate. These hyper-parameters control the update dynamics and are shared across all learners and items.

4.2 The Enhanced ERS-based Model (EN)

In this section, we introduce three extensions to the standard ERS model. These extensions address common limitations related to:

- the representation of learner knowledge at the level of knowledge components,
- the decreasing uncertainty in item difficulty estimates as observations accumulate, and
- the estimation of prior knowledge for each learner.

Empirical results indicate that each extension contributes to improved predictive accuracy compared to the standard ERS.

Together, these extensions define the ENhanced model (EN), which serves as the baseline model, providing both the foundation and the point of comparison for the feedback-aware extensions introduced in the following sections.

4.2.1 Task Type–Specific Knowledge

The standard ERS formulation models each learner with a single global knowledge parameter (Pelánek 2016). While simple and efficient, this representation may obscure meaningful variations in performance across different types of tasks. In practice, learners often exhibit heterogeneous knowledge profiles: for instance, a learner may be proficient in selecting appropriate equipment for a titration protocol, yet struggle to choose the correct solution for rinsing that equipment. Moreover, representing learner knowledge with a single scalar value may limit interpretability from an instructional perspective.

To address this limitation, we extend the model to represent learner knowledge at the level of knowledge components, following the definition proposed by Kenneth Koedinger (Koedinger et al. 2012). In our setting, these knowledge components correspond to task types, as defined by domain experts.

For example, the items "*When using a graduated pipette to dispense liquid, its exact volume is dispensed*" and "*Distilled water is poured into the beaker using a measuring cylinder*" are related to the same task type "*Use of equipment*".

Formally, we associate each item i with a task type $t(i)$, and replace the global learner parameter with task-specific knowledge parameters $\theta_{k,t(i)}$. The probability of success is then defined as:

$$P\left(S_{k,i}=1\right)=\sigma\left(\theta_{k,t(i)}-d_i\right) \tag{EN-1}$$

and the corresponding update rule becomes:

$$\theta_{k,t(i)} \leftarrow \theta_{k,t(i)} + \alpha \left(s_{k,i} - P(S_{k,i} = 1)\right) \tag{2'}$$

It is worth noting that when multiple items associated with the same task type are addressed within a single submission, the corresponding knowledge parameter is updated only once, using the average of the individual knowledge updates. Consequently, multiple errors related to the same task type within a submission result in a single aggregated penalty (for example, two errors related to the same task type during the same submission will be penalized only once).

4.2.2 Difficulty and Knowledge Update Rates

Following (Pelánek 2016), we use a decreasing update rate for item difficulty to reflect decreasing uncertainty as more data is observed. Specifically, the difficulty update rate β of item i is scaled by an uncertainty function $n_i \mapsto \frac{1}{1+\gamma n_i}$ where n_i denotes the number of interactions involving i, and γ denotes the difficulty decay factor:

$$d_i \leftarrow d_i + \frac{\beta}{1+\gamma n_i} \left(P(S_{k,i} = 1) - s_{k,i}\right) \tag{EN-3}$$

Regarding the knowledge update rate, while Kandemir et al. address the issue of vanishing updates by introducing a lower-bound threshold to preserve learning effects (Kandemir et al. 2024), we instead adopt a constant update rate. This choice reflects the iterative nature of the task, in which learner knowledge may fluctuate over time rather than converge monotonically. More sophisticated approaches to modeling knowledge dynamics exist (Vermeiren et al. 2025), and could be incorporated in future work.

4.2.3 Prior Knowledge Estimation

At the beginning of an activity, when a learner has not yet interacted with the system, ERS faces the dual challenge of estimating the learner's initial knowledge (prior knowledge) while simultaneously tracking learning over time. Prior work has explored several approaches to this problem, including dedicated initialization models or estimation based on an initial set of responses (Pelánek 2016; Pelánek et al. 2017).

In this work, we follow the latter approach, introducing a novel and simple way of estimating prior knowledge: by increasing the update rate during the initial interactions between a learner and a given task type. Intuitively, early observations should have a stronger impact on the estimate of learner prior knowledge.

Formally, the knowledge update rule becomes:

$$\theta_{k,t(i)} \leftarrow \theta_{k,t(i)} + \delta(m_{k,t(i)}) \left(s_{k,i} - P(S_{k,i} = 1)\right) \tag{EN-2}$$

where $m_{k,t(i)}$ denotes the number of interactions involving the learner k and the task type $t(i)$ for each exercise, and $\delta(m_{k,t(i)})$ is defined as:

$$\delta(m_{k,t(i)}) = \begin{cases} \alpha \times \eta & \text{if } m_{k,t(i)} \leq \tau \\ \alpha & \text{otherwise} \end{cases} \tag{4}$$

Here, $\eta \geq 1$ is the prior knowledge boost factor and $\tau \geq 1$ determines the number of initial boosted interactions.

Finally, Equations (EN-1), (EN-2), and (EN-3) define the EN model.

4.3 Modeling Feedback

In this section, we introduce two extensions of the EN model that incorporate feedback. Both approaches rely on a binary distinction between attempts made without feedback and those made with feedback, which we formalize in the next subsection.

A central modeling question is how feedback, provided after an erroneous submission, alters the nature of the task faced by the learner. While attempts with and without feedback are treated as distinct conditions, different assumptions can be made regarding the role of feedback beyond this binary distinction.

One possible—yet extreme—assumption is that feedback fundamentally transforms the task, such that each pair (item, feedback level) can be treated as a distinct task. This perspective underlies the *New Item* (NI) model.

An alternative assumption is that, once feedback is present, it does not modify the intrinsic nature of the task but instead increases the likelihood of success in a graded manner depending on its level. This perspective motivates the *Probability Modifier* (PM) model.

These two models capture complementary assumptions about the role of feedback, resulting in different trade-offs between model expressiveness and complexity. While more complex formulations could be considered, we deliberately focus on these two simple and interpretable approaches.

A complete comparison of the models is provided in Table 1.

4.3.1 Binary Task Distinction

When integrating feedback into an ERS-based model, it is important to recognize that addressing an item without feedback differs fundamentally from addressing it with feedback, regardless of the level of assistance. More specifically, solving a task independently is qualitatively different from revising a solution based on targeted feedback.

This distinction is particularly salient in our setting. Learners are not explicitly aware of the individual items they address, as they interact with the system by submitting complete protocols. However, when feedback is provided following an erroneous submission, it highlights a specific error and directs the learner's attention to the corresponding item. As a result, the subsequent attempt is focused on resolving that particular issue.

This observation motivates a key modeling decision: extending the item space along a binary dimension that distinguishes between attempts made without feedback and those made with feedback. Both item versions remain associated with the same underlying task type.

An important implication concerns difficulty estimation. For items addressed without feedback, difficulty is estimated from the full population of learners. In contrast,

items addressed with feedback are observed only for the subset of learners who previously failed the item. This induces a selection bias, as these observations are conditioned on prior failure and therefore overrepresent lower-proficiency learners. Consequently, the estimated difficulty of items with feedback may be inflated relative to their counterparts without feedback.

4.3.2 New Item Model (NI)

We now introduce our first feedback-aware extension of the baseline model, referred to as the New Item (NI) model. This approach builds on the binary task distinction described in Section 4.3.1 and generalizes it to account for multiple levels of feedback. The central idea is to treat each item as a set of distinct items indexed by feedback level.

More precisely, each original item i is expanded into a set of items (i, l), where $l \in [\![0, L]\!]$ denotes the level of feedback provided. Here, $l = 0$ corresponds to attempts made without feedback, while higher values of l represent increasing levels of assistance. Consequently, the item space increases from size I to $(L + 1) \times I$.

For example, consider the item "*The pipette should be rinsed before use.*" In the NI model, this item is decomposed into multiple variants: one corresponding to attempts without feedback, and others corresponding to attempts supported by feedback at different levels of specificity, ranging from general hints to explicit corrective instructions.

Formally, the probability of success is defined as:

$$P\left(S_{k,i,l} = 1\right) = \sigma\left(\theta_{k,t(i)} - d_{i,l}\right) \tag{NI-1}$$

where $d_{i,l}$ denotes the difficulty of item i at feedback level l.

The knowledge update rule is given by:

$$\theta_{k,t(i)} \leftarrow \theta_{k,t(i)} + \delta(m_{k,t(i)})\left(s_{k,i,l} - P(S_{k,i,l} = 1)\right) \tag{NI-2}$$

The difficulty update rule becomes:

$$d_{i,l} \leftarrow d_{i,l} + \frac{\beta}{1 + \gamma n_{i,l}}\left(P(S_{k,i,l} = 1) - s_{k,i,l}\right) \tag{NI-3}$$

where $n_{i,l}$ denotes the number of interactions involving item i at feedback level l.

This formulation preserves the structure of the baseline model while extending the item space to explicitly account for feedback. As a result, the NI model captures the idea that attempts made under different feedback conditions correspond to distinct tasks, while still linking them through a shared underlying knowledge component.

4.3.3 Probability Modifier Model (PM)

We then introduce a second extension that further incorporates feedback-awareness, referred to as the Probability Modifier (PM) model. In contrast to the NI model, which expands the item space across all feedback levels, the PM model adopts a

more parsimonious representation. It distinguishes between attempts made with and without feedback, while modeling the graded effect of feedback levels as a direct shift in the probability of success.

To this end, we introduce a binary mapping $f(l)$ defined as: $f(l) = \begin{cases} 0 & \text{if } l = 0 \\ 1 & \text{if } l > 0 \end{cases}$

where $l \in [\![0, L]\!]$ denotes the feedback level. This mapping allows us to distinguish between unassisted ($f(l) = 0$) and assisted ($f(l) = 1$) attempts, resulting in an item space of size $2 \times I$.

At the same time, the PM model assumes that feedback directly increases the probability of success. This effect is captured through a level-dependent shift in the latent success function. The probability of success is thus defined as:

$$P\left(S_{k,i,l} = 1\right) = \sigma\left(\theta_{k,t(i)} - d_{i,f(l)} + c_l\right) \tag{PM-1}$$

where $d_{i,f(l)}$ denotes the difficulty of item i under feedback condition $f(l)$, and c_l is a feedback-specific offset hyper-parameter satisfying $c_0 = 0$. This formulation ensures that the model reduces to the baseline case in the absence of feedback, while allowing increasing levels of feedback to progressively shift the probability of success.

The shift of the Rasch model is illustrated in Figure 5.

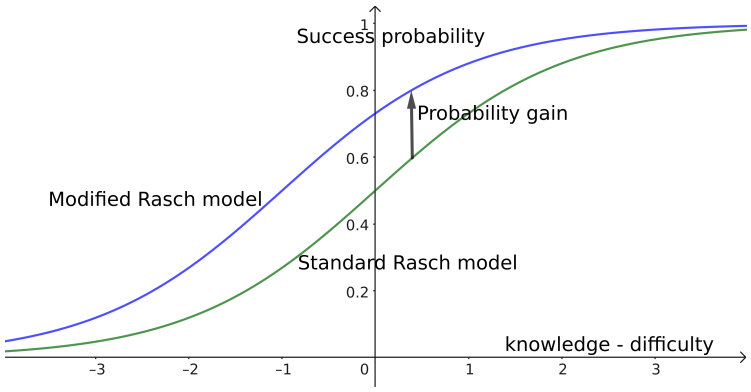


Fig. 5 Effect of feedback on the Rasch model

The knowledge update rule is given by:

$$\theta_{k,t(i)} \leftarrow \theta_{k,t(i)} + \delta(m_{k,t(i)})\big(s_{k,i,l} - P(S_{k,i,l} = 1)\big) \tag{PM-2}$$

and the difficulty update rule becomes:

$$d_{i,f(l)} \leftarrow d_{i,f(l)} + \frac{\beta}{1 + \gamma n_{i,f(l)}}\big(P(S_{k,i,l} = 1) - s_{k,i,l}\big) \tag{PM-3}$$

where $n_{i,f(l)}$ denotes the number of interactions associated with item i under feedback condition $f(l)$.

This formulation captures feedback in two complementary ways: (i) through a binary distinction between assisted and unassisted attempts at the item level, and (ii) through a graded, level-dependent effect on the probability of success. Compared to the NI model, this approach provides a more compact representation of feedback, resulting in a trade-off between expressiveness and model complexity.

4.3.4 Idea of Partial Credit

We considered adapting the notion of partial credit, as used in some Bayesian Knowledge Tracing (BKT)-based models (Wang and Heffernan 2013). The underlying intuition is that a learner who succeeds after receiving feedback likely demonstrates a lower level of mastery than a learner who succeeds independently. Accordingly, successful outcomes obtained with feedback could be discounted to reflect this difference.

While incorporating such a mechanism into ERS would be straightforward—for instance, by introducing a discount factor in the knowledge update equation—this approach raises an important limitation. In particular, it does not naturally account for the presence and level of feedback at prediction time, as the discount is applied only during model updates. As a result, the model would remain unable to differentiate between assisted and unassisted attempts when estimating the probability of success.

To preserve a simple and coherent modeling framework, we therefore leave this approach for future work.

5 Methods

5.1 Model Performance Evaluation

Evaluating learner models is inherently challenging. A commonly adopted—though not exhaustive—approach is to assess predictive performance on the next learner response (Pelánek 2025). Because our data are sequential and the objective is to model learning over time, preserving the temporal order of interactions for each learner is essential.

For each submission (identified by an ID), we predict the outcome of each learner-item interaction using the Rasch formulation (Equations EN-1, NI-1 and PM-1), while iteratively updating learner knowledge and item difficulty based on all preceding interactions.

Applying this process to all submissions yields a vector of predicted success probabilities in the interval $[0, 1]$, with the same length as the dataset. Model quality is then assessed by comparing these predictions with the observed outcomes using standard evaluation metrics. Following prior work (Pelánek 2015), we report Root Mean Square Error (RMSE), and Area Under the ROC Curve (AUC). Although log loss (negative log-likelihood) is widely used, we do not report it in this work because learner interactions are sequentially dependent. Consequently, the standard summed log loss cannot be interpreted as the likelihood of a coherent joint probabilistic model without explicitly modeling these dependencies.

Table 1 Unified comparison of EN, NI, and PM models

Notations	$k \in [\![1, K]\!]$: learner index	θ: knowledge	α: knowledge update rate
	$i \in [\![1, I]\!]$: item index	d: difficulty	β: difficulty update rate
	$t(i) \in [\![1, T]\!]$: task type index	n: number of interactions	γ: difficulty decay factor
	$l \in [\![0, L]\!]$: feedback level	m: number of interactions	η: prior knowledge boost factor
	$(0 =$ no feedback$)$	s: success outcome	τ: number of boosted interactions
	$f(l) \in \{0, 1\}$: feedback or not	$P(S = 1)$: success probability	c_l: probability shift

	EN ENhanced standard	**NI** New Item	**PM** Probability Modifier
Item representation			
Item index	i	(i, l)	$(i, f(l))$
Feedback modeling	Ignored	Full expansion over levels	Binary split + probability shift
Parameters			
Knowledge	$\theta_{k,t(i)}$ $T \times K$ parameters	$\theta_{k,t(i)}$ $T \times K$ parameters	$\theta_{k,t(i)}$ $T \times K$ parameters
Difficulty	d_i I parameters	$d_{i,l}$ $(L+1) \times I$ parameters	$d_{i,f(l)}$ $2I$ parameters
Hyper-parameters	α, β, γ, η, τ 5 parameters	α, β, γ, η, τ 5 parameters	α, β, γ, η, τ, c_l $5 + L$ parameters
Prediction	Common formulation for all models: $\sigma : x \mapsto \frac{1}{1+e^{-x}}$		
Probability of success	$\sigma(\theta_{k,t(i)} - d_i)$	$\sigma(\theta_{k,t(i)} - d_{i,l})$	$\sigma(\theta_{k,t(i)} - d_{i,f(l)} + c_l)$
Knowledge update	Common formulation for all models: $\theta \leftarrow \theta + \delta(m)\big(s - P(S = 1)\big)$ where $\delta(m) = \begin{cases} \alpha \times \eta & \text{if } m \leq \tau \\ \alpha & \text{otherwise} \end{cases}$		
Indexes details	$\theta_{k,t(i)}, m_{k,t(i)}, s_{k,i}, S_{k,i}$	$\theta_{k,t(i)}, m_{k,t(i)}, s_{k,i,l}, S_{k,i,l}$	$\theta_{k,t(i)}, m_{k,t(i)}, s_{k,i,l}, S_{k,i,l}$
Difficulty update	Common formulation for all models: $d \leftarrow d + \frac{\beta}{1+\gamma n}\big(P(S = 1) - s\big)$		
Indexes details	$d_i, n_i, S_{k,i}, s_{k,i}$	$d_{i,l}, n_{i,l}, S_{k,i,l}, s_{k,i,l}$	$d_{i,f(l)}, n_{i,f(l)}, S_{k,i,l}, s_{k,i,l}$
Model interpretation	No feedback effect	Feedback level changes the task	Feedback level modifies success probability

5.2 Hyper-parameter Fitting

All models considered in this study depend on a small number of hyper-parameters: five for the Enhanced standard ERS and the New Item model, and ten for the Probability Modifier model. As in prior studies (Pelánek 2016; Pelánek et al. 2017; Kandemir et al. 2024), hyper-parameters are optimized using grid search. This choice is motivated by the relative stability of ERS-based models, which tend to be weakly sensitive to moderate hyper-parameter variations.

Moreover, our goal is to compare models under a consistent optimization procedure rather than to identify a globally optimal hyper-parameter configuration. Hyper-parameter selection is therefore driven by minimizing RMSE on the dataset.

5.3 Model Comparison

We evaluate whether the proposed feedback-aware models (NI and PM) improve upon the baseline EN model. This comparison is non-trivial, as differences in predictive performance may be small and dependent on the evaluation metric considered (Pelánek 2025). Although evaluation metrics computed on the full dataset provide an initial indication of model quality, isolated performance estimates are insufficient to assess the statistical stability and practical relevance of observed differences.

To address this issue, we adopt three complementary evaluation strategies when comparing two models:

- *Robustness to learner variation*, which evaluates whether observed performance differences remain stable under perturbations of the learner population;
- *Consistency across learners*, which assesses whether one model consistently outperforms another across individual learners rather than only for a limited subset;
- *Generalization to unseen learners*, which evaluates the ability of a model to generalize to previously unseen learner trajectories.

In the following evaluation procedures, the term *training* refers to the estimation of model parameters from interaction data (i.e., learner knowledge and/or item difficulty parameters). The term *testing* refers to the generation of success predictions and the computation of evaluation metrics on the corresponding interaction set. All hyper-parameters remain fixed to the values obtained using the procedure described in Section 5.2.

5.3.1 Robustness Test

This test addresses the following question: *Is the observed superiority of one model over another robust to variations in the learner population?*

To answer this question, we employ a learner-level bootstrap procedure. Specifically, we generate 100 bootstrap replicates by sampling 296 learner trajectories with replacement, where a trajectory corresponds to the complete ordered sequence of interactions associated with a learner. When a learner is sampled multiple times, duplicated trajectories are treated as independent synthetic learners. The chronological order within each trajectory and the integrity of submission blocks are preserved.

This procedure generates perturbed versions of the original dataset with the same number of learners. These datasets simulate variations in the learner population while maintaining the sequential structure of the interaction data. Importantly, this evaluation is not intended to estimate out-of-sample predictive performance, but rather to assess the stability of comparative model differences under learner-population perturbations.

For each bootstrap replicate, both models are trained and evaluated on the generated dataset using RMSE. The paired performance difference between the models is then recorded. Repeating this process across bootstrap samples yields an empirical distribution of performance differences, from which percentile confidence intervals are estimated (Table 2). If zero does not belong to the confidence interval, one model is considered to outperform the other according to the robustness test.

5.3.2 Consistency Test

This test addresses the question: *Does one model consistently outperform the other across learners?*

Model comparison is initially performed at the interaction level, using predictions associated with each (learner, item) pair. However, because learners interact with the system a different number of times, aggregate RMSE values may be disproportionately influenced by highly active learners. The present test therefore evaluates whether one model systematically outperforms the other at the learner level.

Both models are first trained independently on the full dataset. For each learner, an individual RMSE score is then computed using all interactions associated with that learner, yielding paired learner-level performance measurements.

The paired learner-level RMSE differences are analyzed using a one-sided Wilcoxon signed-rank test (Table 2). If the corresponding p-value is below 0.05, one model is considered to outperform the other according to the consistency test.

5.3.3 Generalization Test

This test addresses the question: *Can the model generalize to previously unseen learner trajectories?*

At each iteration, a subset of 148 learner trajectories is randomly selected and removed from the dataset, corresponding to half of the learners in the original dataset. The remaining trajectories form the training set, while the held-out learner trajectories constitute the test set. The chronological order within each learner trajectory and the integrity of submission blocks are preserved.

Both models are trained on the reduced training dataset, during which only item difficulties are estimated. The models are then evaluated on the held-out learner trajectories by estimating learner knowledge online and computing RMSE over the associated interactions.

Repeating this process over 100 random learner splits yields an empirical distribution of paired performance differences, from which percentile confidence intervals are estimated (Table 2). Similarly to the robustness test, if zero does not belong to the confidence interval, one model is considered to outperform the other according to the generalization test.

Table 2 Algorithms used to compare the models

Algorithm 1 Robustness test

Require: Dataset D, models M_1, M_2, number of runs $N = 100$

for $i = 1$ to N **do**
  Sample 296 learners from D with replacement
  Construct bootstrap dataset D_i:
    - interleave learners trajectories
    - preserve chronological order within trajectories
  Train M_1 on D_i: estimate knowledge and difficulties
  Evaluate M_1 on D_i: obtain $RMSE_{1,i}$
  Train M_2 on D_i: estimate knowledge and difficulties
  Evaluate M_2 on D_i: obtain $RMSE_{2,i}$
  Store $\Delta_i = RMSE_{1,i} - RMSE_{2,i}$
end for
Estimate 95% percentile confidence interval CI from $\{\Delta_1, \ldots, \Delta_N\}$

Algorithm 2 Consistency test

Require: Dataset D, models M_1, M_2

Train M_1 on D: estimate knowledge and difficulties
Evaluate M_1 on D: obtain success predictions $\hat{y}_1$
Train M_2 on D: estimate knowledge and difficulties
Evaluate M_2 on D: obtain success predictions $\hat{y}_2$
for each learner u **do**
  Compute $RMSE_1(u)$ using $\hat{y}_1$
  Compute $RMSE_2(u)$ using $\hat{y}_2$
  Store $\Delta(u) = RMSE_1(u) - RMSE_2(u)$
end for
Apply Wilcoxon test on paired RMSE values and report p-value

Algorithm 3 Generalization test

Require: Dataset D, models M_1, M_2, number of runs $N = 100$

for $i = 1$ to N **do**
  Construct dataset D_i^{train}:
    - Randomly select learner subset U_i (half of the learners)
    - D_i^{train} is a copy of D where U_i trajectories have been removed
  Construct dataset D_i^{test}:
    - interleave learners trajectories from U_i
    - preserve chronological order within trajectories
  Train M_1 on D_i^{train}: estimate difficulties
  Evaluate M_1 on D_i^{test}: obtain $RMSE_{1,i}$
  Train M_2 on D_i^{train}: estimate difficulties
  Evaluate M_2 on D_i^{test}: obtain $RMSE_{2,i}$
  Store $\Delta_i = RMSE_{1,i} - RMSE_{2,i}$
end for
Estimate 95% percentile confidence interval CI from $\{\Delta_1, \ldots, \Delta_N\}$

6 Results and Discussion

This section presents the empirical results and summarizes the main findings. Limitations are discussed separately in Section 7. We first introduce the hyper-parameters used throughout the study, then analyze the comparison between models, and finally provide a deeper discussion of the models through the relation between hyper-parameters and parameter estimation dynamics.

6.1 Hyper-Parameters

As described in Section 5.2, hyper-parameters for each model were optimized on the full dataset by minimizing the root mean squared error (RMSE) using a grid-search procedure. Table 3 presents the resulting optimal values; the area under the ROC curve (AUC) is also reported for reference. For comparison purposes, we additionally include results for the standard ERS model (ST), which relies on a single global knowledge parameter per learner, does not incorporate a decay factor for item difficulty, and does not model prior knowledge.

Table 3 Model performance

Model	Hyper-parameters	RMSE	AUC
ST	knowledge update rate: $\alpha = 0.02$ difficulty update rate: $\beta = 0.4$	0.420	0.736
EN	knowledge update rate: $\alpha = 0.2$ difficulty update rate: $\beta = 0.5$ difficulty decay factor: $\gamma = 0.002$ prior knowledge boost factor: $\eta = 4$ number of boosted interactions: $\tau = 1$	0.415	0.752
NI	knowledge update rate: $\alpha = 0.09$ difficulty update rate: $\beta = 1.3$ difficulty decay factor: $\gamma = 0.15$ prior knowledge boost factor: $\eta = 6$ number of boosted interactions: $\tau = 1$	0.393	0.790
PM	knowledge update rate: $\alpha = 0.1$ difficulty update rate: $\beta = 1.5$ difficulty decay factor: $\gamma = 0.2$ prior knowledge boost factor: $\eta = 5$ number of boosted interactions: $\tau = 1$ probability shift for level 1: $c_1 = -1.2$ probability shift for level 2: $c_2 = -1.5$ probability shift for level 3: $c_3 = -1.3$ probability shift for level 4: $c_4 = 0.0$ probability shift for level 5: $c_5 = -0.1$	0.393	0.790

The hyper-parameters remain fixed to these values throughout all model comparison tests reported in Table 4.

6.2 Model Comparison

Table 4 Comparison of model performance

Models M1 Vs M2	Robustness to learner sampling (95% percentile CI)	Consistency across learners (Wilcoxon test p-value)	Generalization to unseen learners (95% percentile CI)
ST Vs. EN	$[0.008, 0.010]$	4.1×10^{-12}	$[0.008, 0.011]$
$mean(RMSE_{ST})$	0.421	0.428	0.423
$mean(RMSE_{EN})$	0.413	0.424	0.413
$mean(RMSE)$ gain	-2.1%	-1.0%	-2.3%
EN Vs. NI	$[0.020, 0.025]$	1.6×10^{-20}	$[0.019, 0.024]$
$mean(RMSE_{EN})$	0.413	0.424	0.413
$mean(RMSE_{NI})$	0.390	0.410	0.392
$mean(RMSE)$ gain	-5.5%	-3.1%	-5.2%
EN Vs. PM	$[0.019, 0.024]$	5.4×10^{-20}	$[0.019, 0.023]$
$mean(RMSE_{EN})$	0.413	0.424	0.413
$mean(RMSE_{PM})$	0.391	0.411	0.392
$mean(RMSE)$ gain	-5.2%	-3.0%	-5.1%
Decision Criteria for: "M2 outperforms M1"	lower bound of CI is greater than 0	p-value < 0.05	lower bound of CI is greater than 0

First, the EN model significantly outperforms the standard ERS formulation (ST) across all three evaluation procedures. The robustness and generalization tests yield strictly positive confidence intervals, while the consistency analysis indicates that the improvement is observed across learners.

More importantly, both feedback-aware extensions (NI and PM) substantially outperform EN. For both models, confidence intervals remain strictly positive in the robustness and generalization tests, indicating stable improvements under learner-population perturbations as well as good generalization to unseen learners. In addition, the extremely small Wilcoxon p-values indicate that these improvements are consistently observed across learners.

Finally, NI and PM produce highly similar results despite relying on different assumptions regarding the role of feedback.

6.3 Discussion of the Optimal Models

Table 3 reports the optimized hyper-parameter values associated with each model, while the comparison tests confirm that the observed performance differences are reliable.

The main result is that the feedback-aware models (NI and PM) achieve lower RMSE values than the baseline EN model. In other words, both NI and PM provide greater predictive accuracy than EN, and substantially outperform the ST model.

Beyond predictive performance, however, the estimated hyper-parameters also provide insight into the learning dynamics captured by each model. Because these

hyper-parameters directly govern the evolution of knowledge and difficulty estimates, it is informative to relate them to the trajectories of the corresponding parameters. Figure 6 illustrates the evolution of knowledge and difficulty parameters as a function of the number of updates.

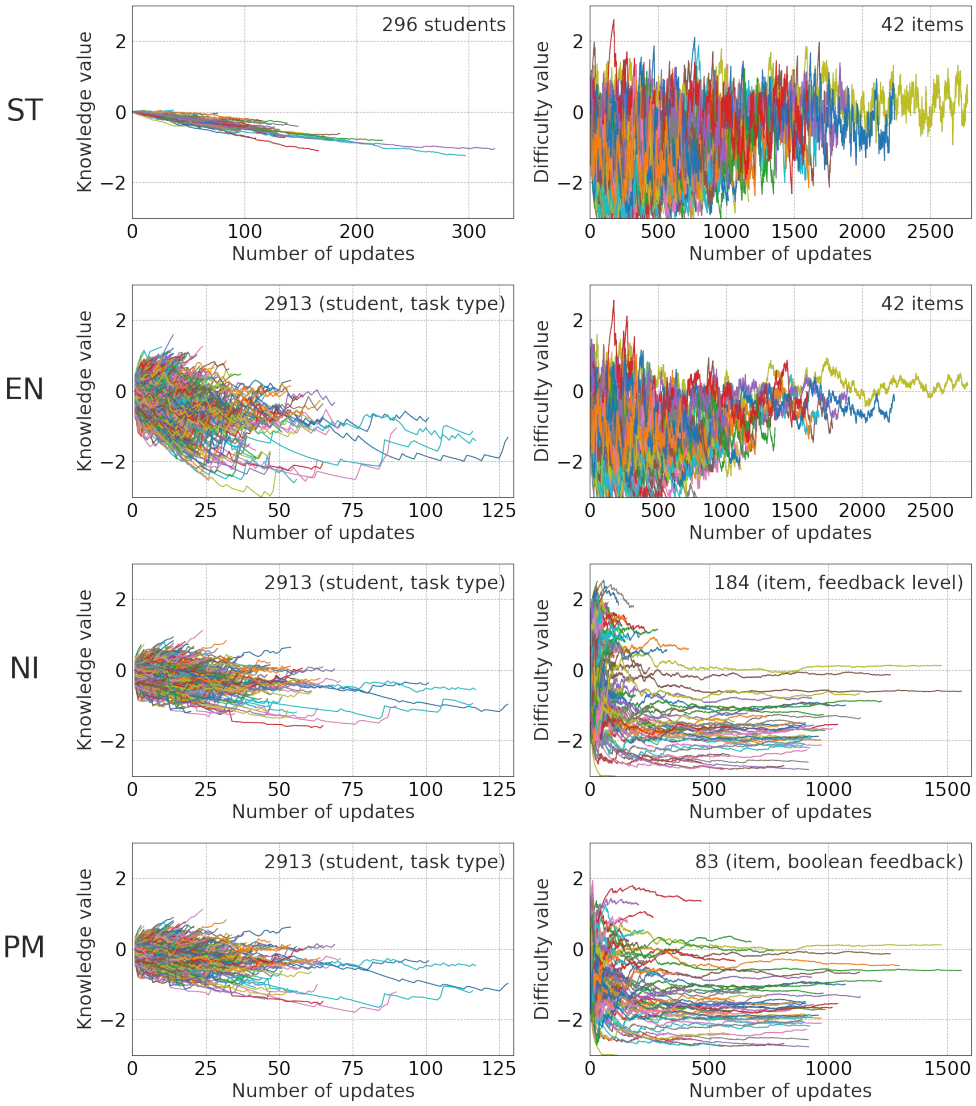


Fig. 6 Knowledge and difficulty parameter history

We first observe that, in the ST model, the knowledge update rate is substantially lower than the difficulty update rate. This suggests that, when knowledge is represented as a single global parameter per learner, predictive performance is improved when learner knowledge is treated as relatively stable over time, with most variation absorbed by item difficulty. This behavior raises two limitations. First, the model provides limited discrimination between learners, restricting its usefulness for adaptive applications such as personalized feedback or task recommendation. Second, item difficulty becomes highly dependent on the observed learner population, potentially reducing its stability and transferability across cohorts.

The EN model partially addresses these limitations. The knowledge and difficulty update rates are more balanced, as illustrated in Figure 6. Moreover, representing knowledge at the level of task types increases discrimination between learners while reducing the number of updates per parameter. The introduction of a boosting factor enables faster initialization of knowledge estimates, while the decay factor reduces fluctuations in item difficulty compared to the ST model. Nevertheless, difficulty estimates still fail to converge. A plausible explanation is that ignoring feedback prevents stabilization, as attempts associated with different levels of assistance are grouped under the same item.

The NI model extends the EN approach by expanding the item space along the feedback dimension. This increases the number of difficulty parameters and consequently reduces the number of observations per parameter. Despite this, difficulty estimates exhibit substantially more stable convergence, supported by a considerably larger decay factor than in the EN model. At the same time, the model maintains strong discrimination between learners, facilitated by effective prior knowledge estimation through a relatively high boosting factor, even with a moderate knowledge update rate. The improved convergence of difficulty estimates may be explained by the fact that the finer-grained representation defined by (item, feedback level) pairs is more homogeneous across learners in terms of difficulty.

The PM model exhibits dynamics similar to those of the NI model while relying on a more compact parameterization of item difficulty. In this case, the number of difficulty parameters is approximately doubled relative to the original item set. Compared with the NI model, this reduction in model complexity comes at the cost of introducing additional hyper-parameters that capture the effect of feedback on the probability of success. These parameters can be interpreted as shifts in item difficulty under different feedback conditions. In the present dataset, two main regimes emerge: lower feedback levels (1 to 3) are associated with lower success probabilities compared with higher levels (4 and 5), which provide more explicit guidance. The comparable performance of the NI and PM models suggests that the effects of different feedback levels are relatively consistent across items.

6.4 RMSE Evolution and Per-learner Analysis

Beyond reporting aggregate RMSE values, we examine how prediction error evolves over submissions. Figure 7 shows the progression of RMSE as a function of the number of predictions.

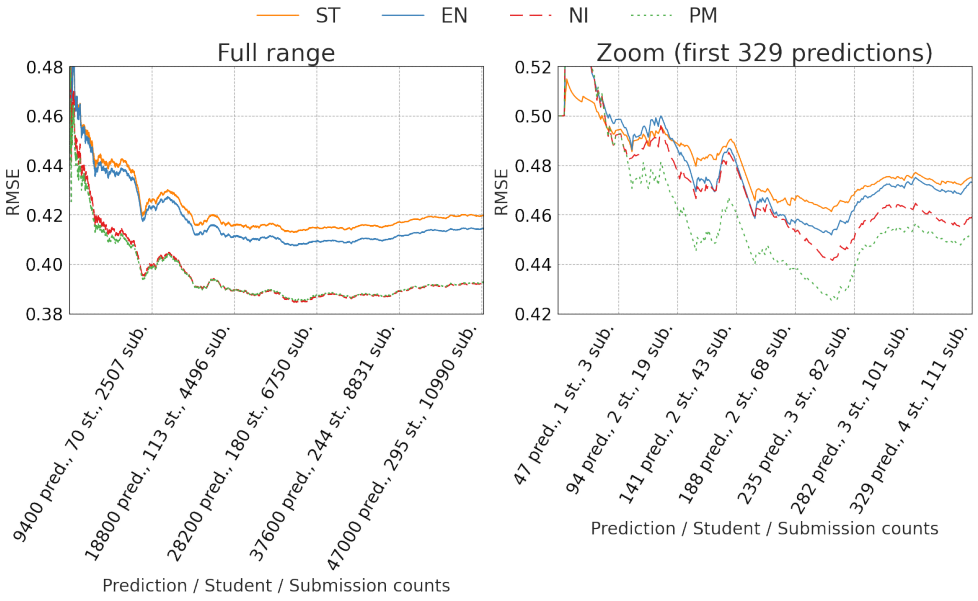


Fig. 7 Evolution of RMSE over the entire dataset. For example, after the first 9,400 predictions, RMSE is slightly above 0.42 for the ST and EN models, and close to 0.40 for the NI and PM models. At this point, 70 learners have interacted with the system and submitted 2,507 protocols.

All models exhibit a similar overall pattern: RMSE decreases rapidly during the initial phase, reaches a plateau after approximately 20,000 predictions (corresponding to about 115 learners), and then shows a slight increase toward the end of the sequence. From approximately 10,000 predictions onward (around 70 learners), the curves evolve in a largely parallel manner.

For reference, a random model producing Bernoulli predictions centered at 0.5 would yield an RMSE of 0.5. The substantially lower RMSE values observed here indicate that all models capture meaningful structure in the data, learning informative representations of both learner knowledge and item difficulty. The stabilization of RMSE after approximately 115 learners is consistent with prior findings (Pelánek 2016).

Differences between models are most pronounced during the early stages. Both the Probability Modifier (PM) and New Item (NI) models achieve lower RMSE values than the Enhanced model (EN) after only a few hundred predictions. This suggests that the feedback-aware models require relatively little data to outperform the baseline. Beyond this initial phase, the RMSE curves remain approximately parallel, indicating that subsequent variations are largely driven by the incoming data rather than by intrinsic differences between models.

After the plateau phase, a slight increase in RMSE is observed for all models. This may reflect changes in the composition of the learner population over time, resulting in a mild distribution shift.

To further examine local adaptation effects, we compute RMSE at the level of individual learners. Figure 8 presents the per-learner RMSE differences relative to the EN model, with each point corresponding to a single learner. First, the large majority of points lie below zero on the y-axis, indicating that most learners benefit from the feedback-aware models and further supporting the consistency analysis presented earlier. Second, both NI and PM tend to yield larger improvements for learners with a greater number of interactions. This pattern suggests that the benefits of feedback-aware modeling become more pronounced as more observations are available for a given learner.

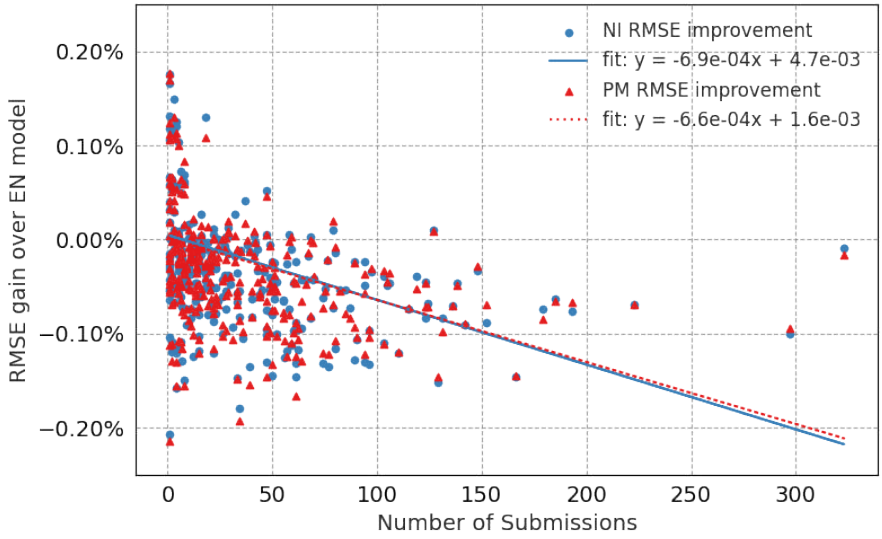


Fig. 8 Per learner RMSE improvement compared to the EN model

7 Limitations and Future Work

This study demonstrates that incorporating feedback into the Elo Rating System can improve predictive performance in iterative learning environments. However, several limitations should be acknowledged.

First, the analysis relies on a single dataset, which limits the generalizability of the findings. Although the dataset is representative of the targeted learning scenario, future work should evaluate the proposed models on additional datasets and domains to assess their robustness across contexts.

Second, the hyper-parameter optimization procedure is relatively simple and may not yield optimal configurations. In this work, a grid search was deemed sufficient for comparative purposes. However, more systematic approaches—such as combining grid search or more advanced optimization techniques with cross-validation—could lead to more reliable estimates and improve model performance. An interesting direction

for future work is the online adaptation of hyper-parameters to account for non-stationary learning environments, such as changes in exercises. While such approaches may improve responsiveness, they also raise challenges related to stability and model identifiability.

Third, the choice of evaluation metric plays a critical role in both model tuning and comparison. In this study, we rely on standard metrics (RMSE and AUC) to ensure comparability with prior work. However, alternative evaluation frameworks could provide additional insights. In particular, likelihood-based metrics have been shown to offer desirable properties for model comparison (Pelánek 2015). Their use would require constructing approximately independent samples, for instance by aggregating at the learner level rather than at the level of individual predictions.

Finally, the proposed enhanced ERS model addresses several known limitations of the standard formulation, but does so in a simplified manner. In particular, the mapping between items and knowledge components is assumed to be many-to-one, which may not hold in more complex settings. More expressive formulations—such as multivariate extensions (Abdi et al. 2019) or (Kandemir et al. 2024)—could provide a more accurate representation of learning processes. Similarly, the proposed approach to modeling prior knowledge remains heuristic. Alternative strategies, including dedicated initialization models (Pelánek et al. 2017), could be explored. More generally, the use of adaptive update rates, as proposed in (Vermeiren et al. 2025), represents a promising direction for improving the flexibility of ERS-based models. Integrating such refinements with feedback-aware modeling constitutes an important direction for future research.

8 Conclusion

This study investigated how automated feedback can be incorporated into Elo Rating System–based models in the context of complex, iterative task solving. Our primary objective was to assess whether explicitly accounting for feedback improves predictive performance while preserving key properties required for adaptive learning systems, namely meaningful discrimination between learners and stable estimation of item difficulty.

To this end, we proposed two feedback-aware extensions of ERS that embody complementary modeling assumptions. Both models distinguish between attempts made with and without feedback, reflecting the different conditions under which learners operate. The New Item (NI) model then treats each combination of task and feedback level as a distinct entity, capturing the idea that feedback fundamentally alters the nature of the task. In contrast, the Probability Modifier (PM) model adopts a more parsimonious formulation. Once feedback is present, it directly shifts the probability of success without redefining the task across feedback levels. This simplification, however, requires feedback levels to have a consistent effect across items.

Empirical results on a real-world dataset show that both approaches consistently improve predictive accuracy compared to feedback-agnostic ERS models. Importantly, these gains are achieved without degrading the interpretability of the model: both

NI and PM preserve the ability to differentiate learner knowledge profiles and produce stable item difficulty estimates, which are essential for practical deployment in adaptive systems.

Beyond predictive performance, our findings highlight the importance of explicitly modeling feedback in iterative learning environments. Ignoring feedback may lead to biased estimates of learner knowledge, particularly when assistance plays a central role in task completion.

Although the study is grounded in a specific experimental setting, the proposed formulations are general and can be applied to a broad class of learning environments characterized by (i) complex tasks, (ii) iterative solution processes, and (iii) structured feedback. This generality makes them particularly relevant for modern digital learning platforms, where scalability and real-time adaptation are critical constraints.

Overall, this work suggests that incorporating feedback into simple, interpretable models can yield substantial benefits, offering a compelling alternative to increasingly complex modeling approaches. In doing so, it contributes to ongoing discussions on the role of lightweight methods in educational data mining and adaptive learning, and opens avenues for further research on integrating richer forms of instructional support into efficient learner modeling frameworks.

Declarations

Funding

Sihem Amer-Yahia received funding from ARMADA, by the European Union's Horizon Europe Marie Skłodowska-Curie Actions (MSCA), under grant agreement No. 101168951.

Ethics Approval and Consent to Participate

All students involved in this research provided informed consent prior to participation in the experiment. The dataset was anonymised.

Data and Code Availability

The data used in this study and the code for the proposed models are publicly available.

Author Contributions

S.R. designed the core models, conducted the initial analyses, and drafted the manuscript. C.dH. contributed to the design of the models and to the analysis and interpretation of the results. S.A.-Y. contributed to strengthening the statistical methodology used for model comparison. C.dH. and S.A.-Y. critically reviewed and revised the manuscript.

Credits

The icons used in Figure 1 are free for personal and commercial use. They were obtained from Flaticon.com and created by Freepik.

References

Asselman, A., Khaldi, M., Aammou, S.: Evaluating the impact of prior required scaffolding items on the improvement of student performance prediction. Education and Information Technologies **25**(4), 3227–3249 (2020) https://doi.org/10.1007/s10639-019-10077-3

Abdi, S., Khosravi, H., Sadiq, S., Gasevic, D.: A multivariate elo-based learner model for adaptive educational systems. In: Desmarais, M.C., Lynch, C.F., Merceron, A., Nkambou, R. (eds.) Proceedings of the 12th International Conference on Educational Data Mining, EDM 2019, Montréal, Canada, July 2-5, 2019. International Educational Data Mining Society (IEDMS), ??? (2019)

Antal, M.: On the use of elo rating system for adaptive assessment. Studia Universitatis Babeş-Bolyai, Informatica **58**(1), 29 (2013)

Abdelrahman, G., Wang, Q., Nunes, B.: Knowledge tracing: A survey. ACM Comput. Surv. **55**(11), 224–122437 (2023) https://doi.org/10.1145/3569576

Bolsinova, M., Gergely, B., Brinkhuis, M.J.S.: Keeping elo alive: Evaluating and improving measurement properties of learning systems based on elo ratings. British Journal of Mathematical and Statistical Psychology **n/a** (2025) https://doi.org/10.1111/bmsp.12395

Feng, M., Heffernan, N., Koedinger, K.: Addressing the assessment challenge with an online system that tutors as it assesses. User Modeling and User-Adapted Interaction **19**(3), 243–266 (2009) https://doi.org/10.1007/s11257-009-9063-7

Glickman, M.E., Jones, A.C.: Rating the chess rating system. Chance **12**, 21–28 (1999)

Hattie, J., Timperley, H.: The Power of Feedback. Review of Educational Research **77**(1), 81–112 (2007) https://doi.org/10.3102/003465430298487 . Accessed 2024-12-04

Koedinger, K.R., Corbett, A.T., Perfetti, C.: The Knowledge-Learning-Instruction Framework: Bridging the Science-Practice Chasm to Enhance Robust Student Learning. Cognitive Science **36**(5), 757–798 (2012) https://doi.org/10.1111/j.1551-6709.2012.01245.x

Kandemir, E.N., Vie, J.-J., Sanchez-Ayte, A., Palombi, O., Ramus, F.: Adaptation of the multi-concept multivariate elo rating system to medical students' training data. In: Proceedings of the 14th Learning Analytics and Knowledge Conference.

LAK '24, pp. 123–133. Association for Computing Machinery, New York, NY, USA (2024). https://doi.org/10.1145/3636555.3636858

Mangaroska, K., Vesin, B., Giannakos, M.: Elo-rating method: Towards adaptive assessment in e-learning. In: 2019 IEEE 19th International Conference on Advanced Learning Technologies (ICALT), vol. 2161-377X, pp. 380–382 (2019). https://doi.org/10.1109/ICALT.2019.00116

Ostrow, K., Donnelly, C., Adjei, S., Heffernan, N.: Improving student modeling through partial credit and problem difficulty. In: Proceedings of the Second (2015) ACM Conference on Learning @ Scale. L@S '15, pp. 11–20. Association for Computing Machinery, New York, NY, USA (2015). https://doi.org/10.1145/2724660.2724667

Ohlsson, S.: Constraint-based student modeling. In: Greer, J.E., McCalla, G.I. (eds.) Student Modelling: The Key to Individualized Knowledge-Based Instruction, pp. 167–189. Springer, Berlin, Heidelberg (1994). https://doi.org/10.1007/978-3-662-03037-0_7

Pelánek, R.: Metrics for evaluation of student models. Journal of Educational Data Mining **7**(2), 1–19 (2015) https://doi.org/10.5281/zenodo.3554666

Pelánek, R.: Applications of the elo rating system in adaptive educational systems. Computers & Education **98**, 169–179 (2016) https://doi.org/10.1016/j.compedu.2016.03.017

Pelánek, R.: Adaptive learning is hard: Challenges, nuances, and trade-offs in modeling. International Journal of Artificial Intelligence in Education **35**(1), 304–329 (2025) https://doi.org/10.1007/s40593-024-00400-6

Pelánek, R., Papoušek, J., Řihák, J., Stanislav, V., Nižnan, J.: Elo-based learner modeling for the adaptive practice of facts. User Modeling and User-Adapted Interaction **27**(1), 89–118 (2017) https://doi.org/10.1007/s11257-016-9185-7

Shen, S., Liu, Q., Huang, Z., Zheng, Y., Yin, M., Wang, M., Chen, E.: A survey of knowledge tracing: Models, variants, and applications. IEEE Trans. Learn. Technol. **17**, 1898–1919 (2024) https://doi.org/10.1109/TLT.2024.3383325

San Pedro, M.O.Z., Baker, R.S.: Knowledge inference models used in adaptive learning. In: Davier, A.A., Mislevy, R.J., Hao, J. (eds.) Computational Psychometrics: New Methodologies for a New Generation of Digital Learning and Assessment: With Examples in R and Python, pp. 61–77. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-74394-9_5

d'Ham, C., Planche, M., Hoffmann, C., Wajeman, C., Girault, I.: Do we really need another learning platform? labnbook as a discipline-oriented authoring environment for scientific reporting. In: Proceedings of the 18th International Conference on

Computer Supported Education. CSEDU 2026, vol. 2, pp. 1678–1685. SciTePress, Benidorm, Spain (2026)

Vermeiren, H., Hofman, A.D., Bolsinova, M., van der Maas, H.L.J., Van Den Noortgate, W.: Balancing stability and flexibility: investigating a dynamic k value approach for the elo rating system in adaptive learning environments. User Modeling and User-Adapted Interaction **36**(1), 4 (2025) https://doi.org/10.1007/s11257-025-09439-z

Van Inwegen, E.G., Adjei, S.A., Wang, Y., Heffernan, N.T.: Using partial credit and response history to model user knowledge. International Educational Data Mining Society (2015)

Vermeiren, H., Kruis, J., Bolsinova, M., Maas, H.L.J., Hofman, A.D.: Psychometrics of an elo-based large-scale online learning system. Computers and Education: Artificial Intelligence **8**, 100376 (2025) https://doi.org/10.1016/j.caeai.2025.100376

Wang, Y., Heffernan, N.: Extending knowledge tracing to allow partial credit: Using continuous versus binary nodes. In: Lane, H.C., Yacef, K., Mostow, J., Pavlik, P. (eds.) Artificial Intelligence in Education vol. 7926, pp. 181–188. Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39112-5_19 . Series Title: Lecture Notes in Computer Science

Zhang, B., Shi, Y., Li, Y., Chai, C., Hou, L.: An enhanced Elo-based student model for polychotomously scored items in adaptive educational system. Interactive Learning Environments **31**(9), 5477–5494 (2023) https://doi.org/10.1080/10494820.2021.2010099

Šarić-Grgić, I., Grubišić, A., Gašpar, A.: Twenty-five years of bayesian knowledge tracing: a systematic review. User Modeling and User-Adapted Interaction **34**(4), 1127–1173 (2024) https://doi.org/10.1007/s11257-023-09389-4