Beyond Knowledge Tracing: Modeling Skill Topologies with Bayesian Networks

Tanja Käser[1], Severin Klingler[1], Alexander G. Schwing[1,2], and Markus Gross[1]

[1] Department of Computer Science, ETH Zurich, Switzerland
[2] Department of Computer Science, University of Toronto, Canada

Abstract. Modeling and predicting student knowledge is a fundamental task of an intelligent tutoring system. A popular approach for student modeling is Bayesian Knowledge Tracing (BKT). BKT models, however, lack the ability to describe the hierarchy and relationships between the different skills of a learning domain. In this work, we therefore aim at increasing the representational power of the student model by employing dynamic Bayesian networks that are able to represent such skill topologies. To ensure model interpretability, we constrain the parameter space. We evaluate the performance of our models on five large-scale data sets of different learning domains such as mathematics, spelling learning and physics, and demonstrate that our approach outperforms BKT in prediction accuracy on unseen data across all learning domains.

Keywords: Bayesian networks, parameter learning, constrained optimization, prediction, Knowledge Tracing

1 Introduction

Intelligent tutoring systems (ITS) are successfully employed in different fields of education. A key feature of these systems is the adaptation of the learning content and the difficulty level to the individual student. The selection of problems is based on the estimation and prediction of the student's knowledge by the student model. Therefore, modeling and predicting student knowledge accurately is a fundamental task of an intelligent tutoring system.

Current tutoring systems use different approaches to assess and predict student performance. Two of the most popular approaches for estimating student knowledge are performance factors analysis [20] and Bayesian Knowledge Tracing (BKT) as presented by Corbett and Anderson [4].

As the prediction accuracy of a probabilistic model is dependent on its parameters, an important task when using BKT is parameter learning. Recently, the prediction accuracy of BKT models has been improved using clustering approaches [19] or individualization techniques, such as learning student- and skill-specific parameters [18, 24, 25] or modeling the parameters per school class [23].

Exhibiting a tree structure, BKT allows for efficient parameter learning and accurate inference. However, tree-like models lack the ability to represent the hierarchy and relationships between the different skills of a learning domain. Employing dynamic Bayesian network models (DBN) has the potential to increase the representational power of the student model and hence further improve prediction accuracy. In ITS, DBNs have been used to model and predict students'

performance [3, 17] engagement states [2, 9] and goals [3]. DBNs are also employed in user modelling [8]. In cognitive sciences, DBNs are applied to model human learning [5] and understanding [1]. Despite their beneficial properties to represent knowledge, DBNs have received less attention in student modeling as they impose challenges for learning and inference.

Recently, [12] showed that a constrained latent structured prediction approach to parameter learning yields accurate and interpretable models. Based on these findings, this paper proposes the use of DBNs to model skill hierarchies within a learning domain. Similar to [1], we use a log-linear formulation and apply a constrained optimization to identify the parameters of the DBN. We define domain-specific DBN models for five large-scale data sets from different learning domains, containing up to 7000 students. Students' age ranges from elementary school to university level. Our results show that even simple skill hierarchies lead to significant improvements in prediction accuracy of up to 10% over BKT across all learning domains. By using the same constraints and parameterizations for all experiments, we also demonstrate that basic assumptions about learning hold across different learning domains and thus our approach is easy to use.

2 Methods

Subsequently, we first give an overview of the BKT model before discussing more complex graphical models that are able to represent skill topologies.

2.1 Bayesian Knowledge Tracing

BKT models are a special case of DBNs [21] or more specifically of Hidden Markov Models (HMM), consisting of observed and latent variables. Latent variables represent student knowledge about one specific skill and are assumed to be binary, *i.e.*, a skill can either be mastered by the student or not. They are updated based on the correctness of students' answers to questions that test the skill under investigation, hence observations are also binary.

There are two types of parameters in an HMM: transition probabilities and emission probabilities. In BKT, the emission probabilities are defined by the slip probability p_S of making a mistake when applying a known skill and the guess probability p_G of correctly applying an unknown skill. The transition probabilities are described by the probability p_L of a skill transitioning from unknown to known state, while p_F is the probability of forgetting a previously known skill. In BKT, p_F is assumed to equal zero. The last parameter required to describe the BKT model is the initial probability p_0 of knowing a skill a-priori.

Employing one BKT model per skill, the learning task amounts to estimating the parameters given some observations: given a sequence of observations $\mathbf{y_m} = (y_{m,1}, ..., y_{m,T})$ with $y_{m,t} \in \{0, 1\}$ and time $t \in \{1, \dots, T\}$ for the m-th student with $m \in \{1, \dots, M\}$, what are the parameters $\theta = \{p_0, p_L, p_F, p_S, p_G\}$ that maximize the likelihood $\prod_m p(\mathbf{y_m} \mid \theta)$ of the available data.

2.2 Dynamic Bayesian Networks

When employing DBNs, we consider the different skills of a learning domain jointly within a single model. Student knowledge is again represented using bi-

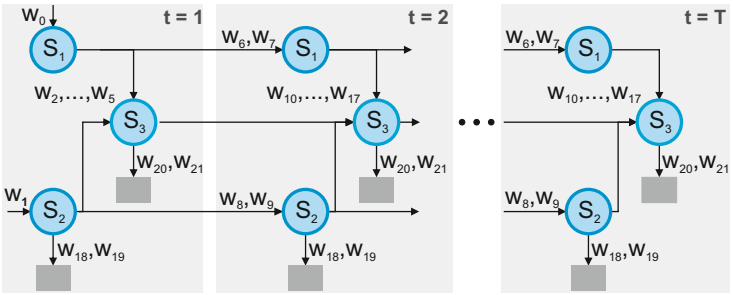

Fig. 1. Structure of the graphical model for a DBN with T time steps. Circular nodes represent skills, while the rectangles are the tasks associated with those skills.

nary latent variables (one per skill), which are updated based on observations associated with the skill under investigation. However, we now also model the dependencies between the different skills, *e.g.*, two skills S_A and S_B are conditionally dependent if S_A is a prerequisite for mastering S_B.

Probabilistic Notation. The learning task of a DBN model is described as follows: let the set of N variables of the model be denoted by $X = \{X_i \mid i \in \{1, \ldots, N\}\}$. In addition, let $\mathcal{H}$ denote the domain of the unobserved variables, *i.e.*, missing student answers and the binary skill variables, while $\mathcal{Y}$ refers to the observed space, disjoint from the latent space $\mathcal{H}$. During learning, we are interested in finding the parameters θ that maximize the likelihood of the observed data $\bigcup_m \mathbf{y_m}$ with $\mathbf{y_m} = (y_{m,1}, ..., y_{m,T})$ representing a sequence of T binary answers from the m-th student. The log-likelihood of a DBN [6] is then given by

$$L(\theta) = \sum_m \ln \left(\sum_{\mathbf{h_m}} p(\mathbf{y_m}, \mathbf{h_m} \mid \theta) \right), \tag{1}$$

where we marginalize over the states of the latent variables $\mathbf{h_m}$ for student m. The joint probability $p(\mathbf{y_m}, \mathbf{h_m} \mid \theta)$ of the model for student m is defined as

$$p(\mathbf{y_m}, \mathbf{h_m} \mid \theta) = \prod_i p(X_{m,i} = x_{m,i} \mid pa(X_{m,i}) = \mathbf{x_{m,pa(X_{m,i})}}) = \prod_i p_{ij_{m,i}\mathbf{k_{m,i}}}, \tag{2}$$

where $pa(X_{m,i})$ are the parents of $X_{m,i}$, while $x_{m,i}$ and $\mathbf{x_{m,pa(X_{m,i})}}$ are the realizations of the random variables $X_{m,i}$ and $pa(X_{m,i})$, *i.e.*, the states assigned to $X_{m,i}$ and $pa(X_{m,i})$ given by $(\mathbf{y_m}, \mathbf{h_m})$. Furthermore, we let $j_{i,m} := x_{m,i}$ and $\mathbf{k_{m,i}} := \mathbf{x_{m,pa(X_{m,i})}}$ to simplify the notation. Therefore, $p_{ij_{m,i}\mathbf{k_{m,i}}}$ denotes exactly one entry in the conditional probability table (CPT) of $X_{m,i}$.

Log-linear models. The log-likelihood of a DBN can alternatively be formulated using a log-linear model. This formulation is flexible and predominantly used in recent literature [16, 22]. Therefore, we reformulate the learning task in the following. Let $\boldsymbol{\phi} : \mathcal{Y} \times \mathcal{H} \to \mathbb{R}^F$ denote a mapping from the latent space $\mathcal{H}$ and the observed space $\mathcal{Y}$ to an F-dimensional feature vector. The log likelihood from Eq. (1) can then be reformulated to

$$L(\mathbf{w}) = \sum_m \ln \left(\sum_{\mathbf{h_m}} \exp\left(\mathbf{w}^\top \boldsymbol{\phi}(\mathbf{y_m}, \mathbf{h_m}) - \ln(Z) \right) \right), \tag{3}$$

where Z is a normalizing constant and $\mathbf{w}$ denotes the weights of the model. Next, we show that this log-linear formulation of the log-likelihood is equivalent to the traditional notation. Comparing Eq. (3) to Eq. (1), it follows that

$$\prod_i p_{ij_{m,i}\mathbf{k}_{\mathbf{m},\mathbf{i}}} = \frac{1}{Z}\exp \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{y_m}, \mathbf{h_m}) = \frac{1}{Z}\exp \sum_i w_i^\top \phi_i(\mathbf{y_m}, \mathbf{h_m}), \tag{4}$$

and therefore

$$\forall i, j, \mathbf{k} : p_{ij\mathbf{k}} = \frac{1}{Z}\exp w_i^\top \phi_i(\mathbf{x}), \tag{5}$$

where $\mathbf{x}$ are the realizations of all random variables in X with $j \in \mathbf{x}$ and $\mathbf{k} \subset \mathbf{x}$. A feature vector $\boldsymbol{\phi}$ and weights $\mathbf{w}$ that fulfill Eq. (5) can be specified as follows: consider the CPT describing the relationship between a node X_A and its $n-1$ parent nodes $pa(X_A)$. The CPT for these n nodes contains 2^n entries. Let $\mathbf{k} \in \{0,1\}^{n-1}$ denote one possible assignment of states to the parent nodes $pa(X_A)$. We can therefore define $p(X_A = 1 \mid pa(X_A) = \mathbf{k}) = 1 - p(X_A = 0 \mid pa(X_A) = \mathbf{k}) = 1 - p_{A,0,\mathbf{k}}$. To continue, let $p_{A,x_A,\mathbf{k}} = \frac{1}{Z}\exp w_{A,\mathbf{k}}(1-2x_A) = \exp w_{A,\mathbf{k}}(1-2x_A)/(\exp w_{A,\mathbf{k}} + \exp(-w_{A,\mathbf{k}}))$, which leads to the feature function $\phi_A(x) = 1 - 2x_A$. We therefore obtain the joint distribution as a product of the exponential terms which translates to a weighted linear combination of feature vector entries in the exponent and thus fulfills Eq. (5). From this formulation also follows that we need 2^{n-1} parameters to specify a CPT including n skills.

Optimization. In contrast to HMMs, the learning task for DBNs is not computationally tractable. However, [22] showed that a convex approximation admits efficient parameter learning. Note that interpretability of the parameters is not ensured, since guarantees exist only for converging to a local optimum. Recently, [12] extended the approach presented by [22] to include constraints on parameters and demonstrated that the constrained optimization increases prediction accuracy on unseen data while yielding interpretable models. Using the log-linear formulation, the algorithm presented in [12] can be directly applied to learn the parameters of a DBN model.

DBN Specification. Next, we illustrate the specification of a simple DBN. Similarly to BKT, we can interpret the parameters of a DBN in terms of a learning context. To specify the CPTs of the example DBN in Fig. 1, we employ $F = 22$ weights that can be associated with a parameter set θ. We subsequently use $\simeq$ to denote proportionality in the log domain; *i.e.*, $w \simeq p$ is equivalent to $w \propto \exp p$. Let O_3 denote the task associated with skill S_3. Then the parameters $w_{20} \simeq p(O_3 = 0 \mid S_3 = 0) = 1 - p_G$ and $w_{21} \simeq p(O_3 = 0 \mid S_3 = 1) = p_S$ represent the guess and slip probabilities. Similarly, w_{18} and w_{19} are associated with p_G and p_S as evident from Fig. 1. Furthermore, parameters $w_6 \simeq p(S_{1,t} = 0 \mid S_{1,t-1} = 0) = 1 - p_L$ and $w_7 \simeq p(S_{1,t} = 0 \mid S_{1,t-1} = 1) = p_F$ are associated with learning and forgetting; the same holds true for w_8 and w_9.
Skills S_1 and S_2 are prerequisites for knowing skill S_3, *i.e.*, the probability that skill S_3 is mastered in time step t depends not only on the state of skill S_3 in the previous time step, but also on the states of S_1 and S_2 in the current time step. Therefore $w_{10} \simeq p(S_{3,t} = 0 \mid S_{3,t-1} = 0, S_{1,t} = 0, S_{2,t} = 0) = 1 - p_{L0}$, where p_{L0}

denotes the probability that the student learns S_3 despite not knowing S_1 and S_2. Also, $w_{17} \simeq p(S_{3,t} = 0 \mid S_{3,t-1} = 1, S_{1,t} = 1, S_{2,t} = 1) = p_{F1}$, the probability of forgetting a previously learnt skill. Furthermore, we set $w_l \simeq 1 - p_{LM}$ if $l \in \{11, 12, 13\}$ and $w_l \simeq 1 - p_{FM}$ if $l \in \{14, 15, 16\}$, where p_{LM} denotes the probability that the student learns S_3 given that he knows at least one of the precursor skills of S_3. Moreover, p_{FM} is the probability that the student forgets the previously known skill S_3, when either S_1 or S_2 or none of them are known. Finally, the parameters w_l with $l \in \{2, 3, 4, 5\}$ describe the dependencies between the different skills. We let $w_l \simeq 1 - p_{P0}$, if $l \in \{2, 3, 4\}$ and $w_5 \simeq p_{P1}$, where p_{P0} is the probability of knowing a skill despite having mastered only part of the prerequisite skills and p_{P1} denotes the probability of failing a skill given that all precursor skills have been mastered already. Moreover, we refer to the probability of knowing a skill a-priori via p_0. Note that w_0 and w_1 are associated with p_0. The example DBN can therefore be described by the parameter set $\theta = \{p_0, p_G, p_L, p_F, p_{L0}, p_{F1}, p_{LM}, p_{FM}, p_{P0}, p_{P1}\}$. Importantly, the method proposed in this work is independent of the exact parametrization used. Therefore, the parametrization introduced here could be easily extended.

3 Results and Discussion

We show the benefits of DBN models with higher representational power on five data sets from various learning domains. The data sets were collected with different tutoring systems and contain data from elementary school students up to university students. We compare the prediction accuracy of DBNs modeling skill topologies with the performance of traditional BKT models.

Fitting the BKT models was done using [25], applying skill-specific parameters and using gradient descent for optimization. As described in [25], we set the forget probability p_F to 0, while p_S and p_G are bounded by 0.3. In the following, we will denote this constrained BKT version as BKT_C.

We used constrained latent structured prediction [12] to learn the parameters of the DBNs. All models are parametrized according to Sec. 2.2 and we impose the constraints described in the following on the parameter set θ of the different models to ensure interpretable parameters. For our first constraint set $\mathcal{C}_1$, we let $p_D \leq 0.3$ for $D \in \{G, S, L, F, L0, F1\}$ to ensure that parameters associated with guessing, slipping, learning and forgetting remain plausible. The constraints on θ can be directly turned into constraints on $\mathbf{w}$. For the example DBN (Fig. 1), the constraints translate into the following linear constraints on the weights for $\mathcal{C}_1$: $w_i \geq 0.4236$, if $i \in \{6, 8, 10, 18, 20\}$ and $w_i \leq -0.4236$, if $i \in \{7, 9, 17, 19, 21\}$. For the second constraint set $\mathcal{C}_2$, we augment $\mathcal{C}_1$ by limiting $p_D \leq 0.3$ if $D \in \{LM, FM, P0, P1\}$, yielding $w_i \geq 0.4236$, if $i \in \{2, 3, 4, 11, 12, 13\}$ and $w_i \leq -0.4236$, if $i \in \{5, 14, 15, 16\}$ for the example DBN (Fig. 1). The additional constraints ensure that parameters are consistent with the hierarchy assumptions of the model. The constraint sets $\mathcal{C}_3$ and $\mathcal{C}_4$ bound the same parameters as $\mathcal{C}_1$ and $\mathcal{C}_2$, but are more restrictive by replacing 0.3 by 0.2. Note that constraints were selected according to previous work [4]. The presented work is, however, independent of the selected constraint sets.

Prediction is performed as follows: we assume the observation at time $t = 1$

to be given and predict the outcome at time t with $t \in \{2, ..., T\}$ based on the previous $t-1$ observations. The number of observations T for the different experiments is the minimal number of observations covering all skills of the according experiment. To assess prediction accuracy, we provide the following error measures: Root mean squared error (RMSE), classification error CE (ratio of incorrectly predicted student successes and failures based on a threshold of 0.5) and the area under the ROC curve (AUC). All error measures were calculated using cross-validation. Statistical significance was computed using a two-sided t-test, correcting for multiple comparisons (Bonferroni-Holm).

Note that we selected skills, where users showed low performance for our experiments, in order to make learning and prediction more challenging. In the following, we describe the DBN models for the five data sets and discuss the prediction accuracy for our models as well as for BKT_C.

Number representation. For the first experiment, we use data collected from `Calcularis`, an intelligent tutoring system for elementary school children with math learning difficulties [10]. The data set contains log files of 1581 children with at least 5 sessions of 20 minutes per user. `Calcularis` represents student knowledge as a DBN consisting of different mathematical skills [11, 13].
The graphical model used in this experiment (see Fig. 1) is an excerpt of the skill model of `Calcularis` described in [11]. Skill S_1 represents knowledge about the Arabic notation system. `Calcularis` does not contain any tasks associated with this skill. The ability to assign a number to an interval is denoted by S_2. The task associated with this skill is to guess a number in as few steps as possible. Finally, S_3 denotes the ability to indicate the position of a number on a number line. We used a maximum of $T = 100$ observations per child for learning and prediction and specified the CPTs of the graphical model with $F = 22$ weights. Prediction errors for the constraint sets $\mathcal{C}_1$ to $\mathcal{C}_4$ as well as BKT_C are given in Tab. 1. The constrained DBN approach yields significant and large improvements in prediction accuracy compared to BKT_C. We highlight the improvement in accuracy by 11.4% ($\text{CE}_{BKT_C} = 0.3141$, $\text{CE}_{\mathcal{C}_2} = 0.2783$) and the reduction of the RMSE by 3.8% ($\text{RMSE}_{BKT_C} = 0.4550$, $\text{RMSE}_{\mathcal{C}_4} = 0.4378$). Also note the large improvement achieved in AUC ($\text{AUC}_{BKT_C} = 0.5975$, $\text{AUC}_{\mathcal{C}_2} = 0.7093$).

Subtraction. The second experiment is based on the same data set as the first experiment. This time, however, we investigate subtraction and number understanding skills. The graphical model (see Fig. 2(a)) is again an excerpt of the skill model [11] of `Calcularis`. Subtraction skills are ordered according to their difficulty, which is determined by the magnitude of involved numbers, task complexity and the means allowed to solve a task. Skills S_1 (*e.g.*, 48-6=?), S_2 (*e.g.*, 48-9=?), S_3 (*e.g.*, 48-26=?), S_4 (*e.g.*, 48-29=?) and S_5 denote subtraction tasks in the number range 0−100. We emphasize that there are no observation nodes associated with S_1 and S_5. The number understanding skill S_6 represents knowledge about the relational aspect of number (number as a difference between other numbers) in the number range 0−1000. Finally, skills S_7 (*e.g.*, 158-3=?), S_8 (*e.g.*, 158-3=?) and S_9 (*e.g.*, 158-9=?) represent subtraction in the number range 0−1000. The difference between S_7 and S_8 lies in the means allowed to

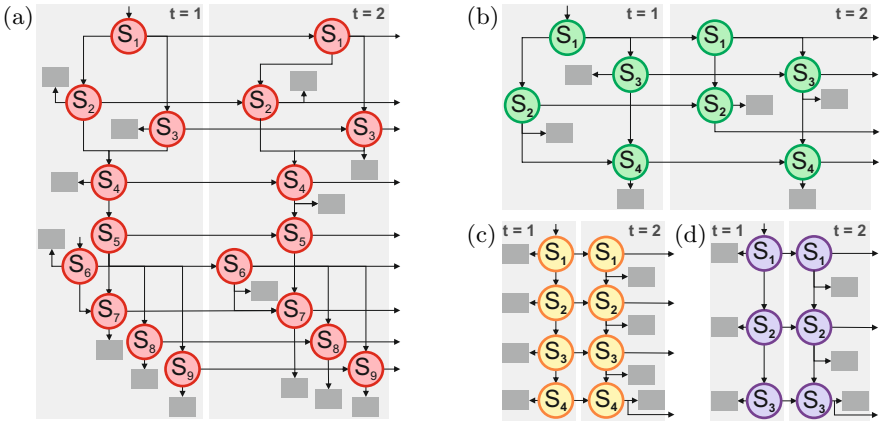

Fig. 2. Graphical models for the subtraction (a), physics (b), algebra (c) and spelling learning (d) experiments. Circular nodes represent skills, while the rectangles are the tasks associated with those skills.

solve the task. A maximum of $T = 100$ observations per child is used for learning and prediction. Specification of the CPTs for the model requires $F = 86$ weights. The resulting prediction accuracy for this experiment (see Tab. 1) again demonstrates that the DBN model outperforms BKT_C. With a reduction of the RMSE by 3.5% ($\text{RMSE}_{BKT_C} = 0.4368$, $\text{RMSE}_{\mathcal{C}_2} = 0.4215$) and an increase of the accuracy by 8.4% ($\text{CE}_{BKT_C} = 0.2818$, $\text{CE}_{\mathcal{C}_4} = 0.2580$), improvements confirm the results observed in the first experiment. Also the growth in AUC ($\text{AUC}_{BKT_C} = 0.5996$, $\text{AUC}_{\mathcal{C}_4} = 0.6916$) is again substantial.

Physics. This experiment is based on the 'USNA Physics Fall 2005' data set accessed via DataShop [15]. Data originate from 77 students of the United States Naval Academy and were collected from `Andes2`, an intelligent tutoring system for physics [3]. The tutor uses rule-based algorithms to build solution graphs that identify all possible solutions to the different tasks. Based on these graphs, a Bayesian network is constructed to assess the general physics knowledge of the student as well as the progress for the problem at hand.

We use the different modules of the data set as skills for our experiment. The graphical model is depicted in Fig. 2(b). Note that we intentionally use a simplified skill model to avoid introducing incorrect assumptions and to assess if even non-experts can exploit skill structures using our proposed methods. The model consists of the following modules: "Vectors" (S_1), "Translational Kinematics" (S_2), "Statistics" (S_3) and "Dynamics" (S_4). These modules consist of more complex tasks for the given topic, *i.e.*, calculating total forces in a system (see example in [3]). A maximum of $T = 500$ observations per child are considered for learning and prediction and the model is described by $F = 33$ weights.

In this experiment, the benefits of the DBN model are again high (see Tab. 1): the accuracy is increased by 10.7% ($\text{CE}_{BKT_C} = 0.2930$, $\text{CE}_{\mathcal{C}_4} = 0.2616$) while the RMSE is reduced by 6.3% ($\text{RMSE}_{BKT_C} = 0.4530$, $\text{RMSE}_{\mathcal{C}_4} = 0.4244$) and the AUC grows to 0.7007 ($\text{AUC}_{BKT_C} = 0.5039$).

Table 1. Prediction accuracy of the experiments, comparing BKT_C with different constraint sets for the DBNs. Numbers in bold denote a significant improvement compared to BKT_C. The best result for each error measure is marked (*).

		BKT_C	$\mathcal{C} = \mathcal{C}_1$	$\mathcal{C} = \mathcal{C}_2$	$\mathcal{C} = \mathcal{C}_3$	$\mathcal{C} = \mathcal{C}_4$
Number representation	RMSE	0.4550	**0.4469**	**0.4452**	**0.4416**	**0.4378***
	CE	0.3141	0.3279	**0.2783***	0.3079	**0.2831**
	AUC	0.5975	**0.7072**	**0.7093***	**0.7087**	**0.7049**
Subtraction	RMSE	0.4368	0.4417	**0.4215***	0.4389	**0.4216**
	CE	0.2818	0.2812	**0.2588**	**0.2757**	**0.2580***
	AUC	0.5996	**0.6157**	**0.6870**	**0.6332**	**0.6916***
Physics	RMSE	0.4530	0.4521	**0.4272**	**0.4465**	**0.4244***
	CE	0.2930	**0.2893**	**0.2677**	**0.2870**	**0.2616***
	AUC	0.5039	**0.6511**	**0.6971**	**0.6795**	**0.7007***
Algebra	RMSE	0.3379	**0.3335**	**0.3254***	**0.3321**	**0.3267**
	CE	0.1461	0.1466	**0.1392**	0.1466	**0.1379***
	AUC	0.5991	**0.6682**	**0.7004**	**0.6718**	**0.7007***
Spelling	RMSE	0.4504	0.4521	**0.4495**	**0.4492**	**0.4472***
	CE	0.2898	0.2893	0.2914	0.2882*	0.2906
	AUC	0.5029	**0.5695**	**0.5771**	**0.5735**	**0.5804***

Algebra. For this experiment we used data from the KDD Cup 2010 Educational Data Mining Challenge (*http://pslcdatashop.web.cmu.edu/KDDCup*). The data set contains log files of 6043 students that were collected by the `Cognitive Tutor` [14], an intelligent tutoring system for mathematics learning. The student model applied in this system is based on BKT.

We use the units of the 'Bridge to Algebra' course as skills for our experiment and select four units of increasing difficulty, where students have to solve word problems involving calculations with whole numbers. The graphical model for this experiment is illustrated in Fig. 2(c). Skill S_1 (*e.g.*, $728624 - 701312$) denotes written addition and subtraction tasks without carrying/borrowing, while S_2 involves carrying/borrowing (*e.g.*, $728624 - 703303$). S_3 (*e.g.*, 33564×18) and S_4 (*e.g.*, $10810 \div 46$) represent long multiplications and divisions. Note that the skill model is again simplified for the reasons explained in the Physics experiment. We use a maximum of $T = 500$ observations per student for learning and prediction and specify the CPTs of the model employing $F = 29$ weights.

Similarly to the previous experiments, DBN significantly outperforms BKT_C (see Tab. 1). The RMSE is reduced by 3.7% ($\text{RMSE}_{BKT_C} = 0.3379$, $\text{RMSE}_{\mathcal{C}_2} = 0.3254$), while accuracy is increased by 5.6% ($\text{CE}_{BKT_C} = 0.1461$, $\text{CE}_{\mathcal{C}_4} = 0.1379$) and the AUC increases to 0.7007 ($\text{AUC}_{BKT_C} = 0.5991$). Note that DBN and BKT_C both perform better than in the other experiments as the high performance of students in the involved skills makes learning and prediction easier.

Spelling learning. The last experiment uses data collected from `Dybuster`, an intelligent tutoring system for elementary school children with dyslexia [7]. The data set at hand contains data of 7265 German-speaking children. `Dybuster` groups the words of a language into hierarchically ordered modules with respect to their frequency of occurrence in the language corpus as well as a word diffi-

culty measure. The latter is computed based on the word length, the number of dyslexic pitfalls and the number of silent letters contained in the word.
We use these modules as skills to build our graphical model (see Fig. 2(d)). Skills S_1, S_2 and S_3 denote the modules 3, 4 and 5 within `Dybuster`. Word examples are "warum" ("why", S_1), "Donnerstag" ("Thursday", S_2) and "Klapperschlange" ("rattlesnake", S_3). We use a maximum of $T = 200$ observations per child for the learning and prediction tasks and parametrize the model using $F = 21$ weights. While the DBN model still significantly outperforms BKT_C in this experiment (see Tab. 1), the magnitudes of improvement are small: the RMSE is reduced by 0.7% ($\text{RMSE}_{BKT_C} = 0.4504$, $\text{RMSE}_{\mathcal{C}_4} = 0.4472$), the highest AUC amounts to 0.5804 ($\text{AUC}_{BKT_C} = 0.5029$) and there is no significant improvement in CE.

Discussion. The results demonstrate that more complex DBN models outperform BKT in prediction accuracy. For hierarchical learning domains, CE can be reduced by 10%, while improvements of RMSE by 5% are feasible. The DBN models generally exhibit a significantly higher AUC than BKT, which indicates that they are better at discriminating failures from successes. As expected, adding skill topologies has a much smaller benefit for learning domains that are less hierarchical in nature (such as spelling learning). The results obtained on the physics and algebra data sets show that even simple hierarchical models improve prediction accuracy significantly. A domain expert employing a more detailed skill topology and more complex constraint sets could probably obtain an even higher accuracy on these data sets. The use of the same parameterization and constraint sets for all experiments demonstrates that basic assumptions about learning hold across different learning domains and thus the approach is easy to use.

4 Conclusion

In this work, we showed that prediction accuracy of a student model is increased by incorporating skill topologies. We evaluated the performance of our models on five data sets of different learning domains and demonstrated that the DBN models outperform the traditional BKT approach in prediction accuracy on unseen data. To conclude, our results show that modeling skill topologies is beneficial and easy to use, as even simple hierarchies and parameterizations lead to significant improvements in prediction accuracy. In the future, we would like to analyze the influence of the skill hierarchies and the different parameters in detail. We furthermore plan to apply the individualization techniques used in BKT [18, 24, 25] to DBNs. Moreover, we would like to explore further modelling techniques such as dynamic decision networks.

Acknowledgements. This work was funded by the CTI-grant 11006.1.

References

1. Baker, C., Tenenbaum, J.B., Saxe, R.: Bayesian models of human action understanding. In: Proc. NIPS (2005)
2. Baschera, G.M., Busetto, A.G., Klingler, S., Buhmann, J., Gross, M.: Modeling Engagement Dynamics in Spelling Learning. In: Proc. AIED (2011)

3. Conati, C., Gertner, A., VanLehn, K.: Using Bayesian Networks to Manage Uncertainty in Student Modeling. UMUAI (2002)
4. Corbett, A.T., Anderson, J.R.: Knowledge tracing: Modeling the acquisition of procedural knowledge. UMUAI (1994)
5. Frank, M.C., Tenenbaum, J.B.: Three ideal observer models for rule learning in simple languages. Cognition (2010)
6. Friedman, N., Murphy, K., Russell, S.: Learning the structure of dynamic probabilistic networks. In: Proc. UAI (1998)
7. Gross, M., Vögeli, C.: A Multimedia Framework for Effective Language Training. Computer & Graphics (2007)
8. Horvitz, E., Breese, J., Heckerman, D., Hovel, D., Rommelse, K.: The Lumière Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users. In: Proc. UAI (1998)
9. Käser, T., Baschera, G.M., Busetto, A.G., Klingler, S., Solenthaler, B., Buhmann, J.M., Gross, M.: Towards a Framework for Modelling Engagement Dynamics in Multiple Learning Domains. IJAIED (2012)
10. Käser, T., Baschera, G.M., Kohn, J., Kucian, K., Richtmann, V., Grond, U., Gross, M., von Aster, M.: Design and evaluation of the computer-based training program Calcularis for enhancing numerical cognition. Front. Psychol. (2013)
11. Käser, T., Busetto, A.G., Solenthaler, B., Baschera, G.M., Kohn, J., Kucian, K., von Aster, M., Gross, M.: Modelling and Optimizing Mathematics Learning in Children. IJAIED (2013)
12. Käser, T., Schwing, A.G., Hazan, T., Gross, M.: Computational Education using Latent Structured Prediction. To appear in Proc. AISTATS (2014)
13. Käser, T., Busetto, A.G., Baschera, G.M., Kohn, J., Kucian, K., von Aster, M., Gross, M.: Modelling and optimizing the process of learning mathematics. In: Proc. ITS (2012)
14. Koedinger, K.R., Anderson, J.R., Hadley, W.H., Mark, M.A.: Intelligent tutoring goes to school in the big city. IJAIED (1997)
15. Koedinger, K., Baker, R., Cunningham, K.and Skogsholm, A., Leber, B., Stamper, J.: A Data Repository for the EDM community: The PSLC DataShop. In: Handbook of Educational Data Mining. CRC Press, Boca Raton, FL (2010)
16. Lafferty, J., McCallum, A., Pereira, F.C.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proc. ICML (2001)
17. Mayo, M., Mitrovic, A.: Optimising its behaviour with bayesian networks and decision theory. IJAIED (2001)
18. Pardos, Z.A., Heffernan, N.T.: Modeling Individualization in a Bayesian Networks Implementation of Knowledge Tracing. In: Proc. UMAP (2010)
19. Pardos, Z.A., Trivedi, S., Heffernan, N.T., Sárközy, G.N.: Clustered knowledge tracing. In: Proc. ITS (2012)
20. Pavlik, P.I., Cen, H., Koedinger, K.R.: Performance Factors Analysis - A New Alternative to Knowledge Tracing. In: Proc. AIED (2009)
21. Reye, J.: Student Modelling Based on Belief Networks. IJAIED (2004)
22. Schwing, A.G., Hazan, T., Pollefeys, M., Urtasun, R.: Efficient Structured Prediction with Latent Variables for General Graphical Models. In: Proc. ICML (2012)
23. Wang, Y., Beck, J.: Class vs. Student in a Bayesian Network Student Model. In: Proc. AIED (2013)
24. Wang, Y., Heffernan, N.T.: The student skill model. In: Proc. ITS (2012)
25. Yudelson, M.V., Koedinger, K.R., Gordon, G.J.: Individualized Bayesian Knowledge Tracing Models. In: Proc. AIED (2013)