Application of Time Decay Functions and the Elo System in Student Modeling

Radek Pelánek
Masaryk University Brno
pelanek@fi.muni.cz

ABSTRACT

One of the key aspects of educational data mining is estimation of student skills. This estimation is complicated by the fact that students skills change during the use of an educational system. In this work we study two flexible approaches to skill estimation: time decay functions and the Elo rating system. Results of experiments in several different settings show that these simple approaches provide good and consistent performance. We argue that since these approaches have several pragmatical advantages (flexibility, speed, ease of application) they should be considered in educational data mining at least as a baseline approach.

1. INTRODUCTION

One of the goals of educational data mining is to estimate skill (knowledge) of students. The problem of skill estimation is the following: we have sequential data about student performance (e.g., answers to exercises, timing) and we want to estimate a latent student skill. The quality of the skill estimate can be evaluated by its ability to predict future performance. Once we have a reliable skill estimate, it can be used in many ways: for guiding adaptive behaviour in intelligent tutoring systems, for computerized adaptive practice, or for providing feedback to students (e.g., in skillometers, open learner models).

In skill estimation, there are two main approaches to dealing with the sequentiality of the data. One approach is simply to ignore the ordering of the data, i.e., to make a simplifying assumption that students do not learn and the skill is a constant. This approach is usually used with "coarse-grained" skills (like "fractions" or even "arithmetic"), where the rate of skill change is slow and thus the assumption of constancy is reasonable. A typical example of this approach is item response theory [3], which is used mainly for adaptive testing. In this case the assumption is justified since we do not expect students to learn during test. But even some models used in adaptive learning systems do not consider the order of data and treat all data points in same way (e.g., performance factor analysis [19] or a model of problem solving times [10]).

The second main approach is to make a fixed assumption about learning and "hard code" it into the model. A typical example of this approach is Bayesian knowledge tracing (BKT) [2, 22], which models knowledge as a binary variable (known/unknown) with a given probability of switching from unknown to known. Another approach of this type are models based on learning curves [16], which typically assume a logarithmic increase in skill with respect to number of attempts (e.g., a model of problem solving times with learning [9]). These types of models are used mainly with "fine-grained" skills (e.g., a specific operations with fractions). For their application it is necessary that the skills are correctly identified, so that the model assumptions hold [2].

In this work we study educational application of two interrelated techniques – time decay functions and the Elo system. These techniques are between the two above described approaches. They do take the sequentiality of the data into account, but do not make fixed assumptions about learning. Both techniques are rather flexible and thus are applicable to wide range of skill granularity.

The first technique is based on time decay functions. Since students skills and knowledge changes over time, the older data are less relevant for the estimation than the recent data. Thus it makes sense to use some kind of data discounting – in analysis of sequential data this can be done using weighting by a time decay function [12, 6]. Only little research in student modeling has so far studied data discounting or some similar temporal dynamics, e.g., using less data in BKT [17], data aging [29], or effect of real time (not just ordering) in BKT [20].

The second technique is the Elo system [4], which was originally devised for chess rating (estimating players skills based on results of matches), but has recently been used also for student modeling [13, 27]. In context of skill estimation we interpret an attempt of a student to answer an item as a "match" between the student and the item. This approach updates a skill estimate based on the result of a last match in such a way that implicitly leads to a discounting of past attempts.

The goal of this work is to explore applicability of time decay functions and Elo system in educational data mining.

More specifically to study the following questions: What is a good time decay function in the context of educational data mining? How sensitive are results with respect to parameters of time decay function and Elo rating? How do these approaches compare to other student modeling techniques? To answer these questions we apply the techniques different contexts and we use for evaluation several different datasets. The obtained results are quite stable and favourable for these approaches, and thus we also discuss their possible application in intelligent tutoring systems.

2. MODELS FOR SKILL ESTIMATION

We study the skill estimation in two context: modeling of correctness of student answers (the only measure of performance is correctness of the answer, possibly also the number of hints used) and modeling of problem solving times (the only measure of performance is a time to solve a problem).

2.1 Overview of Relevant Models

In item response theory the main model is the 3 parameter logistic model, which assumes a constant student skill θ and three item parameters: b is the basic difficulty of the item, a is the discrimination factor, and c is the pseudo-guessing parameter. The model assumes that the probability of a correct answer is given by a (scaled) logistic function:

$$P_{a,b,c,\theta} = c + (1-c)\frac{e^{a(\theta-b)}}{1+e^{a(\theta-b)}}$$

A specific case of this model is a 1 parameter model, which is obtained by setting $c = 0, a = 1$; this model is also called the Rasch model.

A model of problem solving times [10] uses parameters with analogous meaning and assumes a log-normal distribution of problem solving times:

$$f_{a,b,c,\theta}(\ln t) = \mathcal{N}(a\theta + b, c)(\ln t) = \frac{1}{\sqrt{2\pi}c}e^{-\frac{(\ln t-(a\theta+b))^2}{2c^2}}$$

Bayesian knowledge tracing [2, 22] models a changing skill. It is a hidden markov model where skill is the binary latent variable (either learned or unlearned). The model has 4 parameters[1]: probability that the skill is initially learned, probability of learning a skill in one step, probability of incorrect answer when the skill is learned (slip), and probability of correct answer when the skill is unlearned (guess). The skill estimated is updated using a Bayes rule based on the observed answers.

2.2 Time Decay Functions

Time decay function are used in the study of concept drift [6, 12, 21]. Concept drift is relevant example for modeling the change of user preferences in recommender systems, where the inclusion of temporal dynamics into models can improve their performance [15]. A different area that uses temporal discounting is economics and study of decision making [5], where temporal discounting and time decay functions are studied mainly with respect to decisions about future. All these areas can provide useful inspiration for student modeling (e.g., the choice of the time decay function), but are not directly applicable.

[1]BKT can also include forgetting. The described version corresponds to the variant of BKT that is most often used in research papers.

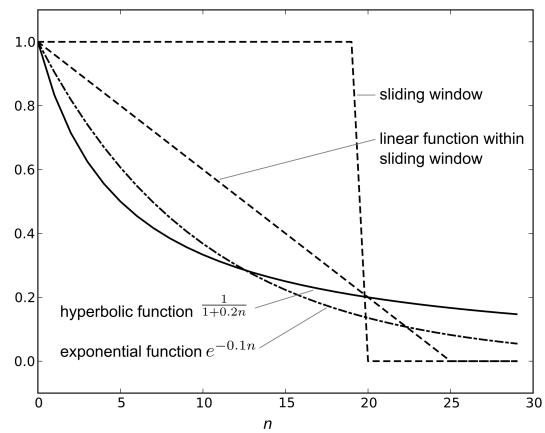

Figure 1: Examples of time decay functions.

A time decay function assigns a weight to a data point (student performance) that happened in the past. As a measure of "time" we use a number of attempts (denoted n). Other possibilities are to use a "real time" (seconds from the attempt) or "semi-real time", which counts the number of attempts but takes into account big pauses (e.g., larger step for a day switch). Figure 1 shows several natural candidates for time decay functions, which we have evaluated in our experiments.

Let us apply time decay functions to student modeling. In the case of modeling correctness of answers, we have data of the following type: student s gave to an item i an answer with correctness c_{si} (usually a binary variable, in case of a "partial credit model" [25] it can also have a continuous value between 0 and 1). The skill of a student s is estimated as a weighted average of c_{sk} with weights given by the time decay function, i.e., $\theta_s = \sum f(k)c_{si_k} / \sum f(k)$, where i_k is the item solved by the student k steps into the past. This skill estimate is in the range $[0, 1]$ and can be directly used to predict future performance.

We also study modeling of problem solving times. In accordance with previous research [10, 23], we work with the logarithm of time, since raw times are usually log-normally distributed. Now we assume data of the type: student s solves a problem p in a logarithm of time t_{sp}. We denote θ_{sp} a "local skill estimate" on a particular problem: $\theta_{sp} = m_p - t_{sp}$, where m_p is a mean time to solve the problem p. A current skill of a student s is estimated as a weighted average of these local estimates with weights given by the time decay function: $\theta_s = \sum f(k)\theta_{sp_k} / \sum f(k)$, where p_k is the problem solved in k steps into the past. The skill estimate can be used to predict performance on an unsolved problem p as follows: $\hat{t_{sp}} = m_p - \theta_s$. Note that with a constant weight function this approach is equivalent to the baseline personalized predictor used in [9, 10].

Compared to more complex students models (BKT, model of problem solving times) the outlined approaches to estimating student skill are quite simple. The advantage of this simplicity (apart of simplicity of implementation and application) is that they make minimal assumptions about the behaviour of students, e.g., this approach can naturally accommodate forgetting (as opposed to BKT, where the inclusion of forgetting means an additional parameter) and also such effects as a change of working environment (e.g., switching from a computer with mouse to notebook with touchpad can increase problem solving times for interactive problems).

2.3 The Elo System

The basic principle of the Elo system is the following. For each player i we have an estimate θ_i of his skill, based on the result R (0 = loss, 1 = win) of a match with another player j the skill estimate is update as follows:

$$\theta_i := \theta_i + K(R - P(R = 1))$$

where $P(R = 1)$ is the expected probability of winning given by the logistic function with respect to the difference in estimated skills, i.e., $P(R = 1) = 1/(1 + e^{-(\theta_i - \theta_j)})$, and K is a constant specifying sensitivity of the estimate to the last attempt.

There exists several extension to the Elo system, the most well-known are Glicko [7], which explicitly models uncertainty in skill estimates, and Trueskill [8], which can be used also for team competitions. The Elo system has also been used previously in modeling of correctness of student answers by interpreting student solution attempt as a match between a student and an item [13, 27].

In the case of problem solving times we can apply the method as follows: for each student we have an skill estimate θ_s, for each problem we have a difficulty estimate d_p. When the student s solves the problem p in the logarithm of time t_{sp} we update these estimates as follows:

$$\theta_s := \theta_s + K(E(t|s,p) - t_{sp})$$

$$d_p := \theta_p + K(t_{sp} - E(t|s,p))$$

where $E(t|s,p)$ is an expected solving time for a student s and problem p, which is given as $E(t|s,p) = d_p - \theta_s$.

The value of the constant K determines the behaviour of the system – if K is small, the estimation converges too slowly, if K is large, the estimation is unstable (it gives too large weight to last few attempts). An intuitive improvement, which is used in most Elo extensions, is to use an "uncertainty function" instead of a constant K. Previous work on using the Elo system for student modeling [13, 27] used ad hoc uncertainty functions selected for particular application.

An important difference of application of the Elo systems in its typical domains (chess and other competitions) and in student modeling, is the asymmetry in student modeling between students and problems. Particularly we typically have much more students than problems and consequently more data about particular problems than students. Thus it makes sense to use different uncertainty functions for students and problems.

2.4 Relation between Time Decay and the Elo System

Both described approaches are closely related – they can both capture changing skill and do not make any specific assumptions about the nature of the change, they just give more weight to recent attempts. The close relation between these two approaches is apparent particularly in modeling of problem solving times. Using the previously described notation of a local performance $\theta_{sp} = d_p - t_{sp}$, the update rule of the Elo system can be rewritten as follows:

$$\begin{aligned}\theta_s := \quad & \theta_s + K(E(t|s,p) - t_{sp}) = \theta_s + K(d_p - \theta_s - t_{sp}) = \\ & \theta_s + K(\theta_{sp} - \theta_s) = (1 - K)\theta_s + K\theta_{sp}\end{aligned}$$

Now if we consider a sequence of n solved problems and assume an initial skill estimate 0, the final skill estimate is given by:

$$\theta_s = K \sum_{i=1}^{n} (1 - K)^{n-i} \theta_{sp_i}$$

The resulting expression is very similar to the estimation with exponential decay function, the main difference is the use of m_p (mean problem solving time) versus d_p (difficulty parameter estimated by the Elo system), but since problems are usually solved by large numbers of students and difficulty parameter is quite easy to estimate [9], this difference is not practically important.

The relation with time decay function is not so straightforward for applications of the Elo system to correctness data, which uses the logistic function, and for extension of the Elo system with uncertainty function. Nevertheless, some sort of temporal dynamics is inherently included in all variants of the Elo system. Extension usually correspond to the use of a steep decay function during first few student attempts and flatter decay function later, when the skill estimate is more stable.

3. EVALUATION

We present evaluation of time decay functions and the Elo system on several different datasets. Rather than performing one exhaustive experiment with one dataset, we performed several basic experiments in different settings (different types of datasets, simulated data).

3.1 Simulated Data

Using simulated data we explore how well can the Elo system and estimation using time decay functions approximate previously studied models (mentioned in Section 2.1). We use the following type of experiment: we generate data using one of the standard models and then try to fit the data using one of the studied approaches (the Elo system, time decay functions).

The first experiment concerns comparison of the Rasch model (one parameter logistic model) and the Elo system. These two approaches are very similar, since both assume one student parameter (skill), one item parameter (difficulty), and the same functional form of the probability of correct answer (logistic function with respect to the difference between skill and difficulty). The differences between these approaches are in the assumption about constancy of parameters and in parameter estimation methods. The Rasch model assumes

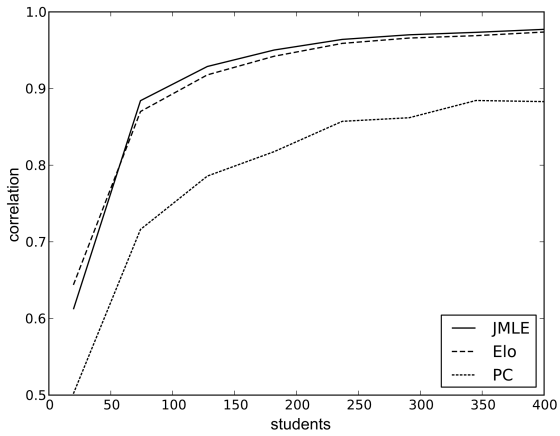

Figure 2: Correlation between generated and estimated difficulty parameters for different number of students. JMLE = Joint maximum likelihood estimation, Elo = Elo system, PC = proportion correct.

that the parameters are constant, specifically that the skill is constant (i.e., no learning). The standard method for estimating parameters of the Rasch model is the iterative procedure joint maximum likelihood estimation (JMLE) [3]. The Elo system does not make any specific assumptions about the constancy or change of the skill or difficulty and tracks these parameter in more heuristic fashion.

We performed the following experiment. The simulated data are generated using the Rasch model with skills and difficulties generated from standard normal distribution. The data are then fitted using JMLE and the Elo system and we compare the fitted values of parameters with the generated values. As a metric of fit we use the correlation coefficient. The Elo system with constant K leads to significantly worse results than JMLE, but if we use a suitable uncertainty function, the two estimation procedures give very similar results (correlation mostly above 0.99). A suitable uncertainty function is for example the hyperbolic function $\frac{a}{1+bx}$, with parameters $a = 4, b = 0.5$. Suitable parameters can be easily found by grid search, the performance of the system is quite stable and the precise choice of parameter values is not fundamental to the presented results.

In the case that we have complete data about answers or data are missing at random, even a simple "proportion correct" statistics gives good prediction of item difficulty. However, in real systems data are not missing at random, particularly in adaptive systems more difficult items are solved only by students with above average skill.

Figure 2 shows the results for such scenario. Data are generated using the Rasch models, portion of the data is missing, the availability of answers is correlated with student skill and item difficulty. The data are generated for 100 items and different number of students, the results are averaged over 50 runs. In this scenario, results for "proportion correct" are significantly worse than for the other two methods, detailed analysis shows that the estimates are wrong particularly in the middle of the difficulty range. Results for JMLE and Elo are nearly identical, the graph demonstrates that the difference in the amount of available data is more important than the difference between the estimation procedure used. If we have enough data, the JMLE is slightly better than Elo, but for small amount of data Elo is even better than JMLE. Note that this scenario is optimistic for the JMLE, since the simulated data adhere to the constancy of skill assumption, whereas any real data will contain at least some variability. We have performed similar experiments in the case of problem solving times. The results are similar, again we get similar performance and a suitable uncertainty function is the hyperbolic function.

Another experiment concerns comparison of the Bayesian knowledge tracing and skill estimation using time decay functions. Similarly to the previous experiment, we simulated data from a BKT model with fixed parameters. Then we use the BKT model and the time decay approach to make predictions and compare them using the AUC metric (results for RMSE metric are similar). For the predictions we use the BKT model with the optimal parameters, i.e., those used to generate the data. This is again overly optimistic case for BKT, as we assume that the data fully correspond to the assumptions of the model and that we know the correct parameter values. To make the comparison fairer, the estimation using time decay has at least the information about the initial probability of learned skill. Even in this setting, time decay approach gets close to BKT. For BKT parameters 0.5, 0.14, 0.09, 0.14 (taken from [20] as average BKT parameter values from the ASSISTments system), the AUC values are 0.822, 0.815. The time decay function used is the exponential function $e^{-0.3n}$; similarly to the previous experiment the choice of optimal value of the parameter can be done easily using an exhaustive search.

3.2 Real Data

At first we describe experiments with models of problem solving times. For this evaluation we use data from the Problem Solving Tutor [11], which is an open web portal with logic puzzles and problems from mathematics and computer science. For comparing different models we use root mean square error (RMSE) metric.

The results show that time decay functions can bring improvement of predictions. Figure 3 shows results for the exponential decay function. As the graph shows, the optimal parameter k for the exponential function e^{-kn} is around 0.1. Hyperbolic function $1/(1 + kn)$ achieves similar results as the exponential function, with optimal values of the parameter k in the interval 0.2 to 1.2. The sliding window and linear function within sliding window achieve significantly worse results.

Different problem types behave similarly with respect to which time decay functions and which parameter values bring the best improvement. They, however, differ in the amount of improvement. For some problems the improvement is only minor – these are for example Tilt maze and Region division, which are rather simple puzzles where we do not expect significant learning or other temporal effects affecting performance. Hence it is not very useful to discount data about past attempts. On the other end are problems like

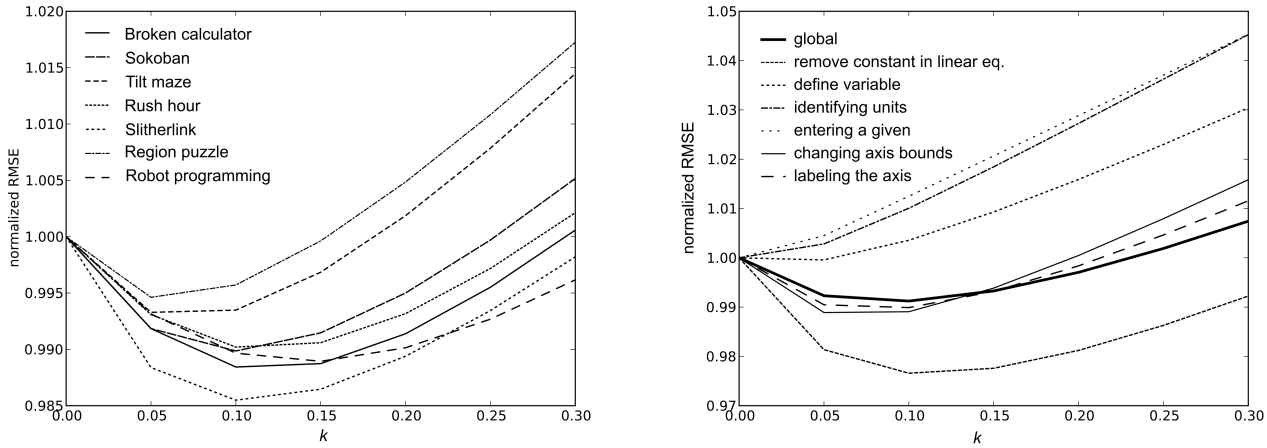

Figure 3: Results for exponential time decay function e^{-kn} for varying k; the graph shows normalized RMSE (with respect to constant time decay function). Left: Data from Problem Solving Tutor (problem solving times), Right: Algebra data set (correctness data).

Slitherlink (more advanced logic puzzle) or Broken Calculator (practice of calculations), where the improvement is larger and the best results are obtained by steeper decay functions. For these problems learning is more significant and thus it is sensible to take into account particularly last few attempts.

In previous work [9] we have proposed a model of problem solving times that makes a fixed assumption about learning, particularly the assumption of logarithmic improvement with respect to the number of attempts (in agreement with the research on learning curves). For the used dataset, this model does not bring any systematic improvement in predictions, whereas time decay functions do improve predictions (see [14] for more detailed analysis of this comparison). Thus it seems that for the used dataset there are temporal effects in the performance of students that do not easily conform to the assumptions of learning curves – the dataset contains nonstandard educational problems and logic puzzles and some of the problems require "insight", not just application of some fixed set of principles.

So far we have used the time decay functions with respect to the number of attempts. Another option is to use the time decay function with respect to real time or to take at least some aspects of the real time into account, e.g., to consider large pause between attempts (similarly to the approach used in [20]). We have performed experiments with this extension, but the results stay very similar or bring only small improvement (using linear combination of number of attempts and the logarithm of passed time [14]).

In Figure 3 we have evaluated the parameter of time decay function with respect to the problem type. We can do similar analysis with respect to students. If the student's performance is improving fast, then the optimal time decay function for him is steep, i.e., there is some relation between learning and optimal choice of the time decay function. However, this relation is not straightforward, as steep decay function can also mean high autocorrelation without learning, e.g., when the student accesses the educational system from different environments (mouse vs touchpad) or at different conditions (morning vs night).

The results for the Elo system over this dataset are similar and we summarise them only briefly. Even the basic Elo system achieves similar predictions as the model of problem solving times from [10]. The extension of the Elo system that uses the uncertainty function with parameters determined from experiments with simulated data can achieve improvement over the previously published model by 1 to 3 percent in RMSE [24].

For experiments with student models that predict correctness of answers we used an Algebra I dataset from KDD Cup 2010 (binary correctness) and ASSISTment dataset with partial credit data [25] (correctness is a number between 0 and 1 depending on the number of hints used). In both of these datasets each item has a knowledge component assigned and we compute skills for these specified knowledge components.

Although this is different setting and completely different datasets from the previous experiments, the results are very similar (Figure 3). For the choice of a time decay function we have analogical results: exponential and hyperbolic functions work best, sliding window (in both versions) is significantly worse. The choice of optimal parameters for time decay functions is again similar (usually around 0.1 for exponential function) and again we observe differences between different skills (knowledge components). For generic skills (like "Identifying units", "Entering a given" in Algebra), time decay does not bring an improvement. For specific skills (like removing constant in linear equation), the optimal time decay function is steep and improves performance, i.e., for these skills there is significant learning and hence it pays to give large weight to recent attempts.

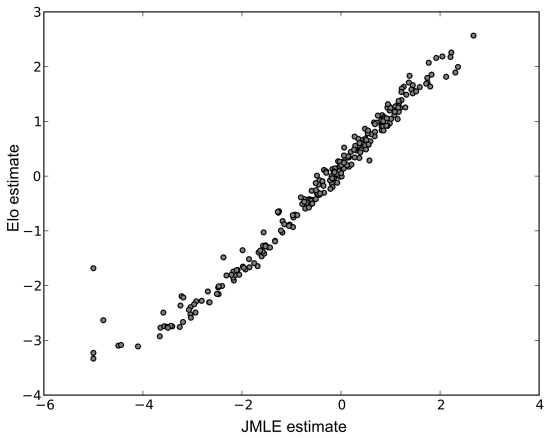

Figure 4: Comparison of JMLE and Elo estimates of difficulty of geography items (country names).

For the Elo system in the context of correctness of answers, we have applied and evaluated the system in an educational application for learning geography (names of countries) – `slepemapy.cz`. We use the Elo system to estimate the prior geography knowledge of students and difficulty of countries. Similarly to the above reported experiments with simulated data, the Elo model (with uncertainty function) achieves very similar results as the joint maximum likelihood estimation for the Rasch model (see Figure 4). The Elo system is much faster and more suitable for online application than the iterative JMLE procedure. To estimate the probability of correct answer after a sequence of attempts at a given country we use a model that combines aspects of performance factor analysis [19] and the Elo system. This combined model achieves better results than both standard performance factor analysis and Bayesian knowledge tracing. More details about this application and evaluation are given in [18].

4. DISCUSSION

We have performed experiments in different settings and with different datasets. Basic results are quite consistent. The Elo system and estimation using time decay functions are simple and flexible approaches, which can match more specific models (Rasch model, Bayesian knowledge tracing) even if the data are generated exactly according to the assumptions of the more specific model. For the choice of time decay function, it seems that for student modeling it is most useful to use either exponential or hyperbolic function (our experiments do not show systematic significant difference between these two). Sliding window and linear function within sliding window lead to worse results.

The choice of specific parameters is also quite consistent, e.g., for the exponential time decay function e^{-kt} the best k is usually around 0.1. The differences between problems of the optimal value of the parameter k (i.e., of the shape of time decay function) are related to the speed of learning for a particular problem type (knowledge component). This relation is however not straightforward, because the time decay approach captures not just learning, but also other temporal effects (e.g., autocorrelation of result due to the use of the system from different working environments). For the uncertainty function of the Elo system a good candidate is a hyperbolic function $\frac{a}{1+bx}$, the specific parameters a, b differ according to the exact application, but the values can be easily found using a grid search and the performance of the system is only mildly sensitive to the exact values.

The advantages of both studied techniques are their flexibility, small number of parameters, and easiness of application. Flexibility is due to the weak assumptions about student behavior and allows for application in wide variety of contexts – this was demonstrated by wide range of data used in our evaluation (e.g., logic puzzles, math problems, knowledge of country names). Small number of parameters reduces the chance of overfitting and leads to stable results. Both techniques are very easy to implement and have low computational demands – predictions are easy to compute, the Elo system and exponential time decay function can be even used in online fashion without storing data about individual student attempts.

Both studied techniques are quite general. Since they do not make any specific assumptions, they should not be expected to bring an optimal performance results for a particular situation. But as we show, they can be easily applied in wide range of situations and provide reasonable performance. Moreover, small improvements in performance (which can be brought by more specific models) are often not practically important for applications of skill estimates. Even if more specific models are available, these simple approaches can be used to get quick insight into the data and should be used in evaluations to judge the merit of more complex models. The basic ideas of the Elo system and time decay functions can also be incorporated into other models, e.g., time decay functions could be quite naturally incorporated into performance factor analysis [19].

Some of the natural features of these approaches can also be useful for intelligent tutoring and adaptive practice. Consider two students with the following history of answers to a particular knowledge component: student A: 1, 1, 1, 1. student B: 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 1, 1, 1, 1. Immediately after these sequences, it is not useful to give any of these two students more problems about this knowledge component, as there is a high probability of a correct answer. But there is clearly a difference between these students – whereas student A probably has solid knowledge and there is little use in returning to the knowledge component in the future, for student B a review in the future would be certainly useful. If we summarise the skill by a single number as is typically done by BKT, it is hard to capture this difference. Using time decay functions, it is easy to cover this situation – we can estimate a "current skill" using a steep time decay function and a "long term skill" with a flat time decay function.

Recently, there has been several works that studied the Elo system in the context of student modeling and adaptive practice [1, 13, 26, 27, 28]. However the impact so far has been rather marginal, particularly compared with Bayesian knowledge tracing. As the discussion above suggest, the approach deserves more attention.

Acknowledgement

The author thanks Matěj Klusáček and Libor Vaněk for performing some of the experiments and for discussions about the topic of the paper.

5. REFERENCES

[1] M. Antal. On the use of elo rating for adaptive assessment. *Studia Universitatis Babes-Bolyai, Informatica*, 58(1), 2013.

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

[3] R. De Ayala. *The theory and practice of item response theory*. The Guilford Press, 2008.

[4] A. E. Elo. *The rating of chessplayers, past and present*, volume 3. Batsford London, 1978.

[5] S. Frederick, G. Loewenstein, and T. O'donoghue. Time discounting and time preference: A critical review. *Journal of economic literature*, 40(2):351–401, 2002.

[6] J. a. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia. A survey on concept drift adaptation. *ACM Comput. Surv.*, 46(4):1–37, 2014.

[7] M. E. Glickman. Parameter estimation in large dynamic paired comparison experiments. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 48(3):377–394, 1999.

[8] R. Herbrich, T. Minka, and T. Graepel. Trueskill: A bayesian skill rating system. In *Advances in Neural Information Processing Systems*, pages 569–576, 2006.

[9] P. Jarušek, M. Klusáček, and R. Pelánek. Modeling students' learning and variability of performance in problem solving. In *Educational Data Mining (EDM)*, pages 256–259. International Educational Data Mining Society, 2013.

[10] P. Jarušek and R. Pelánek. Analysis of a simple model of problem solving times. In *Proc. of Intelligent Tutoring Systems (ITS)*, volume 7315 of *LNCS*, pages 379–388. Springer, 2012.

[11] P. Jarušek and R. Pelánek. A web-based problem solving tool for introductory computer science. In *Proc. of Innovation and technology in computer science education*, pages 371–371. ACM, 2012.

[12] R. Klinkenberg. Learning drifting concepts: Example selection vs. example weighting. *Intelligent Data Analysis*, 8(3):281–300, 2004.

[13] S. Klinkenberg, M. Straatemeier, and H. 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.

[14] M. Klusáček. Modeling learning in problem solving. Master's thesis, Masaryk University Brno, 2013.

[15] Y. Koren. Collaborative filtering with temporal dynamics. *Communications of the ACM*, 53(4):89–97, 2010.

[16] B. Martin, A. Mitrovic, K. R. Koedinger, and S. Mathan. Evaluating and improving adaptive educational systems with learning curves. *User Modeling and User-Adapted Interaction*, 21(3):249–283, 2011.

[17] B. B. Nooraei, Z. A. Pardos, N. T. Heffernan, and R. S. J. de Baker. Less is more: Improving the speed and prediction power of knowledge tracing by using less data. In *Educational Data Mining (EDM)*, pages 101–110, 2011.

[18] J. Papoušek, R. Pelánek, and V. Stanislav. System for learning facts in domains with prior knowledge. In *Educational Data Mining (EDM)*, 2014. To appear.

[19] P. I. Pavlik, H. Cen, and K. R. Koedinger. Performance factors analysis-a new alternative to knowledge tracing. In *Artificial Intelligence in Education (AIED)*, pages 531–538. IOS Press, 2009.

[20] Y. Qiu, Y. Qi, H. Lu, Z. A. Pardos, and N. T. Heffernan. Does time matter? modeling the effect of time with bayesian knowledge tracing. In *Educational Data Mining (EDM)*, pages 139–148, 2011.

[21] A. Tsymbal. The problem of concept drift: definitions and related work. *Computer Science Department, Trinity College Dublin*, 2004.

[22] B. van de Sande. Properties of the bayesian knowledge tracing model. *Journal of Educational Data Mining*, 5(2):1, 2013.

[23] W. Van Der Linden. Conceptual issues in response-time modeling. *Journal of Educational Measurement*, 46(3):247–272, 2009.

[24] L. Vaněk. Elo systém a modelování času řešení. Master's thesis, Masaryk University Brno, 2014. To appear.

[25] Y. Wang and N. Heffernan. Extending knowledge tracing to allow partial credit: Using continuous versus binary nodes. In *Artificial Intelligence in Education (AIED)*, volume 7926 of *LNCS*. Springer, 2013.

[26] K. Wauters, P. Desmet, and W. Van Den Noortgate. Adaptive item-based learning environments based on the item response theory: possibilities and challenges. *Journal of Computer Assisted Learning*, 26(6):549–562, 2010.

[27] K. Wauters, P. Desmet, and W. Van Den Noortgate. Monitoring learners' proficiency: Weight adaptation in the elo rating system. In *Educational Data Mining (EDM)*, pages 247–252, 2011.

[28] K. Wauters, P. Desmet, and W. Van Den Noortgate. Item difficulty estimation: An auspicious collaboration between data and judgment. *Computers & Education*, 58(4):1183–1193, 2012.

[29] G. I. Webb and M. Kuzmycz. Evaluation of data aging: A technique for discounting old data during student modeling. In *Intelligent Tutoring Systems (ITS)*, pages 384–393. Springer, 1998.