Choosing a Student Model for a Real World Application

Jiří Řihák and Radek Pelánek

Masaryk University Brno

Abstract. A student model is a key part of an adaptive educational system. Research literature offers wide choice of student modeling approaches, but little guidance on how to choose a suitable model for a real world application. Which aspects of student modeling are most important for a particular setting? How to decide whether a model is ready for application? We study these questions for a particular case study using generally applicable techniques. Our results illustrate the use of "wide and shallow" models to identify the most important aspects of student modeling. The case study (from the domain of mathematics education) specifically demonstrates that response times deserve more attention in student modeling.

1 Introduction

Intelligent tutoring systems aim at providing students with personalized, adaptively selected content. This behaviour is typically based on a student model, which estimates the current knowledge of students. The research literature provides many student modeling ideas, but little guidance on how to choose and tune a student model for particular application. For developers of real world applications it is thus difficult to choose a suitable model and this hinders spread of intelligent techniques in education.

There are many student modeling approaches and many features that we may include in our model. To illustrate the wide potential of student modeling, we mention several possible aspects of student modeling with examples of research. Basic student models focus on correctness of answers and modeling of learning in different ways (e.g., Bayesian knowledge tracing [2] or models based on logistic function [7]). Models need to know mapping between items and modeled skills and potentially relations between skills. This issue is called skill modeling [3] and can be addressed by wide variety of approaches (e.g., Q-matrix [1] or Bayesian methods). In addition to correctness of answers we can use other observational data, e.g., response times [8], common wrong answers [16], or hints usage [15]. Student population is typically not completely homogeneous and thus models may be improved by student clustering [6]. Research papers typically focus on detailed analysis of one of these aspect without any consideration of others.

From practitioner point of view there are several important questions which are not properly addressed in current literature: Which aspects of student modeling are most important for a particular setting? Which aspects are of key

importance and which provide only minor improvement not worth bothering with? Which techniques should be used to select an appropriate model? Is a student model ready for practical application? Are model parameters stable?

In this work we try to address some of these issues. As a case study we use a real adaptive educational system in its early stage of application, where the choice of student modeling approach is a real, pressing development issue. We explore a range of modeling approaches, discuss their relations and comparison, and study parameter stability. The result illustrate the use of "wide shallow" models to identify the most important aspects of student modeling for a particular application. Specifically, the results show usefulness of paying attention to response times and stability of model parameters, which are both neglected issues in current literature on student modeling.

2 Setting

To explore the issue of model selection we utilize data from a realistic case study and compare a sample of potential student models.

2.1 The MatMat System

The MatMat system (`matmat.cz`) is an adaptive practice application for children, which covers the area of basic arithmetic (counting, addition, multiplication, etc.), its functionality is similar to Math Garden [8]. The system is available freely online (registration is possible, but not required). The application is adaptive, its behaviour (used item selection algorithm) and default student model are described in [14].

The studied application is in many aspects typical online educational system. One specific aspect of the domain of basic arithmetic is that response times seem particularly important. For example multiplication of small numbers starts as procedural knowledge (a child knows that $3 \cdot 5$ is $5 + 5 + 5$ and is able to do the calculation using fingers) but ends as declarative knowledge (a child knows $3 \cdot 5 = 15$ without further thoughts). In both cases a child can give a correct response with high probability, to distinguish between these levels of knowledge it seems useful to utilize response times.

The currently available data from the system comprise 150,000 answers, the distribution of answers between users is highly skewed (many users answer only few questions). The system contains examples divided into 5 high level concepts (counting, addition, subtraction, multiplication, division), each of these concepts contains around 50-700 items, over 2,000 items in total. Although the data set is relatively small (compared to many data sets currently used in research), it constitutes a realistic case study for the studied questions – developers of new systems need to make decisions on the use of student models with relatively small data.

2.2 Student Models

Our purpose in this work is not to find some "optimal" model, but rather to provide insight into importance and relation of different aspects of student models. Thus the following list of models is certainly not an exhaustive enumeration of options feasible for the particular application, but rather a selection of reasonable approaches.

Basic Modeling Approach As a baseline for comparisons we use a simple *item average* model, which predicts performance based on the percentage of correct answers for the particular item (ignoring student characteristics).

As the basic modeling approach, which is further studied and extended, we use student modeling based on logistic function. Note that Bayesian knowledge tracing [2], the currently dominant student modeling approach, is not suitable for the studied application – the binary skill assumption is unrealistic and incorporation of difficulty of items and timing information is possible, but complicated. The modeling approach based on logistic function naturally allows incremental increase in skill (which is expected in the domain of basic arithmetic) and can quite easily incorporate different additional aspects.

Specifically, we use the Elo rating system [4, 11], which can be seen in its basic form as a heuristic for parameter estimation of the Rasch model (it gives nearly the same parameter estimates [11]). We use the Elo rating system due to its many practical advantages (a key factor for real world application) – parameter fitting is done naturally online, it is easy to implement and extend.

The model has a student skill parameter θ and an item difficulty parameter d. The probability that a student answers correctly is estimated using a logistic function of a difference between the skill and the difficulty: $P(correct|\theta, d) = 1/(1 + e^{-(\theta-d)})$. After observing the student's response the parameters are updated as follows: $\theta := \theta + U \cdot (correct - P(correct|\theta, d))$, $d := d + U \cdot (P(correct|\theta, d) - correct)$, where $correct \in \{0, 1\}$ denotes whether the question was answered correctly and U specifies sensitivity of parameter estimates to the last attempt. Based on previous research [9–11] we use an uncertainty function $U(n) = \alpha/(1 + \beta n)$, where n is the number of previous updates to the estimated parameters and α, β are meta-parameters (fitted using a grid search: $\alpha = 1.0, \beta = 0.1$).

Domain Modeling An important modeling aspect is domain modeling, particularly choosing the granularity of the model and the item-skill mapping. For the studied system, the items that are presented to students are for example: "5×3", "$2 + 3$" (with additional visualization), or "choose a number 6 on the number line". There are many ways how to model this domain; for the evaluation we have chosen three representatives. All of them have difficulty parameter for each item, they differ in the way they model student skills:

- *Basic model* – for each student there is a single skill.
- *Concept model* – for each student there are skill parameters for each of the 5 main concepts, these skills are assumed to be independent.

– *Hierarchical model* – skill of each student is described in tree-like structure (see Fig. 1) with 3 levels of concepts and sub-concepts (described in [14]). Due to this structure skills for different concepts are interconnected.

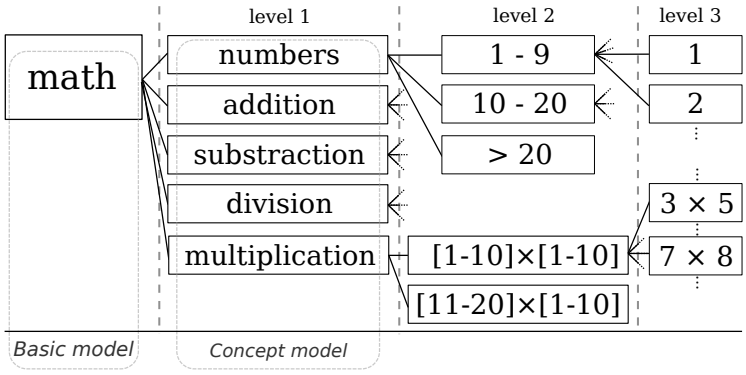

Fig. 1. The structure of the hierarchical model. The basic model and the concept model can be viewed as selected independent concepts without dependencies.

Response Times Most student modeling approaches consider only correctness of answers. Response times, however, can provide useful information about student knowledge, particularly for the studied domain. In this work we use the following approach to incorporating response times to student models. We combine correctness and response time into single performance measure r. For wrong answers we ignore response times (i.e., r is constantly equal to 0). For correct answers we transform the value 1 into an interval $[0, 1]$ by one of the following functions (see Fig. 2):

– *noTime* – no use of time, $r = 1$.
– *thresholdTime* – response is classified as fast or slow (based on the threshold 7 seconds, the median response time in system); $r = 1$ for fast responses, $r = 0.5$ for slow responses.
– *expTime* – similar to the previous one; for fast responses $r = 1$, for slow responses r decreases exponentially (see [14] for more details).
– *linearTime* – response is linearly decreasing with increasing time (until it reaches 0): $r = max(0, 1 - t/14)$.

There are, of course, other possible approaches to modeling time. For example a similar approach based on the Elo rating system uses a "high speed high stakes" rule [8], which takes time into account even for wrong answers. A conceptually different approach is based on separate modeling of correctness and speed [13].

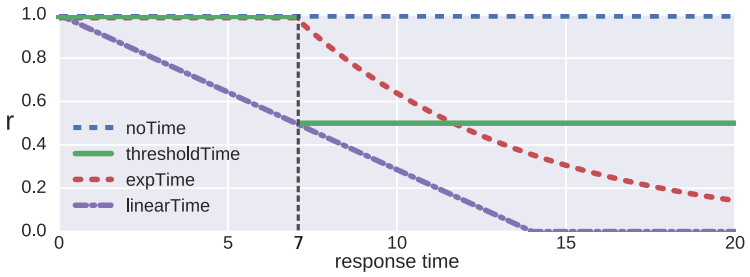

Fig. 2. Variants of response time based transformations of performance measure r for correct answer.

Wrong Answers Another potentially useful source of information about student knowledge is the specific answer, i.e., in the case of incorrect answer we can look at the specific mistake that was made. Previous research for example studied differences between common wrong answers and other mistakes [16]. Analysis of answers from the studied system shows that there is large proportion of missing answers. Moreover, these answers often have very short response times and are often present in sequences (i.e., users are sometimes skipping sequences of items).

We incorporate this insight into student models in the following simple way. We compute probability of missing answer based on the number of immediately preceding missing answers of a student. Overall prediction is the product of the original model prediction and the probability of not missing an answer. This new prediction is also used in the update of model parameters.

3 Evaluation

We analyze the described models on the data from the MatMat system – we start with comparison of prediction accuracy of models and continue with more detailed analysis of parameter values.

3.1 Prediction Accuracy

For comparison of predictive accuracy of models we use rather standard evaluation setting: repeated random cross-validation (20 runs) with student stratified train/test set division (70%/30%). As the basic performance metric for comparing model we use Root Mean Square Error (RMSE), which is a standard choice in the case of predicting correctness of answers [12].

Fig. 3 gives comparison of different domain modeling approaches and of the impact of explicit treatment of missing answers. With respect to domain modeling, we see that more complex models are able to improve predictions, although increasing complexity of models brings only diminishing improvements. The figure also shows that explicit treatment of missing answers can significantly improve accuracy of some models. This very straightforward and simple

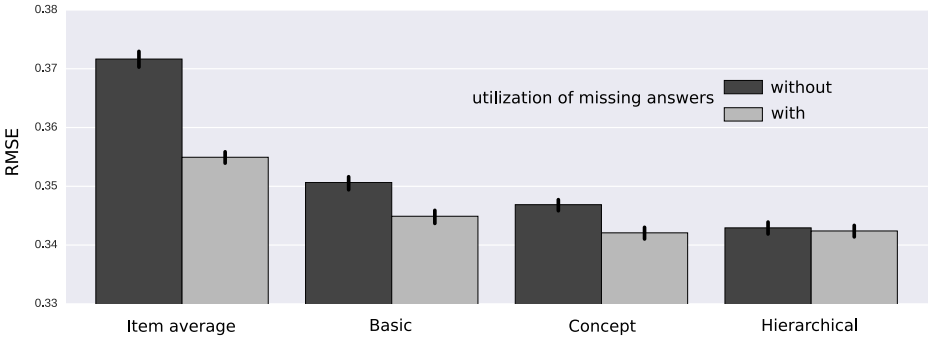

Fig. 3. Comparison of domain modeling approaches and of the impact of utilization of missing answers. Error bars show 95% confidence intervals.

technique brings large improvement for baseline model, nontrivial impact on the basic model and the concept model, but only minimal impact on the hierarchical model, which is more flexible and can quickly adapt to the students who are just skipping items. This shows that different aspects of student modeling are partly compensatory, i.e., novel modeling ideas should not be judged (only) by their ability to improve simple baselines.

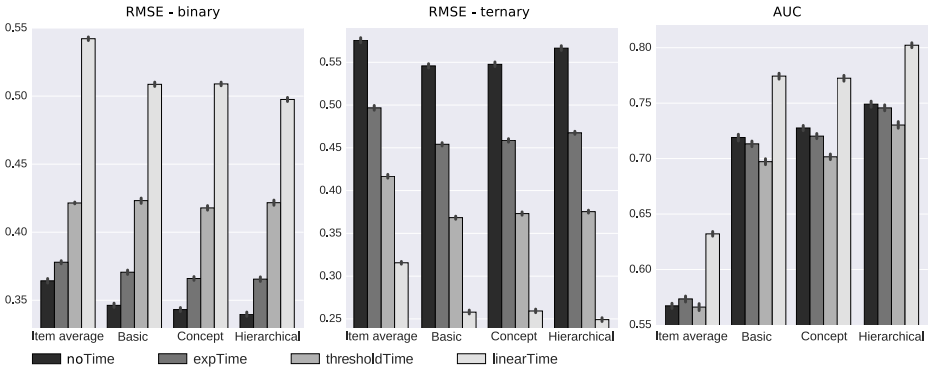

Fig. 4. Model comparison according to RMSE and AUC with and without time information. Error bars show 95% confidence intervals. Note that for the RMSE metric lower values mean better performance, whereas for the AUC metric higher values are better. Also note that the RMSE values of different time uses are not straightforwardly comparable because models are trained to predict different absolute values.

The evaluation of models which consider timing information is more difficult. Standard performance metrics [12] consider only binary information about correctness, but the point of models with timing information is to also distinguish between slow and fast responses. The full discussion of this issue is beyond the

scope of this paper, here we report just three selected metrics to provide basic insight into model behaviour. At first, we use the standard RMSE (i.e., the performance metric ignores the timing information). At second, we use "ternary" version of RMSE where observations are labeled as 0 (wrong), 0.5 (correct, slow), and 1 (correct, fast). At third, we use the Area Under the ROC Curve (AUC) metric [12], which ignores the timing information, but considers the prediction only in relative manner.

The comparison of models according to these metrics is shown in Fig. 4. With respect to domain modeling the results are similar to results presented in Fig. 3. Also note that the modeling of timing information is rather orthogonal to domain modeling. With respect to different variants of modeling response times, we see large differences between models, but these differences are not easy to interpret due to the impact of used metric. The results for the two variants of RMSE are not very surprising, since RMSE takes into account absolute values of predictions and different models are trained to predict different values, e.g., models with *linearTime* have much higher RMSE since they are trained to give much smaller predictions (with different meaning). The best models with respect to reported RMSE metrics are those that match the performance metric used for evaluation. The interesting part of results is the comparison with respect to AUC. Although the evaluation metric does not take response time into account, the best results are achieved using *linearTime* model variants. This modeling approach improves the relative order of predictions (slower students are relatively more likely to make mistake than faster students with the same overall correctness).

3.2 Impact on Estimated Parameters

The summary metrics for predictive accuracy are notoriously hard to interpret [12]. How important are small differences in RMSE values? To get insight into differences between models we analyze correlations between parameter values, particularly the item difficulty parameters, which have clear interpretation and direct impact on the adaptive system behaviour (e.g., adaptive selection of items or provided feedback for students).

Fig. 5 shows correlations between item difficulties for different models. Darker color means higher correlation, i.e., more similarity in difficulty estimates (less important difference between models). Unsurprisingly, there is a large gap between the baseline model and other more sophisticated models. The impact of domain modeling is nontrivial, but not pronounced. Different utilization of time, however, brings considerably different parameters. The degree of change is proportional to the intensity of time utilization, *linearTime* extension is the most different. Also note that the figure contains repetitive 4×4 pattern corresponding to different time usage. This means that domain modeling and time modeling are almost independent modeling aspects and provide change (and possible improvement) in different directions.

To provide better intuition beyond the summary evaluation metrics and correlations, we provide a specific illustration of the model impact on parameters of simple addition and subtraction items. Fig. 6 presents comparison of estimated

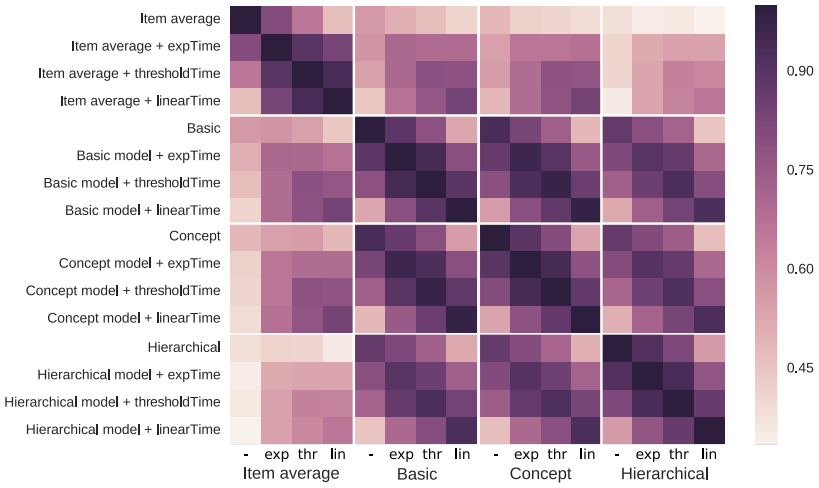

Fig. 5. Spearman correlation coefficient for estimated item difficulty parameters of different models.

parameters for a model with and without timing information. The estimated parameters are only weakly correlated and there are significant differences (e.g., "1-1" and "8-5" have the same difficulty according to model without response times, but quite different difficulty according to the model with time). Particularly, note the highlighted subtraction examples of the type "X-X", which the model with timing information systematically rates as easier than the model without time.

3.3 Parameter Stability

Before we use a model in an adaptive educational system we want to be sure that its parameters are reasonably stable. For example in Fig. 6 we can see that the estimated parameters are probably not yet completely stable (one would intuitively expect for example that the items "X-X" would either have very similar difficulty or be ordered). How to objectively judge parameter stability? How quickly do parameter values stabilize? How much do different models different in their speed of convergence? Such questions do not get much attention in student modeling. A recent exception is the proposal for multifaceted evaluation of student models [5], where authors include parameter stability as a criterion for model evaluation, but they discuss only a specific model (BKT) and do not focus on dynamics of parameter stability.

To evaluate these questions we performed the following experiment. We took two data samples D_1, D_2 of size K (student stratified, without intersection). We consider a particular model type M and train an instance M_1 using the data set D_1 and an instance M_2 using the data set D_2. Then we evaluated the correlation

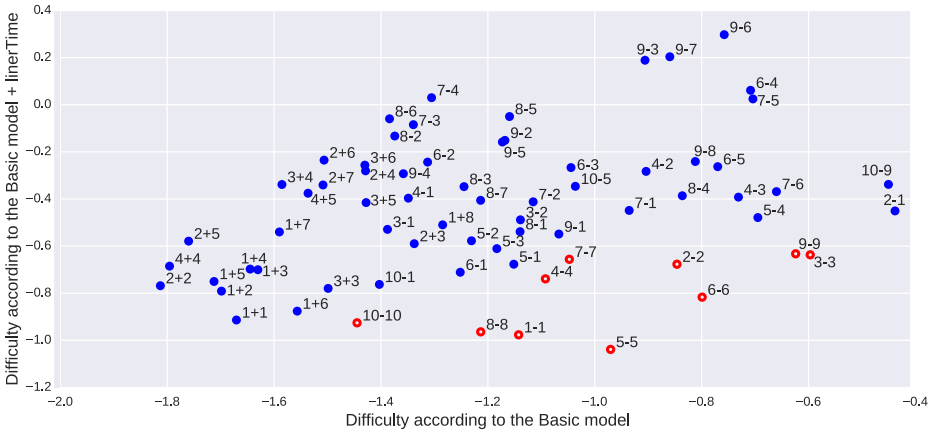

Fig. 6. Comparison of estimated difficulties of selected items by the basic model with and without time usage. Note that absolute values of difficulties are not entirely comparable because models are trained to predict different values.

of parameters of M_1 and M_2 (we use only item difficulty parameters since the data sets are student stratified).

Fig. 7 shows increase in parameter stability with the number of answers used for training the model. The figure compares different domain modeling approaches and different time uses. The left part of the figure shows that the hierarchical model is slightly more stable than simpler models as it can better carry information across items. The right part of the figure shows high increase in stability of models which utilize response times. Fig. 6 provides a specific illustration, note that the group of similar items of the form "X-X" have very similar difficulty according to the model utilizing response time, but widely different difficulties for model without response times. This increase in stability (resp. faster convergence) is probably mainly due to the use of more "bits of information" per each answer. Increase in stability is also proportional to intensity of time utilization – *linearTime* extension makes the most significant use of response times.

4 Discussion

For the studied case study, the main conclusion is that differences in modeling of response times have larger impact than differences in domain modeling. The results give actionable insights for practical application and directions of further analysis (e.g., separate treatment of correctness and speed may be useful).

From the wider perspective, the case study highlights issues with practical applications of research results. The focus of current research is mostly on details of one particular aspect of student modeling (e.g., domain modeling or different ways of modeling learning). For applications of student models in real

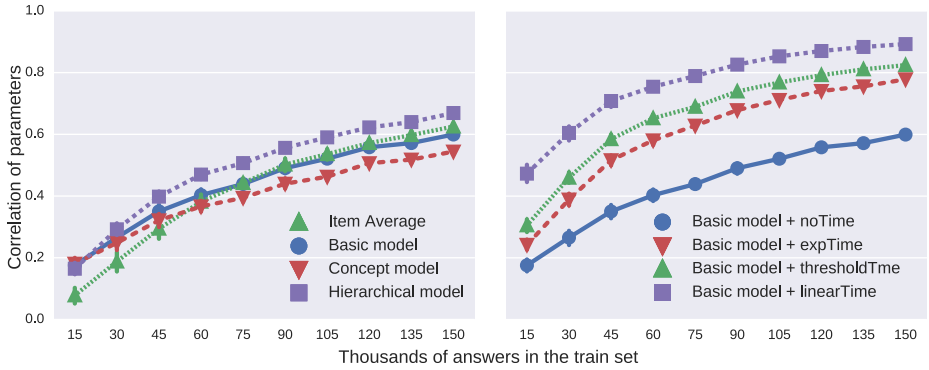

Fig. 7. Parameter stability: correlation between item parameter values for models fitted on two samples of a given size. Left: Comparison of different domain models. Right: Comparison of different response time models. Confidence intervals are mostly too small to be visible.

world contexts, we need more focus on relations of different aspects of student modeling and on their combinations. Our results show that incorporation of different aspects of student modeling (even in simple way) may be more important than detailed modeling of one particular aspect. Different modeling aspects are not, however, always orthogonal. An approach that brings large improvement over baseline (when studied in isolation) may have negligible impact when incorporated into a more complex model – in our experiments this is the case of treatment of skipped answers. From application point of view, more attention also needs to focus on "production readiness" of models.

These concerns are not just pragmatic, development issues, they have important consequences for research, e.g., comparison with baseline not sufficient to show that a modeling technique will improve also more complex models. They can also inspire interesting scientific questions. Can we devise compositional modeling approaches allowing simple combination (and evaluation) of different aspects of student modeling? To what degree is the importance of individual aspects domain specific? Can we formulate any generally applicable guidelines? What is the best way to study stability of model parameters and how can it help in model comparison?

References

1. Tiffany Barnes. The q-matrix method: Mining student response data for knowledge. In *American Association for Artificial Intelligence 2005 Educational Data Mining Workshop*, 2005.
2. Albert T Corbett and John R Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278, 1994.

3. Michel C Desmarais and Ryan SJ d Baker. A review of recent advances in learner and skill modeling in intelligent learning environments. *User Modeling and User-Adapted Interaction*, 22(1-2):9–38, 2012.
4. Arpad E Elo. *The rating of chessplayers, past and present*, volume 3. Batsford London, 1978.
5. Yun Huang, José P González-Brenes, Rohit Kumar, and Peter Brusilovsky. A framework for multifaceted evaluation of student models. In *Educational Data Mining*, 2015.
6. Tanja Käser, Alberto Giovanni Busetto, Barbara Solenthaler, Juliane Kohn, Michael von Aster, and Markus Gross. Cluster-based prediction of mathematical learning patterns. In *Artificial Intelligence in Education*, pages 389–399. Springer, 2013.
7. Tanja Käser, Kenneth R Koedinger, and Markus Gross. Different parameters - same prediction: An analysis of learning curves. In *Educational Data Mining*, pages 52–59, 2014.
8. S Klinkenberg, M Straatemeier, and HLJ Van der Maas. Computer adaptive practice of maths ability using a new item response model for on the fly ability and difficulty estimation. *Computers & Education*, 57(2):1813–1824, 2011.
9. Juraj Nižnan, Radek Pelánek, and Jiří Řihák. Student models for prior knowledge estimation. In *Proc. of Educational Data Mining*, pages 109–116, 2015.
10. Jan Papoušek, Radek Pelánek, and Vít Stanislav. Adaptive practice of facts in domains with varied prior knowledge. In *Educational Data Mining (EDM)*, pages 6–13, 2014.
11. Radek Pelánek. Application of time decay functions and Elo system in student modeling. In *Educational Data Mining (EDM)*, pages 21–27, 2014.
12. Radek Pelánek. Metrics for evaluation of student models. *Journal of Educational Data Mining*, 7(2), 2015.
13. W.J. Van Der Linden. Conceptual issues in response-time modeling. *Journal of Educational Measurement*, 46(3):247–272, 2009.
14. Jiří Řihák. Use of time information in models behind adaptive system for building fluency in mathematics. In *Educational Data Mining, Doctoral Consortium*, 2015.
15. Yutao Wang and Neil Heffernan. Extending knowledge tracing to allow partial credit: Using continuous versus binary nodes. In *Artificial Intelligence in Education*, pages 181–188. Springer, 2013.
16. Yutao Wang, Neil T Heffernan, and Cristina Heffernan. Towards better affect detectors: effect of missing skills, class features and common wrong answers. In *Learning Analytics And Knowledge*, pages 31–35. ACM, 2015.