DAS3H: Modeling Student Learning and Forgetting for Optimally Scheduling Distributed Practice of Skills

Benoît Choffin, Fabrice Popineau, Yolaine Bourda, Jill-Jênn Vie

DAS3H: Modeling Student Learning and Forgetting for Optimally Scheduling Distributed Practice of Skills

Benoît Choffin, Fabrice Popineau, Yolaine Bourda
LRI/CentraleSupélec – University of Paris-Saclay
Gif-sur-Yvette, France
{benoit.choffin, fabrice.popineau, yolaine.bourda}@lri.fr

Jill-Jênn Vie
RIKEN AIP
Tokyo, Japan
vie@jill-jenn.net

ABSTRACT

Spaced repetition is among the most studied learning strategies in the cognitive science literature. It consists in temporally distributing exposure to an information so as to improve long-term memorization. Providing students with an adaptive and personalized distributed practice schedule would benefit more than just a generic scheduler. However, the applicability of such adaptive schedulers seems to be limited to pure memorization, e.g. flashcards or foreign language learning. In this article, we first frame the research problem of optimizing an adaptive and personalized spaced repetition scheduler when memorization concerns the application of underlying multiple skills. To this end, we choose to rely on a student model for inferring knowledge state and memory dynamics on any skill or combination of skills. We argue that no knowledge tracing model takes both memory decay and multiple skill tagging into account for predicting student performance. As a consequence, we propose a new student learning and forgetting model suited to our research problem: DAS3H builds on the additive factor models and includes a representation of the temporal distribution of past practice on the skills involved by an item. In particular, DAS3H allows the learning and forgetting curves to differ from one skill to another. Finally, we provide empirical evidence on three real-world educational datasets that DAS3H outperforms other state-of-the-art EDM models. These results suggest that incorporating both item-skill relationships and forgetting effect improves over student models that consider one or the other.

Keywords

Student modeling, adaptive spacing, memory, knowledge components, q-matrix, optimal scheduling

1. INTRODUCTION

Learners have to manage their studying time wisely: they constantly have to make a trade-off between acquiring new knowledge and reviewing previously encountered learning material. Considering that learning often involves building on old knowledge (e.g. in mathematics) and that efforts undertaken in studying new concepts may be significant, this issue should not be taken lightly. However, only few school incentive structures encourage long-term retention, making students often favor short-term memorization and poor learning practices [37, 22].

Fortunately, there are simple learning strategies that help students efficiently manage their learning time and improve long-term memory retention at a small cost. Among them, the *spacing* and the *testing* effects have been widely replicated [36, 7] since their discovery in the 19^{th} century. Both of them are recommended by cognitive scientists [24, 46] in order to improve public instruction. The spacing effect states that temporally distributing learning episodes is more beneficial to long-term memory than learning in a single massed study session. The testing effect [35, 5] – also known as *retrieval practice* – basically consists in self-testing after being exposed to new knowledge instead of simply reading the lesson again. This test can take multiple forms: free recall, cued recall, multiple-choice questions, application exercises, and so on. A recent meta-analysis on the testing effect [1] found a strong and positive overall effect size of $g = 0.61$ for testing compared to non-testing reviewing strategies. Another meta-analysis [23] investigated whether learning with retrieval practice could transfer to different contexts and found a medium yet positive overall transfer effect size of $d = 0.40$. Combining both strategies is called *spaced retrieval practice*: temporally distributing tests after a first exposure to knowledge.

Recent research effort has been put on developing adaptive and personalized spacing schedulers for improving long-term retention of flashcards [40, 33, 18]. Compared to non-adaptive schedulers, they show substantial improvement of the learners' retention at immediate and delayed tests [19]. However, and to the best of our knowledge, there is no work on extending these algorithms when knowledge to be remembered concerns the application of underlying skills. Yet, the spacing effect is not limited to vocabulary learning or even pure memorization: it has been successfully applied to the acquisition and generalization of abstract science concepts [44] and to the practice of mathematical skills in a real educational setting [3]. Conversely, most models encountered in knowledge tracing involve multiple skills, but do not model forgetting. The goal of the present article is to start filling this gap by developing a student learning and forgetting

Benoît Choffin, Fabrice Popineau, Yolaine Bourda and Jill-Jênn Vie "DAS3H: Modeling Student Learning and Forgetting for Optimally Scheduling Distributed Practice of Skills" In: *Proceedings of The 12th International Conference on Educational Data Mining (EDM 2019)*, Collin F. Lynch, Agathe Merceron, Michel Desmarais, & Roger Nkambou (eds.) 2019, pp. 29 - 38

model for inferring skills knowledge state and memory dynamics. This model will serve as a basis for the future development of adaptive and personalized skill practice scheduling algorithms for improving learners' long-term memory.

Our contribution is two-fold. We first frame our research problem for extending the flashcards-based adaptive spacing framework to contexts where memorization concerns the application of underlying skills. In that perspective, students learn and reinforce skill mastery by practicing items involving that skill. We argue that this extension requires new student models to model learning and forgetting processes when multiple skills are involved by a single item. Thus, we also propose a new student model, coined DAS3H, that extends DASH [18, 22] and accounts for memory decay and the benefits of practice when an item can involve multiple knowledge components. Finally, we provide empirical evidence on three publicly available datasets showing that our model outperforms other state-of-the-art student models.

2. RELATED WORK

In this section, we first detail related work on adaptive spacing algorithms before turning to student modeling.

In what follows, we will index students by $s \in [\![1, S]\!]$, items (or questions, exercises) by $j \in [\![1, J]\!]$, skills or knowledge components (KCs) by $k \in [\![1, K]\!]$, and timestamps by $t \in \mathbb{R}^+$ (in days). To be more convenient, we assume that timestamps are encoded as the number of days elapsed since the first interaction with the system. It is sufficient because we only need to know the duration between two interactions. $Y_{s,j,t} \in \{0, 1\}$ gives the binary correctness of student s answering item j at time t. σ is the logistic function: $\forall x \in \mathbb{R}, \sigma(x) = 1/(1 + \exp(-x))$. $KC(.)$ takes as input an item index j and outputs the set of skill indices involved by item j.

Let us quickly detail what we mean by *skill*. In this article, we assimilate skills and knowledge components. Knowledge components are atomistic components of knowledge by which items are tagged. An item may have one or more KCs, and this information is synthesized by a so-called binary q-matrix [41]: $\forall (j, k) \in [\![1, J]\!] \times [\![1, K]\!]$, $q_{jk} = \mathbb{1}_{k \in KC(j)}$. We assume that the probability of answering correctly an item j that involves skill k depends on the student's mastery of skill k; conversely, we measure skill mastery by the ability of student s to remember skill k and apply it to solve any (possibly unseen) item that involves skill k.

2.1 Adaptive spacing algorithms

Adaptive spacing schedulers leverage the spaced retrieval learning strategy to maximize learning and retention of a set of items. They proceed by sequentially deciding which item to ask the user at any time based on the user's past study history. Items to memorize are often represented by flashcards, i.e. cards on which one side contains the question (e.g. *When did the Great Fire of London occur?* or *What is the correct translation of "manger" in English?*) and the other side contains the answer.

Early adaptive spacing systems made use of physical flashcards [17] but the advent of computer-assisted instruction made possible the development of electronic flashcards [51], thus allowing more complex and personalized strategies for optimal reviewing. Nowadays, several adaptive spacing softwares are available to the general public, e.g. Anki[1], SuperMemo[2], and Mnemosyne[3].

Originally, adaptive reviewing systems took decisions based on heuristics and handmade rules [17, 30, 51]. Though maybe effective in practice [20], these early systems lack performance guarantees [40]. Recent research works started to tackle this issue: for instance, Reddy et al. propose a mathematical formalization of the Leitner system and a heuristic approximation used for optimizing the review schedule [32].

A common approach for designing spaced repetition adaptive schedulers consists in modeling human memory statistically and recommending the item whose memory strength is closest to a fixed value θ [22, 18, 20]. Khajah, Lindsey and Mozer found that this simple heuristic is only slightly less efficient than exhaustive policy search in many situations [14]. It has the additional advantage to fit into the notion of "desirable difficulties" coined by Bjork [4]. Pavlik and Anderson [26] use an extended version of ACT-R declarative memory model to build an adaptive scheduler for optimizing item practice (in their case, Japanese-English word pairs) given a limited amount of time. ACT-R is originally capable of predicting item correctness and speed of recall by taking recency and frequency of practice into account. Pavlik and Anderson extend ACT-R to capture the spacing effect as well as item, learner, and item-learner interaction variability. The adaptive scheduler uses the model estimation of memory strength gain at retention test per unit of time to decide when to present each pair of words to a learner.

Other approaches do not rely on any memory model: Reddy, Levine and Dragan formalize this problem as a POMDP (Partially Observable Markov Decision Process) and approximately solve it within a deep reinforcement learning architecture [33]. However, they only test their algorithm on simulated students. A more recent work [40] formalizes the spaced repetition problem with marked temporal point processes and solves a stochastic optimal control problem to optimally schedule spaced review of items. Mettler, Massey and Kellman [19] compare an adaptive spacing scheduler (ARTS) to two fixed spacing conditions. ARTS leverages students' response times, performance, and number of trials to dynamically compute a priority score for adaptively scheduling item practice. Response time is used as an indicator of retrieval difficulty and thus, learning strength.

Our work can more generally relate to the problem of automatic optimization of teaching sequences. Rafferty et al. formulate this problem as a POMDP planning problem [31]. Whitehill and Movellan build on this work but use a hierarchical control architecture for selecting optimal teaching actions [48]. Lan et al. use contextual bandits to select the best next learning action by using an estimation of the student's knowledge profile [16]. Many intelligent tutoring systems (ITS) use mastery learning within the Knowledge Tracing [8] framework: making students work on a given

[1] `https://apps.ankiweb.net/`
[2] `https://www.supermemo.com/`
[3] `https://mnemosyne-proj.org/`

skill until the system infers that they have mastered it.

We can see that the traditional adaptive spacing framework already uses a spaced retrieval practice strategy to optimize the student's learning time. However, it is not directly adapted to learning and memorization of skills. In this latter case, specific items are the only way to practice one or multiple skills, because we do not have to memorize the content directly. Students who master a skill should be able to generalize to unseen items that also involve that skill. In Section 3, we propose an extension of this original framework in order to apply adaptive spacing algorithms to the memorization of skills.

2.2 Modeling student learning and forgetting

The history of scientific literature on student modeling is particularly rich. In what follows, we focus on the subproblem of modeling student learning and forgetting based on student performance data.

As Vie and Kashima recall [43], two main approaches have been used for modeling student learning and predicting student performance: Knowledge Tracing and Factor Analysis.

Knowledge Tracing [8] models the evolution of a student's knowledge state over time so as to predict a sequence of answers. The original and still most widespread model of Knowledge Tracing is Bayesian Knowledge Tracing (BKT). It is based on a Hidden Markov Model where the knowledge state of the student is the latent variable and skill mastery is assumed binary. Since its creation, it has been extended to overcome its limits and account for instance for individual differences between students [52]. More recently, Piech et al. replaced the original Hidden Markov Model framework with a Recurrent Neural Network and proposed a new Knowledge Tracing model called Deep Knowledge Tracing (DKT) [29]. Despite a mild controversy concerning the relevance of using deep learning in an educational setting [50], recent works continue to develop this line of research [53, 21].

Contrary to Knowledge Tracing, Factor Analysis does not originally take the order of the observations into account. IRT (Item Response Theory) [42] is the canonical model for Factor Analysis. In its simplest form, IRT reads:

$$\mathbb{P}(Y_{s,j} = 1) = \sigma(\alpha_s - \delta_j)$$

with α_s ability of student s and δ_j difficulty of item j. One of the main assumptions of IRT is that the student ability is static and cannot change over time or with practice. Despite its apparent simplicity, IRT has proven to be a robust and reliable EDM model, even outperforming much more complex architectures such as DKT [49]. IRT can be extended to represent user and item biases with vectors instead of scalars. This model is called MIRT, for Multidimensional Item Response Theory:

$$\mathbb{P}(Y_{s,j} = 1) = \sigma\left(\langle \alpha_s, \delta_j \rangle + d_j\right).$$

In this case, α_s and δ_j are d-dimensional vectors, and d_j is a scalar that captures the easiness of item j. $\langle ., . \rangle$ is the usual dot product between two vectors.

More recent works incorporated temporality in Factor Analysis models, by taking practice history into account. For instance, AFM (Additive Factor Model) [6] models:

$$\mathbb{P}(Y_{s,j} = 1) = \sigma\left(\sum_{k \in KC(j)} \beta_k + \gamma_k a_{s,k}\right)$$

with β_k easiness of skill k and $a_{s,k}$ number of attempts of student s on KC k prior to this attempt. Performance Factor Analysis [27] (PFA) builds on AFM and uses past outcomes of practice instead of simple encounter counts:

$$\mathbb{P}(Y_{s,j} = 1) = \sigma\left(\sum_{k \in KC(j)} \beta_k + \gamma_k c_{s,k} + \rho_k f_{s,k}\right)$$

with $c_{s,k}$ number of correct answers of student s on KC k prior to this attempt and $f_{s,k}$ number of wrong answers of student s on KC k prior to this attempt.

Ekanadham and Karklin take a step further to account for temporality in the IRT model and extend the two-parameter ogive IRT model (2PO model) by modeling the evolution of the student ability as a Wiener process [10]. However, they do not explicitly account for student memory decay.

The recent framework of KTM (Knowledge Tracing Machines) [43] encompasses several EDM models, including IRT, MIRT, AFM, and PFA. KTMs are based on factorization machines and model the probability of correctness as follows:

$$\mathbb{P}(Y_t = 1) = \sigma\left(\mu + \sum_{i=1}^{N} w_i x_{t,i} + \sum_{1 \leq i < \ell \leq N} x_{t,i} x_{t,\ell} \langle v_i, v_\ell \rangle\right)$$

where μ is a global bias, N is the number of abstract features, be it item parameters, temporal features, etc., x_t is a sample gathering all features collected at time t: which student answers which item, and information regarding prior attempts, w_i is the bias of feature i and $v_i \in \mathbb{R}^d$ its embedding. The features involved in a sample x_t are typically in sparse number, so this probability can be computed efficiently. In KTM, one can recover several existing EDM models by selecting the appropriate features to consider in the modeling. For instance, if we consider user and item features only, we recover IRT. If we consider the skill features in the q-matrix, and the counter of prior successes and failures at skill level, we recover PFA.

One of the very first works on human memory modeling dates back to 1885 and stems from Ebbinghaus [9]. He models the probability of recall of an item as an exponential function of memory strength and delay since last review. More recently, Settles and Meeder propose an extension of the original exponential forgetting curve model, the half-life regression [38]. They estimate item memory strength as an exponential function of a set of features that contain information on the past practice history and on the item to remember (lexeme tag features, in their case). More sophisticated memory models have also been proposed: for instance ACT-R (Adaptive Character of Thought–Rational) [2] and MCM (Multiscale Context Model) [25].

Walsh et al. [45] offer a comparison of three computational memory models: ACT-R declarative memory model [26],

Predictive Performance Equation (PPE) and a generalization of Search of Associative Memory (SAM). These models differ in how they predict the impact of spacing on subsequent relearning, after a long retention interval. PPE is the only one to predict that spacing may accelerate subsequent relearning ("*spacing accelerated relearning*") – an effect that was empirically underlined by their experiment. PPE showed also superior fit to experimental data, compared to SAM and ACT-R.

DASH [22, 18] bridges the gap between factor analysis and memory models. DASH stands for Difficulty, Ability, and Student History. Its formulation reads:

$$\mathbb{P}\left(Y_{s,j,t}=1\right)=\sigma(\alpha_s-\delta_j+h_\theta(\mathrm{t}_{s,j,1:l},\mathrm{y}_{s,j,1:l-1}))$$

with h_θ a function parameterized by θ (learned by DASH) that summarizes the effect of the $l-1$ previous attempts where student s reviewed item j ($\mathrm{t}_{s,j,1:l-1}$) and the binary outcomes of these attempts ($\mathrm{y}_{s,j,1:l-1}$). Their main choice for h_θ is:

$$h_\theta(\mathrm{t}_{s,j,1:l},\mathrm{y}_{s,j,1:l-1})=\sum_{w=0}^{W-1}\theta_{2w+1}\log(1+c_{s,j,w}) - \theta_{2w+2}\log(1+a_{s,j,w})$$

with w indexing a set of expanding time windows, $c_{s,j,w}$ is the number of correct outcomes of student s on item j in time window w out of a total of $a_{s,j,w}$ attempts. The time windows w are not disjoint and span increasing time intervals. They allow DASH to account for both learning and forgetting processes. The use of log counts induces diminishing returns of practice inside a given time window and difference of log counts formalizes a power law of practice. The time module h_θ is inspired by ACT-R [2] and MCM [25] memory models.

We can see that Lindsey et al. [18] make use of the additive factor models framework for taking memory decay and the benefits of past practice into account. Their model outperforms IRT and a baseline on their dataset COLT, with an accumulative prediction error metric. To avoid overfitting and making model training easier, they use a hierarchical Bayesian regularization.

To the best of our knowledge, no knowledge tracing model accounts for both multiple skills tagging *and* memory decay. We intend to bridge this gap by extending DASH.

3. FRAMING THE PROBLEM

In our setting, the student learns to master a set of skills by sequentially interacting with an adaptive spacing system. At each iteration, this system selects an item (or exercise, or question) for the student, e.g. *What is* $\lim_{x\to 0}(\sin x)/x$*?*. This selection is made by optimizing a utility function l that rewards long-term mastery of the set of KCs to learn. Then, the student answers the item and the system uses the correctness of the answer to update its belief concerning the student memory and learning state on the skills involved by the item. Finally, the system provides the student a corrective feedback.

In a nutshell, our present research goal is to maximize mastery and memory of a fixed set of skills among students during a given time interval while minimizing the time spent studying.

We rely on the following assumptions:

- information to learn and remember consists in a set of skills[4] $k \in [\![1, K]\!]$;
- skill mastery and memorization of student s at time t is measured by the ability of s to answer an (unseen) item involving that skill, i.e. by their ability to generalize to unseen material;
- students first have access to some theoretical knowledge about skills, but learning happens with retrieval practice;
- items are tagged with one or multiple skills and this information is synthesized inside a binary q-matrix [41];
- students forget: skill mastery decreases as time goes by since last practice of that skill;

Unlike Lindsey et al. [18], we do not assume that items involving skill k are interchangeable: their difficulties, for instance, may differ from one another. Thus, the selection phase is two-fold in that it requires to select the skill to practice *and* the item to present. In theory, there should be at least one item for practicing every skill k; in practice, one item would be too few, since the student would probably "overfit" on the item. This formalization easily encompasses the flashcards-based adaptive spacing framework: it only requires to associate every item with a distinct skill. This wipes out the need to select an item after the skill.

Different utility functions l can be considered. For instance, Reddy, Levine and Dragan consider both the likelihood of recalling all items and the expected number of items recalled [33]. In our case, the utility function should account for the uncertainty of future items to answer. Indeed, if the goal of the user is to prepare for an exam, the system must take into account that the user will probably have to answer items that they did not train with.

To tackle this problem, like previous work [22, 18], we choose to rely on a student learning and forgetting model. In our case, this model must be able to quantify mastery and memory for any skill or combination of skills. In the next section, we present our main contribution: a new student learning and forgetting model, coined DAS3H.

4. OUR MODEL DAS3H

We now describe our new student learning and forgetting model: DAS3H stands for item Difficulty, student Ability, Skill, and Student Skill practice History, and builds on the DASH model presented in Section 2. Lindsey et al. [18] show that DASH outperforms a hierarchical Bayesian version of IRT on their experimental data, which consist in student-item interactions on a flashcard-based foreign (Spanish) language vocabulary reviewing system. They already talk

[4]These skills may be organized into a graph of prerequisites, but this goes beyond the scope of this article.

about knowledge components, but they use this concept to cluster similar words together (e.g. all conjugations of a verb). Thus, in their setting, an item has exactly one knowledge component; different items can belong to the same knowledge component if they are close enough. As a consequence, their model formulation does not handle multiple skills item tagging, which is common in other disciplines such as in mathematics. Moreover, they assume that the impact of past practice on the probability of correctness does not vary across the skills: indeed, DASH has only two biases per time window w, θ_{2w+1} for past wins and θ_{2w+2} for past attempts. It may be a relevant assumption to prevent overfitting when the number of skills is high, but at the same time it may degrade performance when the set of skills is very diverse and inhomogeneous.

DAS3H extends DASH to items with multiple skills, and allows the influence of past practice on present performance to differ from one skill to another. One could argue that we could aggregate every existing combination of skills into a distinct skill to avoid the burden of handling multiple skills. However, this solution would not be satisfying since the resulting model would for instance not be able to capture item similarities between two items that share all but one skill in common. The use of a representation of multiple skills allows to account for knowledge transfer from one item to another. The item-skill relationships are usually synthesized by a q-matrix and generally require domain experts' labor.

We also leverage the recent Knowledge Tracing Machines framework [43] to enrich the DASH model by embedding the features in d dimensions and model pairwise interactions between those features. So far, KTMs have not been tried with memory features.

In brief, we extend DASH in three ways:

- Extension to handle multiple skills tagging: new temporal module h_θ that also takes the multiple skills into account. The influence of the temporal distribution of past practice and of the outcomes of these previous attempts may differ from one skill to another;
- Estimation of easiness parameters for *each* item j and skill k;
- Use of KTMs [43] instead of mere logistic regression.

For an embedding dimension of $d = 0$, the quadratic term of KTM is cancelled out and our model DAS3H reads:

$$\mathbb{P}\left(Y_{s,j,t}=1\right) = \sigma(\alpha_s - \delta_j + \sum_{k \in KC(j)} \beta_k + \\ + h_\theta\left(\mathrm{t}_{s,j,1:l}, \mathrm{y}_{s,j,1:l-1}\right)).$$

Following Lindsey et al. [18], we choose:

$$h_\theta(\mathrm{t}_{s,j,1:l}, \mathrm{y}_{s,j,1:l-1}) = \sum_{k \in KC(j)} \sum_{w=0}^{W-1} \theta_{k,2w+1} \log(1 + c_{s,k,w}) \\ - \theta_{k,2w+2} \log(1 + a_{s,k,w}).$$

Thus, the probability of correctness of student s on item j at time t depends on their ability α_s, the difficulty of the item δ_j and the sum of the easiness β_k of the skills involved by item j. It also depends on the temporal distribution and the outcomes of past practice, synthesized by h_θ. In h_θ, w denotes the index of the time window, $c_{s,k,w}$ denotes the amount of times that KC k has been correctly recalled in window w by student s earlier, $a_{s,k,w}$ the amount of times that KC k has been encountered in time window w by student s earlier. Intuitively, h_θ can be seen as a sum of memory strengths, one for each skill involved in item j.

For higher embedding dimensions $d > 0$, in our implementation we use probit as the link function. All features are embedded in d dimensions and their interaction is modeled in a pairwise manner. For a more thorough description of KTMs, see [43]. To implement a model within the KTM framework, one must decide which features to encode in the sparse x vector. In our case, we chose user s, item j, skills $k \in KC(j)$, wins $c_{s,k,w}$ and attempts $a_{s,k,w}$ for each time window w.

Compared to DASH and if we forget about additional parameters induced by the regularization scheme, DAS3H has $(d+1)(K+2W(K-1))$ more feature parameters to estimate. To avoid overfitting, we use additional hierarchical distributional assumptions for the parameters to estimate, as described in the next section.

5. EXPERIMENTS

To evaluate the performance of our model, we compared DAS3H to several state-of-the-art student models on three different educational datasets. These models have been detailed in Section 2.

5.1 Experimental setting

We perform 5-fold cross-validation at the student level for our experiments. This means that the student population is split into 5 disjoint groups and that cross-validation is made on this basis. This evaluation method, also used in [43], has the advantage to show how well an educational data mining model generalizes over previously unseen students.

Following previous work [34, 43] we use hierarchical distributional assumptions when $d > 0$ to help model training and avoid overfitting. More precisely, each feature weight and feature embedding component follows a normal prior distribution $\mathcal{N}(\mu, 1/\lambda)$ where μ and λ follow hyperpriors $\mu \sim \mathcal{N}(0,1)$ and $\lambda \sim \Gamma(1,1)$. In their article [18], Lindsey et al. took a similar approach but they assumed that the α_s and the δ_i followed different distributions. Contrary to us, they did not regularize the parameters θ_w associated with the practice history of a student: our situation is different because we have more parameters to estimate than them. We use the same time windows as Lindsey et al. [18]: $\{1/24, 1, 7, 30, +\infty\}$. Time units are expressed in days.

Our models were implemented in Python. Code for replicating our results is freely available on Github[5]. Like Vie and Kashima [43], we used `pywFM`[6] as wrapper for `libfm`[7] [34] for models with $d > 0$. We used 300 iterations for the MCMC

[5] `https://github.com/BenoitChoffin/das3h`
[6] `https://github.com/jfloff/pywFM`
[7] `http://libfm.org/`

Gibbs sampler. When $d = 0$, we used the `scikit-learn` [28] implementation of logistic regression with L2 regularization.

We compared DAS3H to DASH, IRT, PFA, and AFM within the KTM framework, for three different embedding dimensions: 0, 5, and 20. When $d > 0$, IRT becomes MIRTb, a variant of MIRT that considers a user bias. We do not compare to DKT, due to the mild controversy over its performance [49, 50]. For DASH, we choose to consider item-specific biases, and not KC-specific biases: in their original setting, Lindsey et al. [18] aggregated items into equivalence classes and trained DASH on this basis. This is not always possible to us because items have in general multiple skill taggings; however, we tested this possibility in Subsection 5.3 but it did not yield better results.

We used three different datasets: ASSISTments 2012-2013 (assist12) [11], Bridge to Algebra 2006-2007 (bridge06) and Algebra I 2005-2006 (algebra05) [39]. The two latter datasets stem from the KDD Cup 2010 EDM Challenge. The main problem for our experiments was that only few datasets that combine both time variables and multiple-KC tagging are publicly available. As a result, only both KDD Cup 2010 datasets have items that involve multiple KCs at the same time. As a further work, we plan to test DAS3H on datasets spanning more diverse knowledge domains and having more fine-grained skill taggings. In ASSISTments 2012-2013, the `problem_id` variable was used for the items and for the KDD Cup datasets, the item variable came from the concatenation of the problem and the step IDs, as recommended by the challenge organizers.

We removed users for whom the number of interactions was less than 10. We also removed interactions with NaN skills, because we feared it would introduce too much noise. For the KDD Cup 2010 datasets, we removed interactions which seemed to be duplicates, i.e. for which the (user, item, timestamp) tuple was duplicated. Finally, we sparsely encoded the features and computed the q-matrices. We detail the dataset characteristics (after preprocessing) in Table 1. The mean skill delay refers to the mean time interval (in days) between two interactions with the same skill, and the mean study period refers to the mean time difference between the last and the first interaction for each student.

5.2 Results

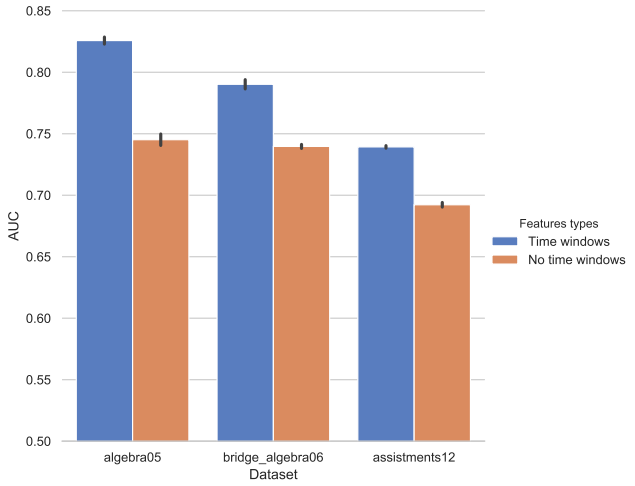
Detailed results can be found in Tables 2, 3 and 4, where mean area under the curve scores (AUC) and mean negative log-likelihood (NLL) are reported for each model and dataset. Accuracy (ACC) is not reported by lack of space. We found that ACC was highly correlated with AUC and NLL; the interested reader can find it on the Github repository containing code for the experiments[8]. Standard deviations over the 5 folds are also reported. We can see that our model DAS3H outperforms all other models on every dataset.

5.3 Discussion

Our experimental results show that DAS3H is able to more accurately model student performance when multiple skill and temporal information is at hand. We hypothesize that this performance gain stems from a more complex temporal modeling of the influence of past practice of skills on current performance.

[8] `https://github.com/BenoitChoffin/das3h`

Figure 1: AUC boost when using time windows features instead of regular wins and attempts (all datasets). Higher is better.



The impact of the multidimensional embeddings and the pairwise interactions seems to be very small yet unclear, and should be further investigated. An embedding dimension of $d = 20$ is systematically worse or among the worst for DAS3H on every dataset, but with a smaller $d = 5$, the performance is sometimes better than with $d = 0$. An intermediate embedding dimension could be preferable, but our results confirm those of Vie and Kashima [43]: the role of the dimension d seems to be limited.

In order to make more sense of our results, we wanted to know what made DAS3H more predictive than its counterparts. Our hypothesis was that taking the past temporal distribution of practice as well as the outcome of previous encounters with skills allowed the model to capture more complex phenomena than just simple practice, such as forgetting. To test this hypothesis, we performed some ablation tests. We empirically evaluated the difference in terms of AUC on our datasets when time windows features were used instead of regular features for wins and attempts. For each dataset, we compared the mean AUC score of the original DAS3H model with a similar model for which the time windows wins and attempts features were replaced with regular wins and fails counts. Thus, the time module h_θ was replaced with $\sum_{k \in KC(j)} \gamma_k c_{s,k} + \rho_k f_{s,k}$ like in PFA. Since wins, fails and attempts are collinear, it does not matter to replace "wins and attempts" with "wins and fails". The results are plotted in Figure 1. Mean and standard deviations over 5 folds are reported. We chose an embedding dimension $d = 0$ since it was in general the best on the previous experiments. We observe that using time window features consistently boosts the AUC of the model.

We also wanted to know if assuming that skill practice benefits should differ from one skill to another was a useful assumption. Thus, we compared our original DAS3H formulation to a different version, closer to the DASH formula-

Dataset	Users	Items	Skills	Interactions	Mean correctness	Skills per item	Mean skill delay	Mean study period
assist12	24,750	52,976	265	2,692,889	0.696	1.000	8.54	98.3
bridge06	1,135	129,263	493	1,817,427	0.832	1.013	0.83	149.5
algebra05	569	173,113	112	607,000	0.755	1.363	3.36	109.9

Table 1: Datasets characteristics

model	dim	AUC ↑	NLL ↓
DAS3H	0	**0.826** $\pm$ 0.003	**0.414** $\pm$ 0.011
DAS3H	5	**0.818** $\pm$ 0.004	**0.421** $\pm$ 0.011
DAS3H	20	**0.817** $\pm$ 0.005	**0.422** $\pm$ 0.007
DASH	5	0.775 $\pm$ 0.005	0.458 $\pm$ 0.012
DASH	20	0.774 $\pm$ 0.005	0.456 $\pm$ 0.017
DASH	0	0.773 $\pm$ 0.002	0.454 $\pm$ 0.006
IRT	0	0.771 $\pm$ 0.007	0.456 $\pm$ 0.015
MIRTb	20	0.770 $\pm$ 0.007	0.460 $\pm$ 0.007
MIRTb	5	0.770 $\pm$ 0.004	0.459 $\pm$ 0.011
PFA	0	0.744 $\pm$ 0.004	0.481 $\pm$ 0.004
AFM	0	0.707 $\pm$ 0.005	0.499 $\pm$ 0.006
PFA	20	0.670 $\pm$ 0.010	1.008 $\pm$ 0.047
PFA	5	0.664 $\pm$ 0.010	1.107 $\pm$ 0.079
AFM	20	0.644 $\pm$ 0.005	0.817 $\pm$ 0.076
AFM	5	0.640 $\pm$ 0.007	0.941 $\pm$ 0.056

Table 2: Performance comparison on the Algebra 2005-2006 (PSLC DataShop) dataset. Metrics are averaged over 5 folds and standard deviations are reported. ↑ and ↓ respectively indicate that higher (lower) is better.

model	dim	AUC ↑	NLL ↓
DAS3H	5	**0.744** $\pm$ 0.002	**0.531** $\pm$ 0.001
DAS3H	20	**0.740** $\pm$ 0.001	**0.533** $\pm$ 0.003
DAS3H	0	**0.739** $\pm$ 0.001	**0.534** $\pm$ 0.002
DASH	0	0.703 $\pm$ 0.002	0.557 $\pm$ 0.004
DASH	5	0.703 $\pm$ 0.001	0.557 $\pm$ 0.001
DASH	20	0.703 $\pm$ 0.002	0.557 $\pm$ 0.002
IRT	0	0.702 $\pm$ 0.001	0.558 $\pm$ 0.001
MIRTb	20	0.701 $\pm$ 0.001	0.558 $\pm$ 0.001
MIRTb	5	0.701 $\pm$ 0.002	0.558 $\pm$ 0.001
PFA	5	0.669 $\pm$ 0.002	0.577 $\pm$ 0.002
PFA	20	0.668 $\pm$ 0.002	0.578 $\pm$ 0.003
PFA	0	0.668 $\pm$ 0.002	0.579 $\pm$ 0.002
AFM	5	0.610 $\pm$ 0.001	0.597 $\pm$ 0.001
AFM	20	0.609 $\pm$ 0.001	0.597 $\pm$ 0.003
AFM	0	0.608 $\pm$ 0.002	0.598 $\pm$ 0.002

Table 3: Performance comparison on the ASSISTments 2012-2013 dataset. Metrics are averaged over 5 folds and standard deviations are reported. ↑ and ↓ respectively indicate that higher (lower) is better.

tion, in which all skills share the same parameters θ_{2w+1} and θ_{2w+2} inside a given time window w. We refer to this version of DAS3H as DAS3H$_{1p}$. The results are given in Table 5. They show that using different parameters for different skills in h_θ increases AUC performance. The AUC gain varies between +0.03 and +0.04. This suggests that some skills have significantly different learning and forgetting curves.

One could argue also that this comparison between DAS3H and DASH is not totally accurate. In their papers, Lindsey et al. cluster similar items together to form disjoint knowledge components. This is not possible to perform directly for both KDD Cup datasets since some items have been tagged with multiple skills. Nevertheless, the ASSISTments 2012-2013 dataset has only single-KC items. To evaluate whether considering the temporal distribution and the outcomes of past practice on the KCs (DASH [KC]) or on the items (DASH [items]) would be better, we compared these two DASH formulations on ASSISTments 2012-2013. Detailed results can be found in Table 6. We see that DASH [items] and DASH [KC] have comparable performance.

Finally, let us illustrate the results of DAS3H by taking two examples of KCs of Algebra I 2005-2006, one for which the estimated forgetting curve slope is steep, the other one for which it is more flat. As a proxy for the forgetting curve slope, we computed the difference of correctness probabilities when a "win" (i.e. a correct outcome when answering an item involving a skill) left a single time window. This difference was computed for every skill, for every couple of time

model	dim	AUC ↑	NLL ↓
DAS3H	5	**0.791** $\pm$ 0.005	**0.369** $\pm$ 0.005
DAS3H	0	**0.790** $\pm$ 0.004	**0.371** $\pm$ 0.004
DAS3H	20	0.776 $\pm$ 0.023	0.387 $\pm$ 0.027
DASH	0	0.749 $\pm$ 0.002	0.393 $\pm$ 0.007
DASH	20	0.747 $\pm$ 0.003	0.399 $\pm$ 0.002
IRT	0	0.747 $\pm$ 0.002	0.393 $\pm$ 0.007
DASH	5	0.747 $\pm$ 0.003	0.399 $\pm$ 0.002
MIRTb	5	0.746 $\pm$ 0.002	0.398 $\pm$ 0.006
MIRTb	20	0.746 $\pm$ 0.004	0.399 $\pm$ 0.007
PFA	20	0.746 $\pm$ 0.003	0.397 $\pm$ 0.004
PFA	5	0.744 $\pm$ 0.007	0.402 $\pm$ 0.007
PFA	0	0.739 $\pm$ 0.003	0.406 $\pm$ 0.008
AFM	5	0.706 $\pm$ 0.002	0.411 $\pm$ 0.004
AFM	20	0.706 $\pm$ 0.002	0.412 $\pm$ 0.004
AFM	0	0.692 $\pm$ 0.002	0.423 $\pm$ 0.006

Table 4: Performance comparison on the Bridge to Algebra 2006-2007 (PSLC DataShop) dataset. Metrics are averaged over 5 folds and standard deviations are reported. ↑ and ↓ respectively indicate that higher (lower) is better.

	d	bridge06	algebra05	assist12
DAS3H	0	**0.790** $\pm$ 0.004	**0.826** $\pm$ 0.003	0.739 $\pm$ 0.001
	5	**0.791** $\pm$ 0.005	0.818 $\pm$ 0.004	**0.744** $\pm$ 0.002
	20	0.776 $\pm$ 0.023	0.817 $\pm$ 0.005	0.740 $\pm$ 0.001
$DAS3H_{1p}$	0	0.757 $\pm$ 0.003	0.789 $\pm$ 0.009	0.701 $\pm$ 0.002
	5	0.757 $\pm$ 0.005	0.787 $\pm$ 0.005	0.700 $\pm$ 0.001
	20	0.757 $\pm$ 0.003	0.789 $\pm$ 0.006	0.701 (<1e-3)

Table 5: AUC comparison on all datasets between DAS3H and $DAS3H_{1p}$, a version of DAS3H for which the influence of past practice does not differ from one skill to another. Standard deviations are reported. Higher is better.

DASH	$d = 0$	$d = 5$	$d = 20$
items	0.703 $\pm$ 0.002	0.703 $\pm$ 0.001	0.703 $\pm$ 0.002
KC	0.702 $\pm$ 0.001	0.701 $\pm$ 0.001	0.701 $\pm$ 0.001

Table 6: AUC comparison on ASSISTments 2012-2013 between DASH [items] and DASH [KC]. Standard deviations are reported. Higher is better.

windows, and for every fold. The differences were then averaged over the 5 folds and over the different time windows, yielding for every skill the probability of correctness average decrease when a win leaves a single time window. One of the skills for which memory decays slowly concerns shading an area for which a given value is inferior to a threshold: in average and everything else being equal, the probability of correctness for an item involving this skill decreases by 1.15% when a single "win" leaves a time window. Such a skill is indeed not difficult for a student to master with a few periodic reviews. On the contrary, the skill concerning the application of exponents is more difficult to remember as time goes by: for this KC, the correctness probability decreases by 2.74% when a win leaves a time window. This is more than the double of the previous amount and is consistent with the description of the KC.

In brief, we saw in this section that DAS3H outperforms the other EDM models to which we compared it – including DASH. Using time window features instead of regular skill wins and attempts counts and estimating different parameters for different skills significantly boosts performance. Considering that DAS3H outperforms its ablated counterparts and DASH, these results suggest that including both item-skill relationships and forgetting effect improves over models that consider one or the other. Using multidimensional embeddings, however, did not seem to provide richer feature representations, contrary to our expectations.

Besides its performance, DAS3H has the advantage to be suited to the adaptive skill practice scheduling problem we described in Section 3. Indeed, it encapsulates an estimation of the current mastery of any skill and combination of skills for student s. It can also be used to infer its future evolution and thus, be leveraged to adaptively optimize a personalized skill practice schedule.

6. CONCLUSION AND FUTURE WORK

In this article, we first formulated a research framework for addressing the problem of optimizing human long-term memory of skills. More precisely, the knowledge to be remembered here is *applicative*: we intend to maximize the period during which a human learner will be able to leverage their retention of a skill they practiced to answer an item involving this skill. This framework assumes multiple skills tagging and is adapted to the more common flashcards-based adaptive review schedulers.

We take a student modeling approach to start addressing this issue. As a first step towards an efficient skill practice scheduler for optimizing human long-term memory, we thus propose a new student learning and forgetting model coined DAS3H which extends the DASH model proposed by Lindsey et al. [18]. Contrary to DASH, DAS3H allows each item to depend on an arbitrary number of knowledge components. Moreover, a bias for each skill temporal feature is estimated, whereas DASH assumed that item practice memory decayed at the same rate for every item. Finally, DAS3H is based on the recent Knowledge Tracing Machines model [43] because feature embeddings and pairwise interactions between variables could provide richer models. To the best of our knowledge, KTMs have never been used with memory features so far. Finally, we showed that DAS3H outperforms several state-of-the-art EDM models on three real-world educational datasets that include information on timestamps and KCs. We showed that adding time windows features and assuming different learning and forgetting curves for different skills significantly boosts AUC performance.

This work could be extended in different ways. First, the additive form of our model makes it compensatory. In other terms, if an item j involves two skills k_1 and k_2, a student could compensate a small practice in k_1 by increasing their practice in k_2. This is the so-called "explaining away" issue [47]. Using other non-affine models [15] could be relevant.
Following Lindsey et al. [18], we used 5 time windows for DAS3H during our experiments: $\{1/24, 1, 7, 30, +\infty\}$. Future work could investigate the impact of alternative sets of time windows – for instance, with more fine-grained time scales. However, one should pay attention not to add too many parameters to estimate.
Future work should also compare DAS3H and DASH to additional student models. For instance, R-PFA [12] (Recent-Performance Factor Analysis) and PFA-decay [13] extend and outperform PFA by leveraging a representation of past practice that puts more weight on more recent interactions. However, they do not explicitly take the temporal distribution of past practice to predict future student performance. Other memory models, such as ACT-R [26] or MCM [25] could also be tested against DAS3H. Latency, or speed of recall, can serve as a proxy of retrieval difficulty and memory strength [19]. It would be interesting to test whether incorporating this information inside DAS3H would result in better model performance.
In a real-world setting, items generally involve multiple skills at the same time. In such a situation, how should one select the next item to recommend a user so as to maximize their long-term memory? The main issue here is that we want to anchor skills in their memory, not specific items. We could think of a two-step recommendation strategy: first, select-

ing the skill $k^\star$ whose recall probability is closest to a given threshold (this strategy is consistent with the cognitive psychology literature, as Lindsey et al. recall [18]) and second, selecting an item among the pool of items that involve this skill. However, it could be impossible to find an item that involves *only* this skill $k^\star$. Also, precocious skill reactivations can have a harmful impact on long-term memory [7]. Thus, a strategy could be to compute a score (weighted according to the recall probability of each individual skill) for each skill combination in the q-matrix and to choose the combination for which the score is optimized.

Finally, we tested our model on three real-world educational datasets collected from automatic teaching systems on mathematical knowledge. To experiment with our model, we were indeed constrained in our choice of datasets, since few publicly available of them provide both information on the timestamps and the skills of the interactions. As further work, we intend to test our model on other datasets, from more diverse origins and concerning different knowledge domains. Collecting large, fine-grained and detailed educational datasets concerning diverse disciplines and making them publicly available would more generally allow EDM researchers to test richer models.

Acknowledgements

We would like to warmly thank Pr. Mozer from Univ. of Colorado, Boulder for providing useful details on their papers [22, 18] and allowing us to access the data of their experiment, and to Alice Latimier (LSCP, Paris) for her crucial comments concerning the cognitive science part. This work was funded by Caisse des Depôts et Consignations, e-Fran program.

7. REFERENCES

[1] O. O. Adesope, D. A. Trevisan, and N. Sundararajan. Rethinking the use of tests: A meta-analysis of practice testing. *Review of Educational Research*, 87(3):659–701, 2017.

[2] J. R. Anderson, M. Matessa, and C. Lebiere. ACT-R: A theory of higher level cognition and its relation to visual attention. *Human-Computer Interaction*, 12(4):439–462, 1997.

[3] K. Barzagar Nazari and M. Ebersbach. Distributing mathematical practice of third and seventh graders: Applicability of the spacing effect in the classroom. *Applied Cognitive Psychology*, 33(2):288–298, 2019.

[4] R. A. Bjork. Memory and metamemory considerations in the training of human beings. *Metacognition: Knowing about knowing*, 185, 1994.

[5] S. K. Carpenter, H. Pashler, J. T. Wixted, and E. Vul. The effects of tests on learning and forgetting. *Memory & Cognition*, 36(2):438–448, 2008.

[6] H. Cen, K. Koedinger, and B. Junker. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *International Conference on Intelligent Tutoring Systems*, pages 164–175. Springer, 2006.

[7] N. J. Cepeda, E. Vul, D. Rohrer, J. T. Wixted, and H. Pashler. Spacing effects in learning: A temporal ridgeline of optimal retention. *Psychological science*, 19(11):1095–1102, 2008.

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

[9] H. Ebbinghaus. Memory: A contribution to experimental psychology. *Annals of neurosciences*, 20(4):155, 2013.

[10] C. Ekanadham and Y. Karklin. T-SKIRT: online estimation of student proficiency in an adaptive learning system. In *Machine Learning for Education Workshop at ICML*, 2015.

[11] M. Feng, N. Heffernan, and K. Koedinger. Addressing the assessment challenge with an online system that tutors as it assesses. *User Modeling and User-Adapted Interaction*, 19(3):243–266, 2009.

[12] A. Galyardt and I. Goldin. Move your lamp post: Recent data reflects learner knowledge better than older data. *Journal of Educational Data Mining*, 7(2):83–108, 2015.

[13] Y. Gong, J. E. Beck, and N. T. Heffernan. How to construct more accurate student models: Comparing and optimizing knowledge tracing and performance factor analysis. *International Journal of Artificial Intelligence in Education*, 21(1-2):27–46, 2011.

[14] M. M. Khajah, R. V. Lindsey, and M. C. Mozer. Maximizing students' retention via spaced review: Practical guidance from computational models of memory. *Topics in cognitive science*, 6(1):157–169, 2014.

[15] A. Lan, T. Goldstein, R. Baraniuk, and C. Studer. Dealbreaker: A nonlinear latent variable model for educational data. In *International Conference on Machine Learning*, pages 266–275, 2016.

[16] A. S. Lan and R. G. Baraniuk. A contextual bandits framework for personalized learning action selection. In *Proceedings of the 9th International Conference on Educational Data Mining, EDM 2016*, pages 424–429, 2016.

[17] S. Leitner. So lernt man lernen [how to learn]. *Freiburg im Breisgau, Germany: Herder*, 1972.

[18] R. V. Lindsey, J. D. Shroyer, H. Pashler, and M. C. Mozer. Improving students' long-term knowledge retention through personalized review. *Psychological science*, 25(3):639–647, 2014.

[19] E. Mettler, C. M. Massey, and P. J. Kellman. A comparison of adaptive and fixed schedules of practice. *Journal of Experimental Psychology: General*, 145(7):897, 2016.

[20] C. Metzler-Baddeley and R. J. Baddeley. Does adaptive training work? *Applied Cognitive Psychology: The Official Journal of the Society for Applied Research in Memory and Cognition*, 23(2):254–266, 2009.

[21] S. Minn, Y. Yu, M. C. Desmarais, F. Zhu, and J.-J. Vie. Deep knowledge tracing and dynamic student classification for knowledge tracing. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 1182–1187. IEEE, 2018.

[22] M. C. Mozer and R. V. Lindsey. Predicting and improving memory retention: Psychological theory matters in the big data era. In *Big Data in Cognitive Science*, pages 43–73. Psychology Press, 2016.

[23] S. C. Pan and T. C. Rickard. Transfer of test-enhanced learning: Meta-analytic review and synthesis. *Psychological bulletin*, 144(7):710, 2018.

[24] H. Pashler, P. M. Bain, B. A. Bottge, A. Graesser, K. Koedinger, M. McDaniel, and J. Metcalfe.

Organizing instruction and study to improve student learning. IES practice guide. NCER 2007-2004. *National Center for Education Research*, 2007.

[25] H. Pashler, N. Cepeda, R. V. Lindsey, E. Vul, and M. C. Mozer. Predicting the optimal spacing of study: A multiscale context model of memory. In *Advances in neural information processing systems*, pages 1321–1329, 2009.

[26] P. I. Pavlik and J. R. Anderson. Using a model to compute the optimal schedule of practice. *Journal of Experimental Psychology: Applied*, 14(2):101, 2008.

[27] P. I. Pavlik, H. Cen, and K. R. Koedinger. Performance factors analysis - A new alternative to knowledge tracing. In *Proceedings of the 14th International Conference on Artificial Intelligence in Education, AIED 2009*, pages 531–538, 2009.

[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in Python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[29] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. In *Advances in neural information processing systems*, pages 505–513, 2015.

[30] P. Pimsleur. A memory schedule. *The Modern Language Journal*, 51(2):73–75, 1967.

[31] A. N. Rafferty, E. Brunskill, T. L. Griffiths, and P. Shafto. Faster teaching by POMDP planning. In *International Conference on Artificial Intelligence in Education*, pages 280–287. Springer, 2011.

[32] S. Reddy, I. Labutov, S. Banerjee, and T. Joachims. Unbounded human learning: Optimal scheduling for spaced repetition. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1815–1824. ACM, 2016.

[33] S. Reddy, S. Levine, and A. Dragan. Accelerating human learning with deep reinforcement learning. In *NIPS'17 Workshop: Teaching Machines, Robots, and Humans*, 2017.

[34] S. Rendle. Factorization machines with libfm. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3(3):57, 2012.

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

[36] H. L. Roediger III and J. D. Karpicke. Intricacies of spaced retrieval: A resolution. In *Successful Remembering and Successful Forgetting*, pages 41–66. Psychology Press, 2011.

[37] H. L. Roediger III and K. B. McDermott. Remembering what we learn. In *Cerebrum: the Dana Forum on Brain Science*, volume 2018. Dana Foundation, 2018.

[38] B. Settles and B. Meeder. A trainable spaced repetition model for language learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1848–1858, 2016.

[39] J. Stamper, A. Niculescu-Mizil, S. Ritter, G. Gordon, and K. Koedinger. Algebra I 2005-2006 and Bridge to Algebra 2006-2007. Development data sets from KDD Cup 2010 Educational Data Mining Challenge. Find them at `http://pslcdatashop.web.cmu.edu/KDDCup/downloads.jsp`.

[40] B. Tabibian, U. Upadhyay, A. De, A. Zarezade, B. Schölkopf, and M. Gomez-Rodriguez. Enhancing human learning via spaced repetition optimization. *Proceedings of the National Academy of Sciences*, 116(10):3988–3993, 2019.

[41] K. K. Tatsuoka. Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of educational measurement*, 20(4):345–354, 1983.

[42] W. J. van der Linden and R. K. Hambleton. *Handbook of modern item response theory*. Springer Science & Business Media, 2013.

[43] J.-J. Vie and H. Kashima. Knowledge Tracing Machines: Factorization Machines for Knowledge Tracing. In *Proceedings of the 33th AAAI Conference on Artificial Intelligence*, page to appear, 2019.

[44] H. A. Vlach and C. M. Sandhofer. Distributing learning over time: The spacing effect in children's acquisition and generalization of science concepts. *Child development*, 83(4):1137–1144, 2012.

[45] M. M. Walsh, K. A. Gluck, G. Gunzelmann, T. Jastrzembski, M. Krusmark, J. I. Myung, M. A. Pitt, and R. Zhou. Mechanisms underlying the spacing effect in learning: A comparison of three computational models. *Journal of Experimental Psychology: General*, 147(9):1325, 2018.

[46] Y. Weinstein, C. R. Madan, and M. A. Sumeracki. Teaching the science of learning. *Cognitive Research: Principles and Implications*, 3(1):2, 2018.

[47] M. P. Wellman and M. Henrion. Explaining "explaining away". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(3):287–292, 1993.

[48] J. Whitehill and J. Movellan. Approximately optimal teaching of approximately optimal learners. *IEEE Transactions on Learning Technologies*, 11(2):152–164, 2018.

[49] K. H. Wilson, Y. Karklin, B. Han, and C. Ekanadham. Back to the basics: Bayesian extensions of IRT outperform neural networks for proficiency estimation. In *Proceedings of the 9th International Conference on Educational Data Mining, EDM 2016*, pages 539–544, 2016.

[50] K. H. Wilson, X. Xiong, M. Khajah, R. V. Lindsey, S. Zhao, Y. Karklin, E. G. Van Inwegen, B. Han, C. Ekanadham, J. E. Beck, et al. Estimating student proficiency: Deep learning is not the panacea. In *Neural Information Processing Systems, Workshop on Machine Learning for Education*, page 3, 2016.

[51] P. Wozniak and E. J. Gorzelanczyk. Optimization of repetition spacing in the practice of learning. *Acta neurobiologiae experimentalis*, 54:59–59, 1994.

[52] M. V. Yudelson, K. R. Koedinger, and G. J. Gordon. Individualized bayesian knowledge tracing models. In *International Conference on Artificial Intelligence in Education*, pages 171–180. Springer, 2013.

[53] J. Zhang, X. Shi, I. King, and D.-Y. Yeung. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on World Wide Web*, pages 765–774. International World Wide Web Conferences Steering Committee, 2017.