When to stop? - Towards Universal Instructional Policies

Tanja Käser
Department of Computer Science
ETH Zurich
kaesert@inf.ethz.ch

Severin Klingler
Department of Computer Science
ETH Zurich
kseverin@inf.ethz.ch

Markus Gross
Department of Computer Science
ETH Zurich
grossm@inf.ethz.ch

ABSTRACT

The adaptivity of intelligent tutoring systems relies on the accuracy of the student model and the design of the instructional policy. Recently an instructional policy has been presented that is compatible with all common student models. In this work we present the next step towards a universal instructional policy. We introduce a new policy that is applicable to an even wider range of student models including DBNs modeling skill topologies and forgetting. We theoretically and empirically compare our policy to previous policies. Using synthetic and real world data sets we show that our policy can effectively handle wheel-spinning students as well as forgetting across a wide range of student models.

CCS Concepts

•**Applied computing** → **Computer-assisted instruction; Computer-managed instruction;** *Distance learning;*

Keywords

instructional policies, student modeling, noisy data, wheel-spinning, individualization

1. INTRODUCTION

Intelligent tutoring systems offer a customized learning experience, since they adapt the difficulty level and task selection to the knowledge of the student. This adaptivity is based on two main components: a student model predicting if a student will answer a specific task correctly and a control algorithm implementing the instructional policy. Therefore, the learning outcome depends on the design of the instructional policy as well as the accuracy of the student model.

A lot of research has be done to construct student models that are able to accurately represent student knowledge. A wide-spread approach for modeling student knowledge is Bayesian Knowledge Tracing (BKT) [10]. Traditional BKT has been improved using individualization and clustering techniques [23, 30, 31, 29, 24]. Latent factors models such as the Additive Factors Model (AFM) [5, 6] and Performance Factors Analysis (PFA) [25] are also popular for modeling student knowledge. Furthermore, dynamic Bayesian networks (DBN) [14, 13, 4, 9] have been used to model the goals, engagement states and knowledge of a student. Recently, DBNs modeling skill topologies have been introduced and were shown to outperform BKT regarding prediction accuracy [16].

Much less attention has been paid to the development of efficient instructional policies. One important task in the design of an instructional policy is the optimization of teaching sequences [21, 22, 8, 26, 20]. Another essential task is the design and analysis of algorithms deciding when to stop teaching a certain skill to a student. 'When-to-stop' policies are essential to avoid over-practice or under-practice of a skill [5, 19]. A common choice for BKT is the *mastery threshold* policy [10], which stops when the predicted probability that a student has mastered a given skill is above a certain threshold. However, the mastery threshold can be seen as a parameter that controls the frequency of false positives and false negatives (compared to true mastery) [19] and therefore the performance of the policy depends on the choice of this parameter. Furthermore, the *mastery threshold* policy is not suitable for other popular models such as AFM or PFA. Other work detected the moment at which learning occurs using machine learning techniques [2], however, no specific policy was created. *Teal* is a metric for the evaluation of student models [12], which implicitly includes a stopping criterion. This stopping criterion is threshold-dependent and was not turned into an instructional policy. The *predictive similarity* policy [28] defines the stopping criterion in a model-independent way and uses a simple functional interface. Therefore, this 'when-to-stop' policy can take any predictive student model as an input. However, as we will show later, it exhibits problems with wheel-spinning students [3] and cannot be applied to recently introduced student models [16].

In this paper, we introduce the *predictive stability* policy, a new instructional 'when-to-stop' policy inspired by [28]. This policy can take any predictive student model as input. Our policy can be applied to a wider range of student models (such as for example DBNs modeling forgetting [16]) than existing 'when-to-stop' policies [10, 28]. Furthermore, the policy is robust to noise in the data set such as wheel-spinning students. In addition we augment our policy with a success criterion, making it a 'when-mastery-is-achieved' policy. As opposed to previous mastery criteria [10], our suc-

LAK '16, April 25-29, 2016, Edinburgh, United Kingdom
© 2016 ACM. ISBN 978-1-4503-4190-5/16/04...$15.00
DOI: http://dx.doi.org/10.1145/2883851.2883961

cess criteria can be used for any probabilistic student model with a limited memory. Using synthetic data, we provide an extensive analysis of the properties of our new policies when applied to popular approaches for student modeling (AFM, PFA, BKT). We demonstrate that our new policies achieve similar results as the *predictive similarity* policy for well performing students, but that our policies are able to identify wheel-spinning students. We verify the results of our synthetic data experiments on three real-world data sets and show that the predicted behavior can be replicated. Finally, we explore the possibility to apply our policy to DBNs modeling skill topologies and forgetting. Our results demonstrate that the newly developed policies deliver meaningful results for these models.

2. STUDENT MODELS

In this section, we give a short overview of the common student models, which we later use for our experiments.

Bayesian Knowledge Tracing. BKT [10] models student learning by using one Hidden Markov Model (HMM) per skill. The latent variable L of the model is binary and indicates whether a student has mastered the skill in question. The observed variable O represents the binary task outcomes, i.e. correct or wrong answers to tasks associated with the modeled skill. The BKT model can be specified using five parameters. The emission probabilities of the model are defined by the guess probability p_G of correctly applying an unknown skill and the slip probability p_S of making a mistake when applying a mastered skill. The probability p_T of a skill changing from the unknown to the known state and the probability p_F of forgetting a previously known skill define the transition probabilities of the model. The initial probability of the model is denoted by the probability p_0 of knowing a skill a-priori. In traditional BKT, forgetting is assumed to be zero $p_F = 0$. Given a sequence of observations $O_1 = o_1, O_2 = o_2, \ldots, O_T = o_T$ the learning task amounts to estimating the five parameters by maximizing the likelihood function

$$\sum_L p(O_1, \ldots, O_T, L_1, \ldots, L_T | p_0, p_T, p_S, p_G), \quad (1)$$

where we marginalize over all the hidden states L. Maximization of the likelihood is relatively simple and is commonly performed using expectation maximization [7], brute-force grid search [1] or gradient descent [31].

Dynamic Bayesian Networks. DBNs offer the possibility to represent different skills jointly in one model [16]. The latent variables of the model are again binary and indicate whether a student has mastered a given skill. Similarly to BKT, these latent variables are inferred based on binary observations, i.e. correct or wrong answers to tasks associated with the given skill. In contrast to BKT, DBNs also model the dependencies between different skills. The number of parameters depends on the number of represented skills and on the structure of the graphical model. The parameters of the network can again be associated with guessing, slipping, learning and forgetting. By modeling the topology and dependencies between skills DBNs have been shown to outperform BKT models in prediction of the next task outcome [16]. Given a sequence of observations $O_1 = o_1, O_2 = o_2, \ldots, O_T = o_T$ the learning task amounts to estimating all initial, transition and emission parameters by maximizing the likelihood function

$$\sum_{\mathbf{L}} p(O_1, \ldots, O_T, \mathbf{L}_1, \ldots, \mathbf{L}_T | \theta), \quad (2)$$

where we again marginalize over all hidden states $\mathbf{L}$. Note that since there are multiple dependent latent states $\mathbf{L_t}$ at any time step t exact inference becomes computationally intractable. However, a convex approximation admits efficient parameter learning and provides interpretable parameter estimates [16].

Latent Factors Models. AFM and PFA model the probability as a mathematical function of latent student and skill parameters. Both methods are essentially logistic regression models with a different feature vector and a different set of latent parameters.

In AFM [5, 6] the probability of correctly solving the next task is modeled as a function of student proficiency θ_p and two skill dependent parameters, item skill difficulty β_k and learning rate λ_k. The AFM model is given as

$$P(C_t) = (1 + \exp(-(\theta_p + \sum q_{kt}(\beta_k + \lambda_k T_{pk}))))^{-1}, \quad (3)$$

where T_{pk} is the number of tasks a student p has seen for skill k. Note that AFM does not differentiate between correctly and incorrectly solved tasks for the prediction of the next task outcome.

PFA [25] extends the AFM model by differentiating between correct and incorrect past observations. To do so the learning rate for each skill is split up into a success γ_k and a failure parameter ρ_k and the probability of correctly solving the next task is given as

$$P(C_t) = (1 + \exp(-(\theta_p + \sum q_{kt}(\beta_k + \gamma_k S_{pk} + \rho_k F_{pk}))))^{-1}, \quad (4)$$

where S_{pk} and F_{pk} are the number of correct and wrong responses to tasks associated with skill k.

3. INSTRUCTIONAL POLICIES

Subsequently, we first give an overview of the *predictive similarity* policy [28], which is a so called 'when-to-stop' policy working for all widely-used student models. We then introduce our new *predictive stability* policy, a 'when-to-stop' policy, which can be applied to all standard student models as well as to more complex DBNs modeling forgetting. Furthermore, we also show that for probabilistic student models, the *predictive stability* policy can be augmented by a success criterion.

3.1 Predictive Similarity

The *predictive similarity* policy is a 'when-to-stop' policy working with any predictive student model [28]. It is based on the assumption that the training should stop, if the predicted probability that a student will give a correct response is not changing anymore. In other words the policy stops as soon as independent of whether the student gets the next task right the predicted probability will not change anymore. To put this formally, the policy proposes to stop, if

$$P(|P(C_t) - P(C_{t+1})| < \epsilon) > \delta, \quad (5)$$

where $P(C_t)$ denotes the probability of observing a correct response at time t. As shown by [28], this expression holds in the following three cases:

1. $P(C_t) > \delta \wedge |P(C_t) - P_{C|1}(t)| < \epsilon$

2. $P(\neg C_t) > \delta \wedge |P(C_t) - P_{C|0}(t)| < \epsilon$
3. $|P(C_t) - P_{C|1}(t)| < \epsilon \wedge |P(C_t) - P_{C|0}(t)| < \epsilon$

where $P_{C|0}(t) = P(C_{t+1}|\neg C_t)$ and $P_{C|1}(t) = P(C_{t+1}|C_t)$. The policy relies on an undemanding functional interface to the student model requiring three functions that are easily implementable by standard student models. The functions as well as their implementation for the different student models are summarized in Table 1. For BKT with meaningful parameters ($p_G \leq 0.5$ and $p_S \leq 0.5$), the policy is highly correlated to the *mastery threshold* policy with $\Delta = 0.95$. In this case, it is therefore equivalent to a 'when-is-mastery-achieved' policy. Although the *predictive similarity* policy functionally works with any student model that provides the interface described in Table 1, the policy fails to stop in several use cases in which Equation (5) will never be fulfilled. We will present two use cases for probabilistic models and illustrate them using (adaptations of) BKT. For both use cases, we set $\delta = 0.95$ and $\epsilon = 0.01$ as suggested by [28]. Note that for BKT with meaningful parameters, the predictive similarity policy will in most of the cases stop because the third condition is met; the first two conditions are fulfilled only for special cases as for example $P(C_t) > \delta$ can be achieved only if $p_S < 1 - \delta$. As the third condition is met only if the two curves $P_{C|0}$ and $P_{C|1}$ are converging (with a maximum distance $< 2\epsilon$), the policy fails for cases, where the curves $P_{C|1}$ and $P_{C|0}$ do not converge.

Our first use case are wheel-spinning students. According to [3], about 10% of the students training a specific skill are wheel-spinning (i.e. they will never master this skill). Depending on the model parameters, the predictions for such a student might never fulfill the stopping criteria. To illustrate this behaviour, we calculated the predictions of a BKT model with meaningful parameters ($p_0 = 0.5, p_G = 0.3, p_S = 0.2, p_T = 0.2, p_F = 0$) for an artificial student. We assumed the limit case of wheel-spinning, i.e. a student who gets all answers wrong. Figure 1 demonstrates, that the three stopping conditions are never met for the hypothetical student.

The second use case are models that rule out the possibility of convergence of $P_{C|1}$ and $P_{C|0}$, such as for examples DBNs with forgetting. To demonstrate this problem, we calculated the predictions of the simplest case of a DBN with forgetting: BKT with $p_F > 0$ for an artificial student with only correct responses. Figure 2 compares the behaviour of the policy for traditional BKT ($p_0 = 0.5, p_G = 0.3, p_S = 0.2, p_T = 0.2, p_F = 0$) to that of BKT with a small amout of forgetting ($p_F = 0.05$). Even though the hypothetical student solves all tasks correctly, the policy will never stop as the curves $P_{C|1}$ and $P_{C|0}$ are never converging.

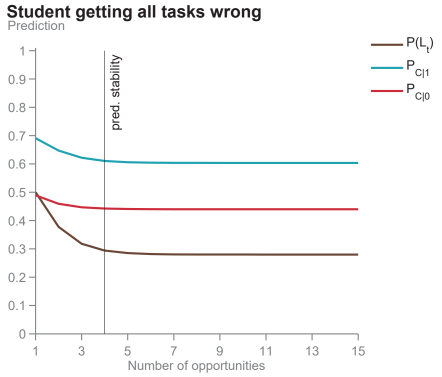

Figure 1: Probabilities $P(L_t)$, $P_{C|1}$, $P_{C|0}$ predicted by a BKT model for a student getting all answers wrong.

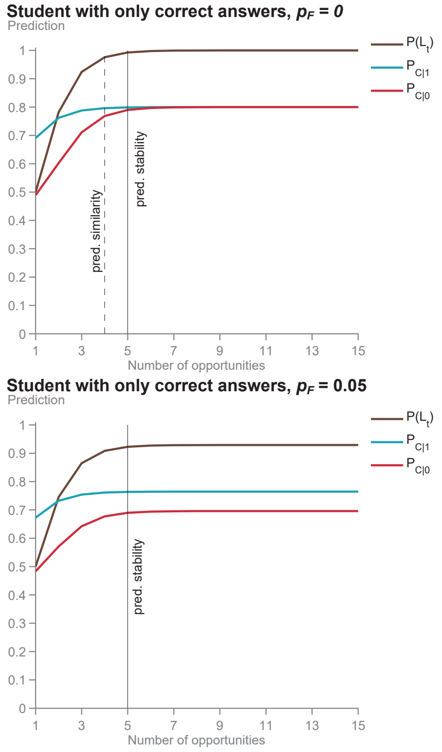

Figure 2: Probabilities $P(L_t)$, $P_{C|1}$, $P_{C|0}$ predicted by a BKT model (top) and a BKT model with forgetting (bottom) for a student getting all responses correct.

3.2 Our Policy - Predictive Stability

Inspired by the *predictive similarity* policy, we propose a new 'when-to-stop' policy, which also works for models with non converging estimates of $P_{C|1}$ and $P_{C|0}$ such as DBNs with a forgetting factor or data sets containing wheel-spinning students. Similarly to [28] we assume that every student will reach one of two end states: the student will either master a given skill or being unable to master this skill. Further, we assume that if the two estimates $P_{C|1}$ and $P_{C|0}$ individually converge to a value (not necessarily to the same value), the student has reached one of the end states. Based on theses assumptions, we propose to stop if the following expression is true

$$|P_{C|0}(t+1) - P_{C|0}(t)| < \epsilon \wedge |P_{C|1}(t+1) - P_{C|1}(t)| < \epsilon \quad (6)$$

where $P_{C|0}(t) = P(C_{t+1}|\neg C_t)$ and $P_{C|1}(t) = P(C_{t+1}|C_t)$. If the change in prediction given the student answered the current task correctly does not deviate significantly from the prediction of the task outcome given he answered the previous task correctly then we should stop. The same holds true

Table 1: Three functions are sufficient to implement the two universal 'when-to-stop' policies *predicitive similarity* [28] and *predictive stability*.

Interface	BKT	PFA	DBN
startState()	$P(L_t) \leftarrow p0$	$S \leftarrow 0, F \leftarrow 0$	$obs \leftarrow \emptyset, P(L_{ti}) \leftarrow p_{0i}$
updateState(s, o_{t-1})	$P(L_t) \leftarrow P(L_t \vert P_{t-1}, o_{t-1})$	$S \leftarrow S + \mathbf{1}_{o_{t-1}=1}, F \leftarrow F + \mathbf{1}_{o_{t-1}=0}$	$obs \leftarrow (obs, o_{t-1})$
predictCorrect(s)	$(1 - pS)P(L_t) + (1 - P(L_t))p_G$	$(1 + \exp(-(\theta_p + \beta_k + \gamma_k S + \rho_k F)))^{-1}$	$\frac{\sum_{L_i} p(obs, L_1, \ldots, L_n)}{\sum_{L_i, O_i} p(O_1, \ldots, O_n, L_1, \ldots, L_n)}$

for the prediction given we would have observed incorrect responses. We therefore call our new policy the *predictive stability* policy. Since our new policy is based only on $P_{C|1}$ and $P_{C|0}$, it relies on the same functional interface (see Table 1) as the *predictive similarity* policy and can therefore take any predictive student model as input.

To investigate the behavior of our *predictive stability* policy and to compare it to the *predictive similarity* policy we re-investigate the use cases from above (see Section 3.1). We computed predictions for artificial students on simple DBNs. For all simulations, we used $\epsilon = 0.01$. In the first example of a simulated hypothetical student getting all answers correct and a traditional BKT model with meaningful parameters ($p_0 = 0.5, p_G = 0.3, p_S = 0.2, p_T = 0.2, p_F = 0$), our policy stops after five observations (see Figure 2, top). This is comparable to the *predictive similarity* policy, which stops after four observations. When we introduce forgetting into the BKT model ($p_F = 0.05$), we observe that while $P_{C|1}$ and $P_{C|0}$ do not converge to the same value, they do converge individually (see Figure 2, bottom). While the *predictive similarity* policy will never stop, our policy stops after four observations. In the second example with the simulated wheel-spinning student (see Section 3.1), Figure 1 again demonstrates that $P_{C|1}$ and $P_{C|0}$ do converge individually, however, they again do not converge to the same value. While the *predictive similarity* policy does not stop, our policy is able to detect the wheel-spinning student after four observations.

Our *predictive stability* policy is a 'when-to-stop' policy only, i.e. it does not give any indication whether the stopped student passed the given skill. The *mastery threshold* policy that is often used for BKT is not suitable for DBNs modeling forgetting, since $P(L_t = 1)$ depends on p_F. However, popular probabilistic student models tend to have a limited memory of past observations (BKT is for example Markovian). For any set of model parameters we can therefore empirically calculate an upper bound $P_{up}(t)$ for $P(C_t)$ at time t by simulating a student that gets all answers correct. Using this upper bound $P_{up}(\cdot)$, we define a skill to be mastered at the point t if

$$S(t) \equiv |P_{up}(t) - P(C_t)| < \epsilon. \tag{7}$$

Both $P_{up}(t)$ and $S(t)$ of course depend not only on t but on the student and skill as well. However, for notational simplicity we omit this dependency in the equation.

Let us illustrate this principle of mastery by again simulating two artificial students on a BKT model with forgetting ($p_F = 0.05$). While the first student gets all answers correct, the second student commits some mistakes in the beginning. Figure 3 demonstrates that after $t = 12$ observations, the prediction for the second student reaches the upper bound $P_{up}(12)$ from the first student. For student models with limited memory, we therefore introduce the *predictive stability ++* policy, a 'when-is-mastery-achieved' policy, yielding the following output:

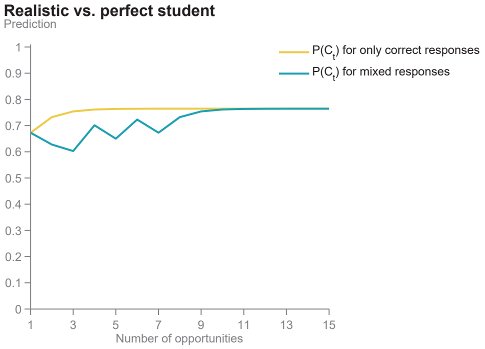

Figure 3: Predictions $P(C_t)$ of BKT for a student getting all answers correct (yellow) and a student making some mistakes in the beginning (blue).

Passed: The training of the skill stopped at time t based on the *predictive stability* policy and $S(t)$ is true.

Failed: The training of the skill stopped at time t based on the *predictive stability* policy and $S(t)$ is false.

The policy relies on the same functional interface (see Table 1) as the *predictive similarity* policy. Note, however, that the *predictive stability ++* policy is not suitable for models taking all past failures and successes into account such as PFA.

4. THEORETICAL COMPARISON OF DIFFERENT POLICIES

In this section we show that our proposed policies *predictive stability* and *predictive stability++* are comparable to the *predictive similarity* policy on the popular student models AFM, PFA and BKT. For AFM, we will derive mathematically that the *predictive stability* and the *predictive similarity* policies are actually equivalent. For BKT and PFA, we investigate the behavior of the policies on idealized simulated data. Synthetic data is more useful than real world data for our purposes, since real-world data sets are biased by the 'when-to-stop' and mastery rules used by the tutoring system. By generating synthetic data, we can study the properties of the policy without this bias, as we are able to generate student responses of arbitrary sequence length. The experiments conducted for BKT and PFA are designed to answer the following research questions. 1) *How do our policies compare to the predictive similarity policy?* 2) *What are the individual advantages and disadvantages of the re-*

spective policy? and 3) *How can students exhibiting wheel-spinning benefit from our stop criterion?*

4.1 AFM

Since the probability of a correct response in AFM for a specific student and skill only depends on the number of previous opportunities at this skill, P_{C_1} and P_{C_0} are equal:

$$P(C_{t+1}|C_t) = P(C_{t+1}|\neg C_t). \tag{8}$$

The three cases where the stopping criteria of the *predictive similarity* policy are fulfilled therefore reduce to

1. $P(C_t) > \delta \land |P(C_t) - P(C_{t+1}|C_t)| < \epsilon$
2. $P(\neg C_t) > \delta \land |P(C_t) - P(C_{t+1}|C_t)| < \epsilon$
3. $|P(C_t) - P(C_{t+1}|C_t)| < \epsilon$

Since the third condition is contained within the first two conditions, the predictive similarity policy will stop when the third condition is met. The stopping criterion for our *predictive stability* policy reduces to

$$|P(C_t) - P(C_{t+1}|C_t)| < \epsilon,$$

which is equivalent to the *predictive similarity* stopping criterion.

4.2 PFA

In the following, we show how our *predictive stability* policy compares to the *predictive similarity* policy by using a PFA model on simulated data and we highlight interesting differences in the opportunity count per student.

Experimental setup. Student responses are sampled based on PFA models with different parameter sets. To ensure that our parameters match real world conditions we generated synthetic data by sampling from BKT models, using the parameter clusters found for BKT [27]. We then learned the corresponding PFA parameters from the generated data and sampled $N = 200$ students with $T = 25$ tasks per student from the PFA models with the learnt parameters. We used the following measures (adapted from [12]) to evaluate the different policies: We define the effort E to be the number of observations until the policy stops and the score S to be the ratio of correctly solved tasks after the policy stopped.

Results. We compared effort and score as well as the percentage of students for which the policies stopped (see Table 2). Our policy stops for 99% of all students while the *predictive similarity* policy stops only for 88% of all students. On average our policy stops after $E = 4$ training opportunities compared to $E = 8$ opportunities for the *predictive similarity* policy. However, stopping earlier comes at the cost of a slightly decreased score (from $S = 0.87$ to $S = 0.84$). To further investigate the performance differences of the models we compared the effort of the *predictive similarity* policy and the *predictive stability* policy over the parameter space of the PFA model showing γ_k and ρ_k (success and failure parameters), as displayed in Figure 4. We can confirm the observation made by [28] that the *predictive similarity* policy together with PFA tends to lead to either really short training sequences (meaning a low effort, small radius) or many training opportunities (high effort, large radius). The effort for nine parameter sets is at most $E \leq 2$ and for five parameter sets we observe an effort of more than $E \geq 15$. Comparing this effort distribution to the effort values for our policy (nine parameter sets with $E \leq 2$, one parameter set with $E \geq 15$) we notice that with our policy effort values are distributed in a smaller range. This effectively reduces the "all-or-nothing" effect found with the *predictive similarity* policy [28].

Table 2: Evaluation measures for selected PFA clusters as well as for the weighted average over all clusters, comparing the *predictive similarity* policy and the *predictive stability* policy.

	$\mathbf{C_6}$	$\mathbf{C_{19}}$	**Average**
Predictive similarity			
% stopped	0.82	0.80	0.88
effort	19.40	14.30	8.20
score	0.88	0.77	0.87
Predictive stability			
% stopped	1.00	1.00	0.99
effort	4.20	4.00	4.00
score	0.80	0.72	0.84

To study this behavior, we investigated two example clusters for which the *predictive similarity* and the *predictive stability* policies exhibit high differences in effort (see Table 2). The parameters for the two clusters are as follows: C_6 ($\beta = 0.6699, \gamma = 0.0871, \rho = -0.0320$) and C_{19} ($\beta = 0.6118, \gamma = 0.0514, \rho = -0.0373$). These parameters suggest that the clusters with high differences in effort between the two policies correspond to difficult skills. While the *predictive stability* policy stops for all students, the *predictive similarity* policy stops for only part of the students (for example for 80% of the students in C_{19}). For cluster C_6, the *predictive similarity* policy stops after $E = 19.4$ observations on average, while the *predictive stability* policy shows an effort of only $E = 4.2$. Of course, this leads also to a lower average score of $S = 0.80$ compared to $S = 0.88$ for the *predictive similarity* policy. To investigate this effect even further, we split the students into a a set U_{stop} of students for whom the *predictive similarity* policy stopped and a set $U_{notstop}$ with students for whom the *predictive similarity* policy failed to stop. Table 3 shows the effort and score of the *predictive stability* policy separately for U_{stop} and $U_{notstop}$. The scores for U_{stop} achieved by the *predictive stability* policy are now closer to the scores achieved by the *predictive similarity* policy. The scores achieved for $U_{notstop}$ are low for both clusters. We therefore assume, that the *predictive similarity* policy acts as a 'when-is-mastery-achieved' policy when applied to PFA: it only stops for students mastering the skill. The *predictive stability* policy on the other hand is a pure 'when-to-stop' policy: it stops for students mastering the skill as well as for students who are not able to pass the skill.

4.3 BKT

Similar to the results for PFA we investigated the differences of the policies for idealized student data. A special focus in this investigation is our policy *predictive stability ++* that allows to decide on whether the student has reached mastery.

Experimental setup. We sampled student responses based on the BKT model for varying parameters. The parameters are based on the clusters found by [27] on real world data.

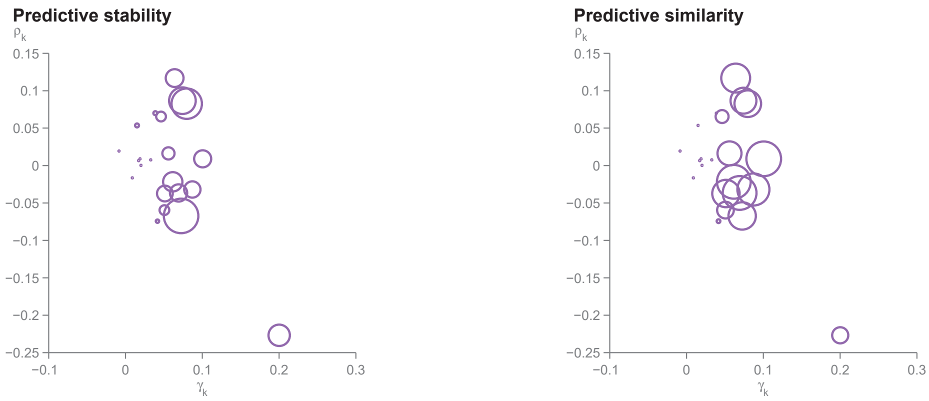

Figure 4: Effort (circle radius) of the *predictive stability* policy (left) and the *predictive similarity* policy (right) when applied to PFA models with different parameter sets. With the *predictive stability* policy, efforts are distributed in a smaller range.

Table 3: Effort and score of the *predictive stability* policy for selected PFA clusters. The measures were computed separately for U_{stop} and $U_{notstop}$.

	$\mathbf{C_6}$	$\mathbf{C_{19}}$
effort U_{stop}	4.50	4.29
score U_{stop}	0.84	0.76
effort $U_{notstop}$	2.69	2.85
score $U_{notstop}$	0.62	0.53

For every cluster we sampled student responses for $N = 200$ students with $T = 25$ tasks per student. According to [3], about 10% of students are wheel-spinning in an intelligent tutoring system, as such we generated a second data set that includes $N = 20$ wheel-spinning students (i.e. the tasks are not suitable to enable the student to achieve mastery of the skill). We simulated wheel-spinning students by setting p_T and p_0 to 0 (this means that none of these simulated students will achieve mastery of the skill) while we kept the cluster specific output probabilities p_G and p_S. As before (see Section 4.2) the effort E denotes the number of observations until the policy stops and the score S denotes the ratio of correctly solved tasks after the policy stops. In addition, the recent score RS is the ratio of correctly solved tasks over the most recent responses. It was shown that the 3-5 most recent observations are of most interest for student modeling [11], therefore we only included the three most recent responses before stopping into the calculation of RS.

Table 4: Comparison of the *predictive similarity* and *predictive stability* policies using a weighted average over all clusters for BKT for a data set with perfect students and a data set containing 10% of wheel-spinning students.

	perfect	**10% wheel-spinning**
Predictive similarity		
% stopped	0.99	0.93
effort	5.50	5.72
score	0.86	0.83
Predictive stability		
%passed	0.88	0.80
effort (passed)	6.44	6.38
score (passed)	0.86	0.85
%failed	0.11	0.18
effort (failed)	4.46	6.03
recent score (failed)	0.04	0.04

Results. Table 4 contains a summary of the comparison between the two policies on the BKT model. Effort and scores have been computed by a weighted average incorporating the size of the parameter clusters found in [27]. On the perfect data (no wheel-spinning students) both policies stop for about 99% of the students, but according to the *predictive stability ++* policy, 11% of the students have not mastered the respective skill. The average recent score of these students amounts to $RS = 0.04$. This means that most of the observations were incorrect responses and indicates that our policy is successful at discriminating students that reach mastery from those who do not. On the data set with 10% wheel-spinning students, the rate at which the predictive similarity policy is able to stop drops to 0.93 while our policy stops for 97% of the students after about six observations on average. Investigating the three most recent observations of the cases where according to our policy mastery was not achieved, we again observe very low scores. Showing all clusters together, Figure 5 confirms that both policies provide very similar values for effort and score on a wide variety of BKT parameters.

To gain a better understanding of the differences we investigated again cluster C_6 ($p_0 = 0.4, p_G = 0.47, p_S = 0.14, p_T = 0.12$) with and without wheel-spinning students. The pie chart in Figure 6, top left shows that our policy stops for all but 4% of the students for whom the *predictive similarity* policy could not stop. On the other hand our policy did not stop for only 1% of the students for whom the *predictive similarity* policy stopped. Comparing these results to the data set with wheel-spinning (Figure 6, top right) shows that the *predictive similarity* policy is not able

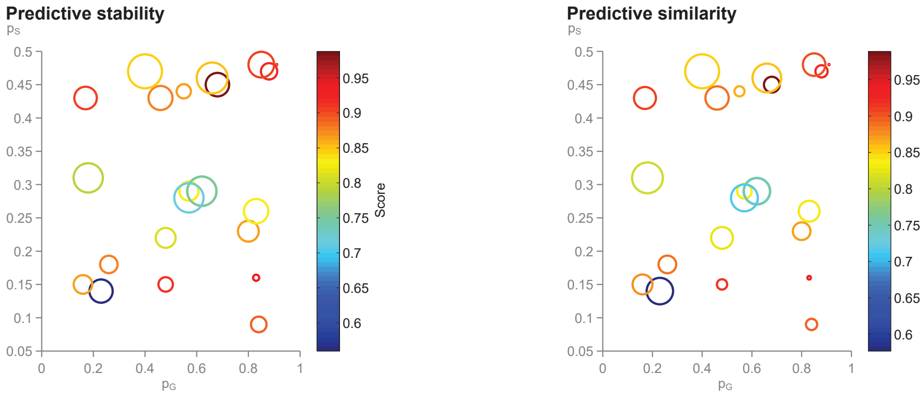

Figure 5: Evaluation measures for the *predictive stability* (left) and the *predictive similarity* policies (right) over p_G and p_S across different BKT clusters. The radius of the circle denotes the effort (growing with a larger radius) and the color the score (with red denoting higher scores).

to detect students exhibiting wheel-spinning (non-stops increased from 7.5% to 19%) whereas our policy is not able to stop in only 6.5% of the cases (increased from 4.5%). Scores for both methods in both scenarios are very comparable (Figure 6, bottom). Comparing the efforts we notice that the effort for the students that according to our policy did not master the skill is significantly lower than for students mastering the skill.

5. EXPERIMENTAL EVALUATION

We verified the performance of our *predictive stability* criterion on real-world data sets by conducting two experiments. In the first experiment, we replicate the behavior of our criterion on synthetic data (see Section 4) by evaluating it on BKT and PFA models. In the second experiment, we demonstrate that the *predictive stability ++* policy is a useful stop (and mastery) criterion for DBNs modeling skill topologies and including forgetting. For both experiments we fit the exact same skills from the real-world data sets.

Data Sets and Models. The first two data sets stem from data logs of 1581 children training with `Calcularis` [15]. `Calcularis` is an intelligent tutoring system for elementary school children with difficulties in learning mathematics. The student model used in `Calcularis` is a DBN representing different mathematical skills. We build two DBNs from this data set: The first DBN (denoted as 'Number Representation Model') contains skills training basic numerical abilities. The second DBN (denoted as 'Subtraction Model') represents subtraction skills in the number range from $0-1000$. Both models are excerpts of the skill model used in `Calcularis`. The third data set is the USNA Physics Fall 2005 data set accessed via DataShop [18]. It was collected from `Andes2` [9], an intelligent program for physics and contains data logs from 77 students of the United States Naval Academy. We used four modules of this data set to build the DBN (denoted as 'Physics Model') for the experiment.

Note that DBNs modeling skill topologies have been proposed only recently. We used the graphical models suggested in previous work for our experiments. Further details regarding the building process of our DBNs and the skills used can be found in [16].

Experimental Setup. As for the evaluation on the synthetic data sets (see Section 4.2) the effort E denotes the number of observations until the policy stops and the score S denotes the ratio of correctly solved tasks after the policy stopped. Further, the recent score RS is the ratio of correctly solved tasks in the last three observations before the policy stops. We computed all evaluation measures using student-stratified cross validation. The parameters of the DBN models were trained using latent structured prediction [17], bounding all parameters related to guessing, slipping and forgetting by 0.3. We fitted the BKT models using [31] and setting $p_G \leq 0.3$ and $p_S \leq 0.3$. The parameters of the PFA models were trained using the `lme4` package of `R`. PFA requires a student parameter (the student proficiency θ): For the unseen students in the test sets, we set θ to the mean of the trained student parameters.

Traditional Models. To verify our results on synthetic data, we evaluated our *predictive stability* policy on BKT and PFA models fit to the skills of the four DBN models. Figure 7 (top) compares the average effort computed for the *predictive stability* policy to the effort yield by the *predictive similarity* policy. Each circle denotes one skill and the colours indicate the different skill models. Both policies tend to stop within $E = 3$ observations. As expected, the two policies show a high correlation ($r = 0.73$, $p = 0.01$), with the *predictive stability* policy being slightly more conservative. Also the results for PFA (depicted int Figure 7 (bottom)) confirm our findings on the synthetic data set: The *predictive similarity* policy tends to either stop immediately or to go on for a very long time, while the *predictive stability* policy usually stops after much less observations.

Dynamic Bayesian Network Models. DBNs modeling skill topologies are a recent addition to the student model familiy, which outperformed BKT on several data sets of different learning domains [17]. We therefore evaluate our *predictive stability ++* policy on the three DBNs. Note that on all DBN models, stopping is equivalent to passing: In

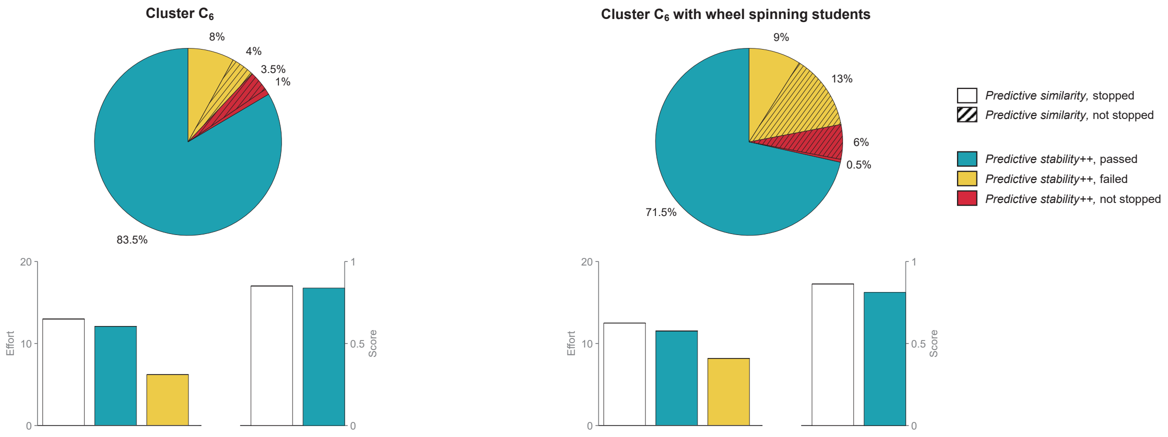

Figure 6: Percentages of students in all possible intersection sets between the decisions of the *predictive stability* (decision $\in$ {not stopped, passed, failed}) and the *predictive similarity* policies (decision $\in$ {not stopped, stopped}) for data sets without wheel-spinning (top left) and with wheel-spinning students (top right) generated from cluster C_6 ($p_0 = 0.4, p_G = 0.47, p_S = 0.14, p_T = 0.12$). Effort and score for the two policies on the same data sets (bottom).

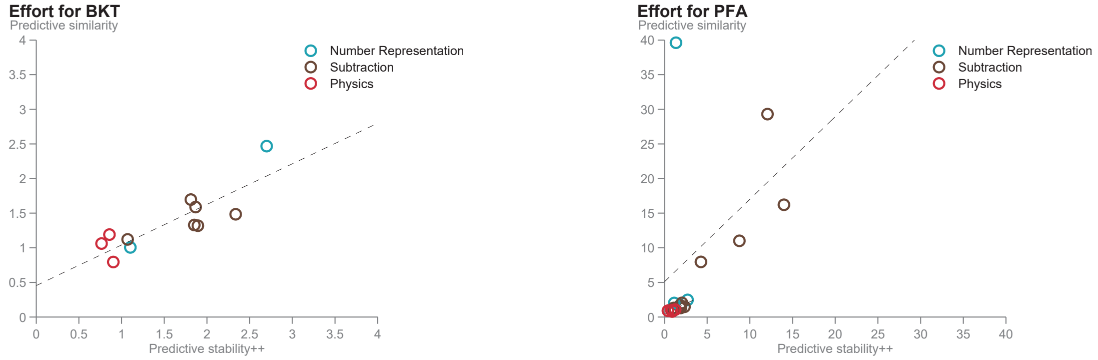

Figure 7: Effort scatter plots for BKT (left) and PFA (right) using the *predictive similarity* and the *predictive stability* policies. The colors denote the different DBN models (blue = 'Number Representation Model', brown = 'Subtraction Model', red = 'Physics Model').

all the cases where the *predictive stability ++* policy was able to stop, the students were rated as having mastered the skill. Figure 8 shows the percentage of stopped (passed) students as well as the average effort and recent score for the students who passed the respective skills.

For the 'Physics Model', the policy stopped for $94\% - 100\%$ of the students within $4 - 9$ observations on average. The recent score is also high with a minimum of $RS = 82\%$ and a maximum of $RS = 96\%$. The results on the 'Subtraction Model' are also convincing. For the skills $S_1, ..., S_5$ the policy managed to stop in 71% to 96% of the cases. The average efforts are low with a maximum of $E = 9$ for skill S_2 and also the recent scores seem reasonable with a minimum of $RS = 72\%$ again for skill S_2. For skill S_6, however, the policy managed to stop only for 6% of the students who trained this skill. Further investigations of the data set revealed that only 7% of the students trained this skill, solving on average only two tasks. Therefore, for most students, the available observation sequences were too short for the policy to stop. For the students, who passed skill S_6, the policy shows a low effort ($E = 2.5$) and a high recent score ($RS = 0.78$). The measures for skill S_7 could not be computed, since no student in the data set practiced this skill. On the 'Number Representation Model', the *predictive stability++* exhibits mixed results. Skill L_2 has a low effort ($E = 3.1$) and a high recent score ($RS = 0.95$). However, the policy stopped for only 30% of the students who practiced this skill. Although 60% of the students in the data set practiced skill L_2, data is sparse: 55% of the students have less than two practice opportunities. Skill L_1 was mastered by 91% of the students with an average effort of $E = 6.8$. The recent score for this skill is low with $RS = 0.53$. Inspections on the data set showed that almost all students practiced this skill and observation sequences tend to be long - indicating that this is a rather difficult skill. To check whether this example is an artifact of the

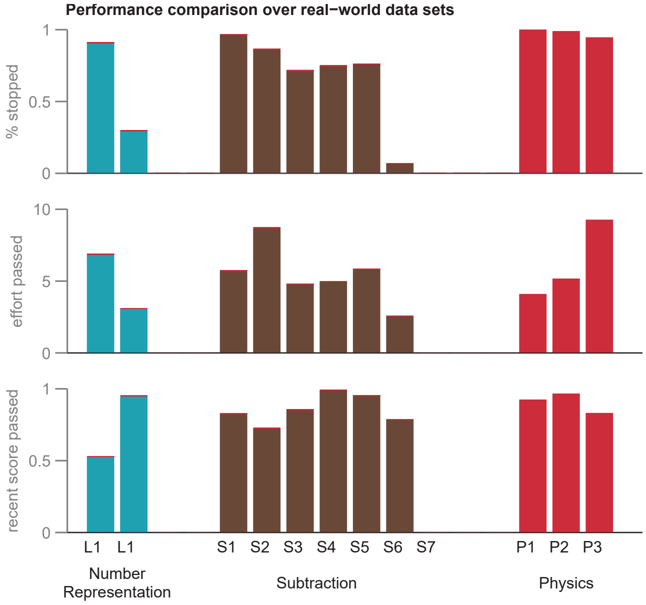

Figure 8: Performance comparison of the *predictive stability++* policy on DBN models from different data sets employing a different number of skills.

policy or the model, we examined the evaluation measures of both the *predictive similarity* policy and the *predictive stability++* policy on the BKT model for skill L_1. On this model, both policies stop for only 66% of the students with a low effort $E = 1$. The recent score, however, is high with $RS = 0.89$. Therefore, it seems that the low recent score is caused by the DBN model parameters for this skill.

6. DISCUSSION AND CONCLUSION

Instructional policies are an important aspect of tutoring systems as they influence the learning outcome. An essential part of the instructional design is the 'when-to-stop' policy, which decides when to stop teaching a certain skill to a student and therefore significantly influences the time and effort students spend on acquiring particular skills. As shown by [5] overpracticing is not necessary tied to a better performance. Instead, a better model of the learning process leads to a smaller effort without affecting the performance [5, 19]. Recently, the 'when-to-stop' policy *predictive similarity* was introduced [28]. This policy works with any predictive student model and therefore allows to compare different predictive student models not only with respect to their prediction accuracy, but also in terms of the number of practice opportunities they yield. While the *predictive similarity* policy functionally works with all common student models, we demonstrated in this work two important use cases where the stop criteria of the policy are never met. In the case of noise in the data set, i.e. students showing a behavior diverging from the model parameters (such as wheel-spinning), the policy fails to stop. Recent advances in student modeling have shown promising results regarding prediction accuracy, using DBNs modeling skill topologies and forgetting. For these models the *predictive similarity* policy is also not able to stop. In this work, we therefore introduced a new 'when-to-stop' policy that can be applied to a wider range of student models than previous policies [28, 10] including DBN modeling forgetting.

We demonstrated that for AFM models, this new policy called *predictive stability* is equivalent to the *predictive similarity* policy. By conducting experiments using simulated data from PFA and BKT models, we showed that for models with plausible parameters and no wheel-spinning students, performance of the two policies is very similar. We confirm the observations of [28] who found that the *predictive similarity* policy tends to lead to either very short or very long training sequences when applied to PFA. Results from our policy on the same data, however, suggest that the *predictive stability* policy might be more applicable to PFA models, as it circumvents the problem of extreme cases in the number of training opportunities. We furthermore demonstrated that our policy is able stop for wheel-spinning students and thus is more robust to noise in the data. For probabilistic models with a limited memory, we additionally introduced a 'when-is-mastery-achieved' policy called *predictive stability++*. Synthetic data experiments using BKT showed, that this policy can consistently identify students unable to achieve mastery of a skill. A current limitation of the *predictive stability++* policy is that it does not work for AFM and PFA. In the future, we plan to explore possibilities to adapt the 'when-is-mastery-achieved' policy to models with unlimited memory. We also conducted experiments applying PFA and BKT models on three different real-world data sets. The results of these experiments confirm our findings on synthetic data. Experiments on the same data sets using DBNs modeling the topologies of the involved skills, showed meaningful behavior compared to traditional models. However, no comparison to other policies were possible since existing 'when-to-stop' polices [28, 10] can not handle models with forgetting. To investigate the potential of the *predictive stability* policy in combination with DBNs further, we plan to investigate the performance of the policy on large scale synthetic data sets employing different skill topologies. The real-world data sets used in this work stem from mastery learning systems with an instructional policy already in place, resulting in the introduction of a bias into the data set that is hard to capture precisely. Therefore, the reported results on real world data are an approximation to the true performance of the policies.

To conclude, we presented a new instructional policy applicable to a wider range of student models than previous policies [28, 10]. We compared our new policy to existing policies theoretically as well as empirically and showed using synthetic and real world data sets that our policy effectively handles wheel-spinning and student models with forgetting.

7. ACKNOWLEDGMENTS

This work was supported by ETH Research Grant ETH-23 13-2.

8. REFERENCES

[1] R. S. Baker, A. T. Corbett, S. M. Gowda, A. Z. Wagner, B. A. MacLaren, L. R. Kauffman, A. P. Mitchell, and S. Giguere. Contextual Slip and Prediction of Student Performance after Use of an Intelligent Tutor. In *Proc. UMAP*, 2010.

[2] R. S. J. D. Baker, A. B. Goldstein, and N. T. Heffernan. Detecting Learning Moment-by-Moment. *IJAIED*, 21(1-2):5–25, 2011.

[3] J. E. Beck and Y. Gong. Wheel-spinning: Students who fail to master a skill. In *Proc. AIED*, pages 431–440, 2013.

[4] E. Brunskill. Estimating Prerequisite Structure From Noisy Data. In *Proc. EDM*, 2011.

[5] H. Cen, K. R. Koedinger, and B. Junker. Is Over Practice Necessary? -Improving Learning Efficiency with the Cognitive Tutor through Educational Data Mining. In *Proc. AIED*, 2007.

[6] H. Cen, K. R. Koedinger, and B. Junker. Comparing Two IRT Models for Conjunctive Skills. In *Proc. ITS*, 2008.

[7] K.-M. Chang, J. Beck, J. Mostow, and A. Corbett. A Bayes Net Toolkit for Student Modeling in Intelligent Tutoring Systems. In *Proc. ITS*, pages 104–113, 2006.

[8] B. Clement, D. Roy, P. Oudeyer, and M. Lopes. Multi-Armed Bandits for Intelligent Tutoring Systems. *JEDM*, 7, 2015.

[9] C. Conati, A. Gertner, and K. VanLehn. Using Bayesian Networks to Manage Uncertainty in Student Modeling. *UMUAI*, 2002.

[10] A. T. Corbett and J. R. Anderson. Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. *UMUAI*, 1994.

[11] A. Galyardt and I. Goldin. Move your lamp post: Recent data reflects learner knowledge better than older data. *JEDM*, 7(2):235–278, 2004.

[12] J. P. González-Brenes and Y. Huang. Your model is predictive - but is it useful? Theoretical and Empirical Considerations of a New Paradigm for Adaptive Tutoring Evaluation. In *Proc. EDM*, 2015.

[13] J. P. González-Brenes and J. Mostow. Dynamic Cognitive Tracing: Towards Unified Discovery of Student and Cognitive Models. In *Proc. EDM*, 2012.

[14] J. P. González-Brenes and J. Mostow. Topical Hidden Markov Models for Skill Discovery in Tutorial Data. *NIPS - Workshop on Personalizing Education With Machine Learning*, 2012.

[15] T. Käser, G.-M. Baschera, J. Kohn, K. Kucian, V. Richtmann, U. Grond, M. Gross, and M. von Aster. Design and evaluation of the computer-based training program Calcularis for enhancing numerical cognition. *Front. Psychol.*, 2013.

[16] T. Käser, S. Klingler, A. G. Schwing, and M. Gross. Beyond Knowledge Tracing: Modeling Skill Topologies with Bayesian Networks. In *Proc. ITS*, pages 188–198, 2014.

[17] T. Käser, A. G. Schwing, T. Hazan, and M. Gross. Computational Education using Latent Structured Prediction. *Proc. AISTATS*, 2014.

[18] K. Koedinger, R. Baker, A. Cunningham, K.and Skogsholm, B. Leber, and J. Stamper. A Data Repository for the EDM community: The PSLC DataShop. In C. Romero, S. Ventura, M. Pechenizkiy, and R. Baker, editors, *Handbook of Educational Data Mining*. CRC Press, Boca Raton, FL, 2010.

[19] J. I. Lee and E. Brunskill. The Impact on Individualizing Student Models on Necessary Practice Opportunities. In *Proc. EDM*, pages 118–125, 2012.

[20] T. Mandel, Y.-E. Liu, S. Levine, E. Brunskill, and Z. Popovic. Offline Policy Evaluation Across Representations with Applications to Educational Games. In *Proc. AAMAS*, pages 1077–1084, 2014.

[21] K. Muldner and C. Conati. Evaluating a decision-theoretic approach to tailored example selection. In *Proc. IJCAI*, pages 483–488, 2007.

[22] R. C. Murray, K. Vanlehn, and J. Mostow. Looking ahead to select tutorial actions: A decision-theoretic approach. *IJAIED*, 14(3-4):235–278, 2004.

[23] Z. A. Pardos and N. T. Heffernan. Modeling Individualization in a Bayesian Networks Implementation of Knowledge Tracing. In *Proc. UMAP*, 2010.

[24] Z. A. Pardos, S. Trivedi, N. T. Heffernan, and G. N. Sárközy. Clustered knowledge tracing. In *Proc. ITS*, 2012.

[25] P. I. Pavlik, H. Cen, and K. R. Koedinger. Performance Factors Analysis - A New Alternative to Knowledge Tracing. In *Proc. AIED*, 2009.

[26] A. N. Rafferty, E. Brunskill, T. L. Griffiths, and P. Shafto. Faster Teaching by POMDP Planning. In *Proc. AIED*, pages 280–287, 2011.

[27] S. Ritter, T. K. Harris, T. Nixon, D. Dickison, R. C. Murray, and B. Towle. Reducing the Knowledge Tracing Space. In *Proc. EDM*, 2009.

[28] J. Rollinson and E. Brunskill. From Predictive Models to Instructional Policies. In *Proc. EDM*, 2015.

[29] Y. Wang and J. Beck. Class vs. Student in a Bayesian Network Student Model. In *Proc. AIED*, 2013.

[30] Y. Wang and N. T. Heffernan. The student skill model. In *Proc. ITS*, 2012.

[31] M. V. Yudelson, K. R. Koedinger, and G. J. Gordon. Individualized Bayesian Knowledge Tracing Models.
In *Proc. AIED*, 2013.