Shih, JL. et al. (Eds.) (2023). Proceedings of the 31st International Conference on Computers in Education. Asia-Pacific Society for Computers in Education

Studying Memory Decay and Spacing within Knowledge Tracing

Cristina MAIER[a*], Isha SLAVIN[b], Ryan S. BAKER[c], & Steve STALZER[d]
[a]*McGraw-Hill Education, USA*
[b]*McGraw-Hill Education, USA*
[c]*University of Pennsylvania, USA*
[d]*McGraw-Hill Education, USA*
*cristina.maier@mheducation.com

Abstract: Knowledge Tracing estimates a student's knowledge on a set of skills and predicts whether the student will answer correctly if given a question linked to subsets of such skills. We conduct an in-depth analysis on the best ways to apply the cognitive science principles such as memory decay and spacing within knowledge tracing, proposing new algorithms, MemDec and MemDec Spacing, to do so. We explore different methods of modeling the rate and weight of decay, with and without spacing, and analyze their impact on predicting performance in real-world data. Variations of the model are compared between each other as well as against other existing algorithms.

Keywords: Knowledge tracing, memory decay, spacing effect, learning systems

1. Introduction

Knowledge tracing attempts to estimate student mastery on a set of skills from their performance. The outputs of knowledge tracing have several uses, including use in behavioral and self-regulated learning models, reports to instructors with actionable insights, and driving mastery learning within adaptive learning systems (Pelánek, 2017). Although the majority of the most recent work in knowledge tracing has investigated refinements to deep learning algorithms (Piech et al., 2015), there has also been interest in developing knowledge tracing algorithms that leverage findings from cognitive science (Choffin et al., 2019; Pavlik et al., 2021) and studying how broadly applicable they are (Schmucker & Mitchell, 2022). In this paper, we investigate the implementation of cognitive science principles within knowledge tracing in detail, specifically focusing on memory decay and the spacing effect when applied to variants of some components of Logistic Knowledge Tracing (LKT) (Pavlik et al., 2021) with an emphasis on the components from Performance Factor Analysis (PFA) (Pavlik et al., 2009) and Recent-Performance Factor Analysis (R-PFA) (Galyardt & Goldin, 2014). We chose to study memory decay and spacing within this framework because of PFA's interpretability and its support of multi-skill items. To study these concepts, we introduce a new algorithm that incorporates memory decay into the knowledge estimation and investigate how the spacing effect and different ways of governing the increase in memory decay influence modeling student performance. We consider two methods for modeling memory decay: one that uses the practice order (as in Galyardt & Goldin, 2014), and a second that uses a time window approach. Incorporating these concepts allows for the differentiation between a student's current knowledge versus previous comprehension that may have been lost over time.

In Section 2 we discuss the memory decay and spacing effect theories, and describe the well-known knowledge tracing algorithm, Performance Factor Analysis (PFA) (Pavlik et al., 2009). Section 3 contains Related Work investigating issues surrounding memory within knowledge tracing. In Section 4 we introduce a new algorithm, MemDec, and a variant, MemDec Spacing, that incorporates the spacing effect. Section 5 presents the real-world dataset used for the experiments. Analysis and experimental results are presented in Section 6. Finally, conclusions and final remarks are discussed in Section 7.

2. Knowledge Tracing and Memory

2.1 Memory Decay and Spacing Effect

Psychological effects influence student learning in classrooms and virtual learning environments. One such principle is the decay theory, a principle of forgetting which states that memory fades with the passage of time. Unless reinforced by repetition, the information we learn is forgotten over time. Without incorporating decay into knowledge estimation, it is difficult to account for situations where a student forgets. Memory decay has been widely studied and modeled by cognitive scientists and learning scientists (e.g. Mozer et al., 2009; Pavlik & Anderson, 2008). For example, in MCM, memory decay is incorporated through a power function in which each item-specific memory trace decays exponentially (Mozer et al., 2009). This model was then used to determine optimal practices to yield the highest retention of material in a classroom. Similarly, a study was conducted to measure and compare the relearning of forgotten material by three computational models, all of which incorporate a component of decay over time in its prediction equation (Walsh et al., 2018).

There is increasing evidence that knowledge retention can be enhanced through methods such as distributed learning, which sequences learning such that concepts are practiced in a distributed schedule over time rather than with massed schedules where practice attempts related to the same set of skills occur in quick succession. Research in this area has demonstrated that both time elapsed between practices of the same material and time elapsed between final study episode and an exam affect final-test retention (Wiseheart et al., 2006). Work has attempted to use models to determine optimal spacing, such as (Mozer et al., 2009; Pavlik & Anderson, 2008), each of which introduced a model that can predict the influence of specific study schedules on retention for specific items. Knowledge tracing algorithms incorporating spacing effects into learner models (e.g. Walsh et al., 2018; Pavlik et al., 2021) have been used for an increasing number of applications, including the adaptive sequencing of practice (Eglington & Pavlik, 2020). In total, there is considerable evidence that spacing is important for long-term retention of knowledge. Within this paper, we investigate different variations on how to represent the spacing effect in knowledge estimation, analyzing how these models perform for different representations of memory decay, through altering the rate and strength of decay over time.

2.2 Performance Factor Analysis

Performance Factor Analysis (PFA) is a knowledge tracing algorithm (Pavlik et al., 2009) that can be used with multi-skill items, in contrast to some of its extensions which only work for single-skill items (Galyardt & Goldin, 2014). PFA has been successful at predicting both knowledge within the learning system and latent knowledge carried outside the system (Scruggs et al., 2020). Our proposed algorithm and experiments use PFA as a baseline model.

The original PFA model predicts the performance of a student on a given item/problem, at a given time. It does this by using the student's past number of successes, multiplied by a weight γ fit to each of the item's skills; a student's past number of failures, multiplied by a weight ρ fit to each of the item's skills; a weight β which represents the difficulty of an item. Depending on the variant of PFA, β is either applied across all contexts, across all items linked to the current skill (the most common approach and what we will use here), across all items of the same "item-type", or for individual items. These features are inputted into a logistic function to obtain a prediction, $p(m)$, which gives the probability of success for a given student on a given (future) item. The PFA formula is given below:

$$m(i;\ j \in KC; s;\ f) = \sum_{j \in KC} (\beta_j + \gamma_j s_{i,j} + \rho_j f_{i,j}) \qquad p(m) = \frac{1}{1 + e^{-m}}$$

where i represents a student, KC are the knowledge components (i.e. skills) linked to the item, j represents a skill. Parameters β_j, γ_j and ρ_j are the learned parameters for skill j.

The PFA model in its original form does not incorporate the notion of memory decay. All previous practices are given the same weight, regardless of the time or order in which they

took place. As discussed above, some previous studies have incorporated the notion of memory decay into their knowledge tracing models. Doing so addresses the phenomenon that memory fades with the passage of time. Ignoring decay may be temporarily safe when practice sessions are massed (as in some intelligent tutoring systems) but will lead to less accurate inference when the student's work on a skill is spread out over time.

3. Related Work

A variety of knowledge tracing frameworks and models have been proposed and studied. This includes methods based on Hidden Markov Models such as Bayesian Knowledge Tracing (BKT) (Corbett & Anderson, 1995), which models a student's latent knowledge of knowledge components as a set of binary variables which represent mastery or lack of mastery of each concept, and neural network models such as Deep Knowledge Tracing (Piech et al., 2015), which uses RNNs (recurrent neural networks) to learn concept patterns using long-short term memory (LSTM) without human annotations. There are also recent efforts to combine knowledge tracing and Item Response Theory (Oeda & Asai, 2016; Khajah et al., 2014b) with a decay effect, modeled as elapsed time and a forgetting parameter.

In the last few years, extensions to these categories of models have attempted to include item recency/decay. Researchers demonstrated that by adding a forgetting parameter to BKT, the algorithm is more sensitive to the effect of interspersed trials (Khajah et al., 2016). Another variant of BKT called Multistate BKT (Argawal et al., 2020) incrementally increases the weight of newer attempts. In Attentive Knowledge Tracing (Ghosh et al., 2020), researchers implemented a monotonic attention mechanism that uses an exponential decay curve to downweight past questions. DAS3H (Choffin et al., 2019) modeled both learning and forgetting curves, using factorization machines to handle multiple skills tagging. Learning Process-consistent Knowledge Tracing (LPKT) (Shen et al., 2021) models the student learning process as a set of tuples which includes the time series information of the assignments, thereby embedding both the answer time as well as the interval between activities.

Previous research has investigated variations of PFA that incorporate decay into their mastery predictions. One such model, PFA-Decay (Gong et al., 2011), took practice order into account using a decay factor δ (0 < δ <= 1) raised to a power representing the distance in practice number, and multiplied to the counts of successes and counts of failures. Another modification of PFA, Recent-PFA (R-PFA), incorporated memory decay into the model's performance prediction in the form of a weighted proportion of success, with weight being dependent on the recency of the practice (Galyardt & Goldin, 2014). However, R-PFA does not take time specifically into account; it just considers practice order. R-PFA modifies the PFA formula by replacing the total number of failed practices, f_{ij}, with the total number of all practices thus far (essentially equaling $f_{ij} + s_{ij}$), and replacing the total number of successful practices s_{ij} with a component, R_{ijt}, that incorporates the notion of memory decay:

$$R_{ijt} = \frac{\sum_{p=-2}^{t-1} b^{(t-p)} X_{ijp}}{\sum_{p=-2}^{t-1} b^{(t-p)}}$$

where $b \in (0,1]$ represents the decay factor, and X_{ijp} represents the outcome of practice *p* (1 if successful and 0 if failed) for student *i* on skill *j*. In the original R-PFA paper, authors used three ghost/synthetic practices (Galyardt & Goldin, 2014).

Note that in the first R-PFA paper each item was linked to a single skill, losing one of the main original benefits of PFA. To the best of our knowledge, there is no previous work on extending R-PFA to create a variation which can handle data containing multi-skill items. However, recency has been considered, including adding time-based weights to components of existing models and incorporating a weighted proportion for failures (Pavlik et al., 2021).

Further work is seen in Logistic Knowledge Tracing (LKT) (Pavlik et al., 2021), a logistic regression-based framework that can enable multiple components from different existing models, including PFA (Pavlik et al., 2009), PFA-Decay (Gong et al., 2011), R-PFA (Galyardt & Goldin, 2014), and PPE (Walsh et al., 2018). LKT showcased a suite of components that could be combined to form new models, some of which incorporated the notion of recency and decay. Two of the comparison models we implement, Alg1 and Alg2 (described further in

Section 6), are based on components described in LKT. Important components from (Pavlik et al., 2021) that we used in our comparison models are as follows:

- Intercept for each KC/skill, which is a simple linear model intercept. Used in Alg1 and Alg2.
- (Alg1) Log performance (logsuc, logfail), which is the log-transformed performance factor (the total successes or failures), representing declining marginal returns, e.g. $\ln(s_{ij}) + \ln(f_{ij})$.
- (Alg2) Exponential decay of proportion, which uses the prior probability correct for each knowledge component, as in the R-PFA model. This uses a parameter to describe the exponential rate of decay, or recency, for observations of a knowledge component.
- (Alg2) The LKT version of Recency, defined as the power log decay applied to the time interval since the previous encounter with the KC. This feature considers only the just prior observation and simulates performance improvement when the prior practice was recent.

In this paper we propose a model called MemDec that captures memory decay and spacing, and compare its performance to PFA, R-PFA, and components of LKT, while accounting for multiple skills per item. We implement and analyze two methods of incorporating decay through practice order and through time windows. This proposed algorithm considers the effects of spacing between practices, thus modeling the spacing effect when predicting student performance. It also studies multiple ways of inducing decay, whereas previous studies focus only on one method, mostly practice-order. More study of the spacing effect in knowledge tracing is warranted. While a few algorithms have utilized the notion of time windows (Choffin et al., 2019; Lindsey et al., 2014), their time windows overlap and are defined in a relatively restrictive and limited manner. Additionally, (Choffin et al., 2019; Lindsey et al., 2014) do not consider time elapsed between practices when modeling spacing, whereas our proposed model MemDec Spacing considers the time elapsed between each current and previous practice, which we believe is crucial for modeling spacing accurately. Also, in MemDec, this is done for both practice-order and time-window variations. For MemDec and MemDec Spacing with time windows, we decay the weight of a practice based on the time window the practice falls into. Using equivalent, disjoint time windows allows for consistency in the exponential rate of decay through time. Additionally, we conduct an in-depth analysis on the impact of changing the decay factors and the number of ghost practices.

4. MemDec Algorithm

4.1 MemDec (PFA Memory Decay)

We propose a new model, MemDec, a variation of the PFA algorithm inspired by R-PFA components, that can also be seen as fitting within the LKT framework. In MemDec, memory decay is applied to both successful and failed practices, and the model can be used with multi-skill items (whereas the original R-PFA only supports single-skill items).

In addition to this, we build on the approach in (Maier et al., 2021), which splits skills into “common” or “rare” categories, so that it can be applied for learning systems where some skills are rare. Rare skills can occur when – for example -- items are tagged with skills that represent prerequisites not taught in the courseware. When training PFA with datasets containing rare skills, several challenges including degenerate parameters can occur (Maier et al., 2021). Depending on how rare some skills are, there might not be enough data points to precisely estimate parameters when training a model. (Maier et al., 2021) proposes a PFA variant that splits out common and rare skills. When training a model, each common skill has its own set of parameters, while all rare skills are combined into a single common set of default parameters, improving predictions and reducing model degeneracy (Maier et al., 2021).

The formula for MemDec is given below (note that as for PFA and R-PFA, m will be inputted into a logistic function to obtain a prediction, $p(m)$):

$$m(i,j,KC,RS,RF) = \sum_{j \in \, common\ KC} \left(\beta_j + \gamma_j RS_{ijt_{ij}} + \delta_j RF_{ijt_{ij}}\right) + \sum_{j \in \, rare\ KC} \left(\beta_d + \gamma_d RS_{ijt_{ij}} + \delta_d RF_{ijt_{ij}}\right)$$

$$RS_{ijt_{ij}} = \frac{\sum_{p=0}^{t_{ij}-1} b_s^{(t_{ij}-p)} X_{ijp}}{\sum_{p=0}^{t_{ij}-1} b_s^{(t_{ij}-p)}} \qquad RF_{ijt_{ij}} = \frac{\sum_{p=0}^{t_{ij}-1} b_f^{(t_{ij}-p)} (1-X_{ijp})}{\sum_{p=0}^{t_{ij}-1} b_f^{(t_{ij}-p)}}$$

Where i represents a student, j represents a skill, t_{ij} is the current trial (i.e. practice) number student i is on with skill j. X_{ijp} represents the correctness of the practice (i.e. it is 1 if student i's practice *p* with skill *j* was successful, and 0 otherwise). Constant $b_s \in (0,1]$ is the decay rate for successful practices, and constant $b_f \in (0,1]$ represents the decay factor for failed practices. RS and RF represent the recency-weighted proportions of past successes and past failures, respectively. The values for β_j, γ_j and δ_j are parameters that are learned for each skill j during the training. Parameters β_d, γ_d and δ_d are the default parameters learned for the rare skills. Only one value is learned for each parameter for the set of rare skills.

Like R-PFA, MemDec can incorporate ghost/synthetic practices (imaginary practices that improve model performance initially) in the RS and RF formulas. To allow for ghost practices, we start *p* from a negative number (instead of starting from 0). In the original R-PFA formula, the authors proposed three ghost practices (all failed practices). As in LKT, we investigate using no ghost practices, two ghost practices (one successful and one failed), and three ghost practices (all failed). The main differences between R-PFA and the MemDec base variant is that MemDec does not use a total term; instead, it contains a component that takes into consideration the weighted proportion of failed practices, giving more weight to recent ones. Also, MemDec can handle multi-skill items whereas R-PFA is designed for only single-skill items. Additional, R-PFA only considered a practice order approach, whereas one MemDec variant considers a time window approach.

In MemDec, as in most models that incorporate the notion of decay, the order of practices plays an important role. Every time the student completes a new practice, the model introduces more decay to previous practices. This means that the more practices the student has, the more decay is applied to older practices. In this approach, while decay is incremented by more practice, the time elapsed between practices does not affect the calculation of decay.

Practices can be separated by different amounts of time, from seconds to months. It is unlikely that substantial forgetting will occur with small amounts of elapsed times, such as seconds or minutes. This could be a limitation to the practice-order approach, if the amount of elapsed time can vary considerably. Thus, we propose a variation of MemDec that uses a time window, a constant duration of time (for example: 1 day) in which items answered within the same time window are given equal decay. This accounts for memory decay not occurring instantaneously. Practices answered in time windows farther from the current practice can be expected to have decayed more than practices from more recent time windows. For this variant, MemDec's RS and RF formulas become:

$$RS_{ijt_{ij}} = \frac{\sum_{p=0}^{t_{ij}-1} b_s^{timewindow(time(t_{ij})-time(p))} X_{ijp}}{\sum_{p=0}^{t_{ij}-1} b_s^{timewindow(time(t_{ij})-time(p))}}$$

$$RF_{ijt_{ij}} = \frac{\sum_{p=0}^{t_{ij}-1} b_f^{timewindow(time(t_{ij})-time(p))} (1-X_{ijp})}{\sum_{p=0}^{t_{ij}-1} b_f^{timewindow(time(t_{ij})-time(p))}}$$

4.2 MemDec Spacing

The spacing effect has to do with the temporal distribution of practices linked to the same skill. If minimal time has elapsed between practices, the learning is said to be massed. Existing research suggests that if practices are spaced out, information is retained longer in memory (Pavlik & Anderson, 2005). In this section, we investigate an extension of MemDec that incorporates the notion of the spacing effect into knowledge estimation. The MemDec model was adjusted to use b_s values that are calculated based on a formula which takes into account how spaced apart practices are. The values of b_f are calculated in a similar manner:

$$b_s(t_{ij}) = \begin{cases} b_{s_{min}}, & if\ no\ prev\ practices, or\ elapsed\ time = 0 \\ \min(b_{s_{min}} + \log_z(time(t_{ij}) - time(t_{ij} - 1)), b_{s_{max}}), & otherwise \end{cases}$$

$$b_f(t_{ij}) = \begin{cases} b_{f_{min}}, & if\ no\ prev\ practices, or\ elapsed\ time = 0 \\ \min(b_{f_{min}} + \log_z(time(t_{ij}) - time(t_{ij} - 1)), b_{f_{max}}), & otherwise \end{cases}$$

where $b_{s_{min}} \in (0,1]$ and $b_{s_{max}} \in (0,1]$ represent the min and the max values that we allow for b_s. Constants $b_{f_{min}} \in (0,1]$ and $b_{f_{max}} \in (0,1]$ represent the min and the max values that can be used for b_f. Constant z is the base of the logarithm, and expression $time((t_{ij}) - time(t_{ij} - 1))$ calculates the elapsed time between the current practice with skill *j* and the previous practice with skill *j* performed by student *i*. Note that if the student has no previous practices with a skill, or the elapsed time is 0 (elapsed time of 0 can occur in real systems if timestamps are not captured at enough granularity), then we assign the minimum values. MemDec Spacing could be used with either practice order (p.o.) or with a time window.

5. Dataset

For the experiments presented in this article, we used data from Reveal Math Course 1, a McGraw-Hill digital math product that covers grade 6 US math curriculum. The items from the assessments from this data are tagged with one or more skills. The data we used came from two Midwestern school districts and one Southwestern school district. One of these school districts is within a large U.S. city where over half of students are classified as Black, around 10% of students are classified as Hispanic, and a fifth of families live under the poverty line. A second district is within a small town where around 5% of students are classified as Black, around 90% of students are classified as White, and around 10% of families live under the poverty line. A third is within a larger town where just over half of students are classified as Hispanic, just under half of students are classified as White, and about ¾ of families live under the poverty line. All use the NGA Center/CCSSO Common Core Standards.

Extracted data spans between August 2019 and May 2021. There are 4,363 unique items; 2,009 are tagged with multiple skills. The items include multiple choice, fill in the blank, and entering equation items. Overall, the dataset had 71 unique skills which were linked to the items from the dataset. Out of these skills, 42 were classified as common (at least 200 students with at least 3 practices – Maier et al., 2021) and the remaining 29 were classified as rare.

The dataset has 489,359 datapoints. Datapoints represent students' responses and their normalized scores (1 if the response is correct, 0 if incorrect). 1.25% of the datapoints contained a partially correct score, which were treated as 0 for the purposes of this analysis. For the experiments, we split our dataset into training and testing sets. We randomly selected about 20% of the students (647 students, 98,604 data points, 64 skills) for the testing set, leaving 80% of the students (2,588 students, 390,755 data points, 71 skills) for training.

6. Experimental Results

For validation, we ran several experiments using the proposed approaches from this article, as well as other existing algorithms. For comparison reasons, we implemented the original PFA (Pavlik et al., 2009) with adjustments to handle rare skills as described in (Maier et al., 2021). We call this the Baseline model. We also implemented other algorithms to benchmark against MemDec and MemDec Spacing: R-PFA (Galyardt & Goldin, 2014), and two algorithms that were inspired by models from LKT (Pavlik et al., 2021). We provide information on those in the Comparison Models sub-section. For all models, we allowed for multi-skill items by using a summation factor across multiple skills linked to an item.

In an effort to study the differences and the effectiveness between each model, we calculated the AUC and RMSE validation metrics. Also, we present validation results for different groups of datapoints within the testing dataset: "all data" means we validated against all datapoints; "1+ non-default" means that we only used datapoints for which the item was

linked to at least one common skill; “only non-default” means we only used datapoints whose items were linked with exclusively common skills; “1+ default” means we only used datapoints whose items were linked to at least one rare skill; and “only default skills” means datapoints with items solely tagged to rare skills.

6.1 Baseline Results

We trained a model that learned three parameters for each of the 42 common skills and three parameters for the rare skills. The validation results are presented in Table 1 below:

Table 1. *Baseline PFA* and *MemDec*

Category	# of Data Points	AUC, RMSE (Baseline)	AUC, RMSE (MemDec Practice Order)
All data	98604	0.6975, 0.4443	0.7679, 0.4076
1+ non-default	97460	0.6959, 0.4446	0.7675, 0.4074
Only non-default	95859	0.6952, 0.4444	0.7677, 0.407
1+ default	2745	0.7554, 0.4417	0.7628, 0.431
Only default	1144	0.8083, 0.4163	0.8077, 0.4244

6.2 MemDec Models Results

To study the difference between methods used to govern the increase in decay, we implemented and tested a variation of MemDec that used the practice-order approach, as well as a variation that used the time-window method. For the decay factors $\boldsymbol{b_s}$ and $\boldsymbol{b_f}$ we tried several combinations of values from (0,1]. While other combinations gave similar results, the best were obtained with $\boldsymbol{b_s}$= 0.6 and $\boldsymbol{b_f}$= 0.7. We present this model’s results in Table 1. We observe a significant improvement when compared with the Baseline model. By incorporating the notion of memory decay, MemDec achieved an AUC of about 0.77 on all testing datapoints, whereas the baseline reached only an AUC of about 0.7. Significant improvements were observed in all other categories of datapoints, except for categories involving default skills, for which the two models achieved similar performance, likely because rare skills do not contain enough datapoints in the dataset for our model to substantially learn from.

We also ran experiments with a time-window of 14d (14 days), 7d, 2d, and 1d (see Table 2). The best results were observed for a 1d time window with an AUC equaling 0.756 across all datapoints from the testing set, which is slightly lower than the AUC of the practice-order model. The 2d window model obtained an AUC of 0.754 for all data, the 7d window an AUC of 0.749, and the 14d window model an AUC of 0.747. This demonstrated that for this dataset, the model that uses the practice-order approach performs slightly better. For the time-window variation we observed that the smaller the window, the better the results.

Table 2. *MemDec, with time-windows of different sizes*

Category	AUC, RMSE (1d)	AUC, RMSE (2d)	AUC, RMSE (14d)
All data	0.7561, 0.4126	0.7541, 0.4134	0.7468, 0.4166
1+ non-default	0.7557, 0.4124	0.7536, 0.4132	0.7462, 0.4164
Only non- default	0.7557, 0.4119	0.7536, 0.4127	0.746, 0.416
1+ default	0.7564, 0.4346	0.7552, 0.4343	0.7556, 0.4357
Only default	0.8053, 0.4287	0.8018, 0.4294	0.7992, 0.4311

To study whether modeling a combination of both decay and spacing could further improve the predictions, we ran experiments with the MemDec Spacing model. We experimented with different values for the hyperparameters which represents the lower and upper bounds of the decay factor, and for practice order we obtained very similar results compared to the non-spacing practice order MemDec models. By applying practice order and

parameters *($b_{s_{min}}$ = 0.55, $b_{s_{max}}$ = 0.65, $b_{f_{min}} = 0.7, b_{f_{max}} = 0.7, \log_z = \log_2$)*, shown in Table 3 left column we obtained a slightly better result than MemDec without spacing, with an overall AUC of 0.768. For this approach and time window of 1d (1 day) the AUC was 0.756, which is slightly lower than the AUC when using practice-order (p.o.). Large time windows performed more poorly still. Overall, with time windows, MemDec Spacing gave slightly poorer results than the MemDec model. This finding may be due to certain properties of the dataset we use. Many datapoints are not spaced apart more than a few seconds in time, which would cause incorporating the effects of spacing (through time windows) to have negligible effects on the model's calculation of student knowledge. When using practice order to represent decay, the effect of spacing seems to be negligible. It is possible that if a different dataset with more widely spaced practices is used, the effect of spacing on MemDec with practice order might be more beneficial. In Section 7 we will discuss the interpretation of these results further.

Table 3. *MemDec Spacing, with params ($\boldsymbol{b_{s_{min}}}$ = 0.55, $\boldsymbol{b_{s_{max}}}$ = 0.65, $\boldsymbol{b_{f_{min}} = 0.7, b_{f_{max}} = 0.7}$)*

Category	AUC, RMSE (p.o.)	AUC, RMSE (1d)	AUC, RMSE (7d)
All data	0.768, 0.4076	0.7558, 0.4127	0.7492, 0.4156
1+ non-default	0.7675, 0.4074	0.7554, 0.4125	0.7487, 0.4155
Only non- default	0.7678, 0.407	0.7554, 0.4121	0.7485, 0.4151
1+ default	0.7635, 0.4307	0.7569, 0.4347	0.7559, 0.4351
Only default	0.8087, 0.4245	0.8038, 0.4291	0.7992, 0.431

6.3 Comparison Models

We compare our models with R-PFA (Galyardt & Goldin, 2014), as well as two algorithms that incorporate components from LKT (Pavlik et al., 2021), which we call Alg1 and Alg2. The m function of each model is inputted into the sigmoid function, to get a probability value between 0 and 1. Alg1 contains a recency component that captures the elapsed time (*t*) between current and previous practice of the student *i* with skill *j* raised to a decay factor *d*. It also takes the natural logarithm of the number of successes $\boldsymbol{s_{i,j}}$ and number of failures $\boldsymbol{f_{i,j}}$. Alg2 contains a component that uses a weighted proportion of previous practices along with a parameter *b* that represents the exponential rate of decay, and ghost parameters.

Alg1: $\boldsymbol{m(i;\ j \in KC; s;\ f) = \sum_{j \in KC}(\beta_j + \gamma_j \ln(s_{i,j}) + \rho_j \ln(f_{i,j}) + \alpha_j t_{i,j}^{d})}$

Alg2: $\boldsymbol{m(i;\ j \in KC; s;\ f) = \sum_{j \in KC}\left(\beta_j + \gamma_j \frac{\sum_{p=-2}^{t-1} b^{(t-p)} X_{ijp}}{\sum_{p=-2}^{t-1} b^{(t-p)}}\right)}$

Because many of these models used two (1 failed, 1 successful) ghost practices, or three (3 failed) ghost practices, we also implemented and ran experiments with MemDec and MemDec Spacing using this combination of two or three ghost practices. For MemDec, the presence of ghost practices had a negligible influence on results. The results of R-PFA with different numbers of ghosts (successes and failures) are given in Table 4. These findings show that MemDec significantly outperformed R-PFA, regardless of the number of ghost practices. The number of ghost practices did not seem to have a major influence on the R-PFA results. Alg1 performed better than R-PFA, with an AUC of 0.7234 for all test data points. Alg2, with both two and three ghost practices, performed better than Alg1, with an overall AUC of 0.747 for two ghost practices and 0.739 for the model with three ghost practices. These results, shown in Table 5, are still worse than MemDec and MemDec Spacing.

Table 4. *R-PFA ($\boldsymbol{b_s}$= 0.6), for different success and fail ghost practices, AUC, RMSE*

Category	3 ghost (0s, 3f)	2 ghost (1s, 1f)	0 ghost
All data	0.6065, 0.4639	0.6067, 0.4638	0.6067, 0.4638
1+ non-default	0.6051, 0.4637	0.6053, 0.4636	0.6053, 0.4636
Only non- default	0.6069, 0.463	0.6071, 0.463	0.6071, 0.463
1+ default	0.574, 0.492	0.574, 0.492	0.5743, 0.4919
Only default	0.7614, 0.4802	0.7614, 0.4802	0.7615, 0.4801

Table 5. *Alg1 and Alg2, AUC and RMSE*

Category	Alg1, 0 ghost	Alg2 (1s, 1f)	0 ghost
All data	0.7235, 0.4258	0.7471, 0.4208	0.7395, 0.4227
1+ non-default	0.7223, 0.4258	0.7469, 0.4205	0.7387, 0.4226
Only non- default	0.7224, 0.4253	0.7476, 0.4199	0.7395, 0.422
1+ default	0.7492, 0.4438	0.735, 0.4509	0.7279, 0.4485
Only default	0.8234, 0.428	0.8294, 0.4489	0.8191, 0.4355

Overall, MemDec and MemDec Spacing outperformed all other models implemented in this study, including PFA, R-PFA, Alg1, and Alg2. We find that the practice-order variation of MemDec Spacing and MemDec provided the best predictions, with a minimal higher performance seen in MemDec Spacing. Both were followed by the time-window MemDec variation with a slightly more significant difference. While MemDec Spacing with time-window was outperformed by MemDec with time-window, it was still more effective than any other tested models in this experiment. The practice-order model was able to estimate student knowledge much more accurately than Baseline PFA or R-PFA. Additionally, within the MemDec variants, practice-order models were more effective than time-window models, and ghost practices had a negligible effect on performance predictions.

7. Discussion and Conclusions

In this work we studied the cognitive science concepts of memory decay and the spacing effect in the context of variants of the Logistic Knowledge Tracing framework. We created a new algorithm called MemDec which expands on R-PFA, a variation of PFA that incorporates decay. Despite the early emphasis on multi-skill items being a strength of PFA (Pavlik et al., 2009), to the best of our knowledge, there is no previous work on R-PFA or other time-involved extensions that looked at data containing multi-skill items, although components in LKT can handle multi-skill items (Pavlik et al., 2021). Addressing this limitation increases the algorithm's relevance to real-world educational systems where items are associated with multiple skills. We further expanded MemDec to capture the spacing effect in our model MemDec Spacing.

We also studied different ways of modeling decay, through the order of practices and by intervals of time elapsed between practices (time windows). To the best of our knowledge previous extensions of PFA and LKT components mostly focused on a practice-order approach. We tested whether different values of the decay factor led to improved model predictions, in all variations. Our new algorithms were compared against two comparison algorithms based on LKT components, Alg1 and Alg2, as well as against PFA and R-PFA.

The results of this study showed that MemDec and MemDec Spacing outperformed all other comparison models. Practice-order MemDec variations showed better results than time-window variations. We investigated different time window sizes, from 1d to 14d windows, and found that smaller time-windows achieved better results. The study also found that modeling decay with the spacing effect did not seem to provide an advantage over solely modeling decay. For practice-order, MemDec Spacing performed slightly better than MemDec. Across time windows, MemDec outperformed MemDec Spacing by a small amount.

However, these findings may be due to the relatively massed nature of the current dataset. Therefore, it may be valuable for future work to compare these models within datasets containing more spaced items, to determine whether the time-window approach could be beneficial over practice-order in this situation. This would also show whether incorporating the spacing effect along with decay can have a bigger impact on prediction for such datasets.

Another area of future work involves looking into how well the approaches presented perform at predicting retention long-term. Finally, future work in this area may benefit from going beyond simply assessing predictive goodness to assessing the practical implications of when instructors are told a student has mastered a skill, when in fact they have forgotten it.

Overall, the fact that MemDec and MemDec Spacing outperformed the other models highlights the importance of capturing cognitive science principles such as memory decay and spacing when modeling student knowledge and predicting future performance. The analysis

conducted also showcases the difference in model performance between increasing decay by either order or through time. The results show that the proposed models are suitable knowledge tracing approaches for real-world adaptive learning systems with multi-skill items, where the real possibility of students forgetting skills can significantly impact the results.

References

Choffin, B., Popineau, F., Bourda, Y., & Vie, J.J. (2019). DAS3H: Modeling Student Learning and Forgetting for Optimally Scheduling Distributed Practice of Skills. *Proceedings of the 12th International Conference on Educational Data Mining (EDM 2019)*. arXiv:1905.06873v1.

Corbett, A.T., & Anderson, J.R. (1995). Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. *User Modeling and User-Adapted Interaction* 4:253-278.

Eglington, L. G., & Pavlik Jr, P. I. (2020). Optimizing practice scheduling requires quantitative tracking of individual item performance. *NPJ Science of Learning, 5*(1): 15.

Galyardt, A. & Goldin, I. (2014). Recent-performance factors analysis. *Proc. 7th Int. Conf. Educational Data Mining*, 411–412.

Ghosh, A., Heffernan, N., & Lan, A.S. (2020). Context-Aware Attentive Knowledge Tracing. *KDD '20: Proc. 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*.

Gong Y., Beck J. E., & Heffernan N. T. (2011). How to construct more accurate student models: Comparing and optimizing knowledge tracing and performance factor analysis. *Int. J. Artif. Intell. Educ.*, vol. 21, pp. 27–46, Jan. 2011, http://doi:10.3233/JAI-2011-016.

Khajah, M., Huang, Y., Gonzalez-Brenes, J.P., Mozer, M.C., & Brusilovsk, M.C. (2014b). Integrating Knowledge Tracing and Item Response Theory: A Tale of Two Frameworks. *Proceedings of Workshop on Personalization Approaches in Learning Environments*, pp. 7–12, 2014.

Khajah, M., Lindsey, R.V., & Mozer, M. (2016). How Deep is Knowledge Tracing?. *Proceedings of the 9th International Conference on Educational Data Mining (EDM 2016)*.

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

Maier, C., Baker, R.S., & Stalzer, S. (2021). Challenges to Applying Performance Factor Analysis in Existing Learning Systems. *Proc. 29th International Conference on Computers in Education*.

Mozer, M.C., Pashler, H., Cepeda, N., Lindsey, R., & Vul, E. (2009). Predicting the Optimal Spacing of Study: A Multiscale Context Model of Memory. In *Advances in Neural Information Processing Systems 22*. (pp. 1321-1329).

Oeda, S., & Asai, K. (2016). Student Modeling Method Integrating Knowledge Tracing and IRT with Decay Effect. In *EKM@ EKAW*, (pp. 19–26).

Pavlik, P., & Anderson, J. (2008). Using a Model to Compute the Optimal Schedule of Practice. *Journal of Experimental Psychology Applied 14*(2):101-17. DOI: 10.1037/1076-898X.14.2.101.

Pavlik, P., Eglington, L., & Harrell-Williams, L. (2021). Logistic knowledge tracing: A constrained framework for learner modeling. *IEEE Transactions on Learning Technologies, 14*(5), 624–639.

Pavlik, P.I., Cen, H., & Koedinger, K.R. (2009). Performance Factors Analysis – A New Alternative to Knowledge Tracing. *Proc. Int'l Conference on Artificial Intelligence in Education* (pp. 531-538).

Pavlik, P.I., Jr., & Anderson, J.R. (2005). Practice and Forgetting Effects on Vocabulary Memory: An Activation-Based Model of the Spacing Effect. *Cognitive Science, 29*: 559-586.

Pelánek, R. (2017). Bayesian knowledge tracing, logistic models, and beyond: an overview of learner modeling techniques. *User Modeling and User-Adapted Interaction, 27*, 313-350.

Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L.J., & Sohl-Dickstein, J. (2015). Deep Knowledge Tracing. *Advances in Neural Information Processing Systems 28*:505-513.

Schmucker, R., Mitchell, T.M. (2022) Transferrable Student Performance Modeling for Intelligent Tutoring Systems. *Proc. 30th Int'l. Conf. on Computers in Education.*

Scruggs, R., Baker, R.S., McLaren, B.M. (2020) Extending Deep Knowledge Tracing: Inferring Interpretable Knowledge and Predicting Post System Performance. *Proceedings of the 28th International Conference on Computers in Education*.

Shen, S., Liu, Q., Chen, E., Huang, Z., Huang, W., Yin, Y., Su, Y., & Wang, S. (2021). Learning Process-consistent Knowledge Tracing. *KDD '21, August 14–18, 2021.*

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

Wiseheart, M., Pashler, H., Vul, E., Wixted, J., & Rohrer, D. (2006). Distributed Practice in Verbal Recall Tasks: A Review and Quantitative Synthesis. *Psychological Bulletin 132*(3):354-80. DOI: 10.1037/0033-2909.132.3.354.