T. Hirashima et al. (Eds.) (2011). Proceedings of the 19th International Conference on Computers in Education. Chiang Mai, Thailand: Asia-Pacific Society for Computers in Education

Towards a Bayesian Student Model for Detecting Decimal Misconceptions

George GOGUADZE[a*], Sergey SOSNOVSKY[b] , Seiji ISOTANI[c] & Bruce M. MCLAREN[c]
[a]*Saarland University, Germany*
[b]*German Research Center for Artificial Intelligence, Germany*
[c]*Carnegie Mellon University, USA*
george@activemath.org

Abstract: This paper describes the development and evaluation of a Bayesian network model of student misconceptions in the domain of decimals. The Bayesian model supports a remote adaptation service for an intelligent tutoring system within a project focused on adaptively presenting erroneous examples to students. We have evaluated the accuracy of the student model by comparing its predictions to the outcomes of the interactions of 255 students with the software. Students' logs were used for retrospective training of the Bayesian network parameters. The accuracy of the student model was evaluated from three different perspectives: its ability to predict the outcome of an individual student's answer, the correctness of the answer, and the presence of a particular misconception. The results show that the model is capable of producing predictions of high accuracy (up to 87%).

Keywords: Student model evaluation, Bayesian networks, Bayesian student modeling

Introduction

The quality of an adaptive educational system (AES) strongly depends on the quality of its student modeling. The system might implement a solid adaptation strategy and provide students with well-designed learning content, but if its estimation of students' knowledge is incorrect, the adaptive interventions it produces are unlikely to be effective. In recent years, significant efforts have been expended to develop a methodology for layered evaluation of AES that allows examining student modeling components in isolation [4, 14]. Various approaches have been used for measuring the goodness of a particular student modeling mechanism [15], guiding the improvement of a student model (SM) [10] or selecting the best SM configuration among several alternatives [17]. All these evaluation studies have been based on rigorous analyses of students' logs generated by the systems.

In this paper, we describe the development of a Bayesian network (BN) SM and a data mining study aimed at validating its quality. The model represents students' misconceptions in the domain of decimals. It was designed within the framework of the AdaptErrEx project (*http://www.cs.cmu.edu/~bmclaren/projects/AdaptErrEx/*), which focuses on presenting and adapting erroneous examples (step-by-step solutions to decimal math problems in which at least one step is incorrect) to remediate students' misconceptions. Besides a general concern for the quality of adaptation, there are at least two other reasons for a thorough validation of the AdaptErrEx SM:

- *The novel modeling approach of the project, as compared to other Bayesian SMs of decimal misconceptions*. Since the student modeling approach of this project is somewhat atypical, it is especially critical that it be carefully verified.
- *The on-going experimental studies of the project, in which the system has and will be evaluated for subtle learning effects on students*. Since different strategies for

presenting erroneous examples will be tried, it must be clear that the model is working as expected.

The evaluation of the model was done based on the data logs of 255 middle school (6^{th} through 8^{th} grade) students working with test problems in the domain of decimals. Data from 70% of the students was used for training model parameters. The remaining 30% of the data was used to compute three different metrics, estimating how well the model can make predictions about the student's state from the following perspectives:

1. predicting the exact answer to the next problem tackled by a student;
2. predicting the correctness of the next answer provided by a student; and
3. predicting the presence of a misconception the student has.

In order to compute these metrics, we compared the predictions of the individual SMs with the students' results on the posttest. Although the values achieved for all three metrics could potentially be improved, they by far exceed the baseline of a random prediction. These results support our belief that the model is capable of accurate adaptation and encourage us to continue investigating ways to improve it.

1. Modeling Students' Misconceptions in AdaptErrEx

BNs are well-established tools for representing and reasoning about uncertainty in student models [2, 5, 11, 12]. Perhaps the closest example to the BN-based SM developed for AdaptErrEx is the SM of the DCT tutor that helped students learn decimal comparisons [16]. In the DCT's model, the misconceptions were represented as mutually exclusive alternatives of two probabilistic nodes identifying the "coarse" and the "fine" class of misconceptions. The "coarse" class represents the basic judgments that a student uses when comparing decimals (e.g. "longer decimals are larger") and the "fine" class represents possible misconceived reasons for such judgments (e.g. "because longer integers are larger"). The causal relation between the two nodes is modeled as a set of conditional probabilities defining the chance a student will come up with a basic judgment if she has a particular finer-grained misconception. The evidence nodes representing learning tasks are conditionally dependent on both misconception nodes.

A different approach to domain and student modeling with the help of BNs is described in [6]. The domain model here is represented as a hierarchy of skills, where the probability of mastering a super-skill is conditionally dependent on mastery of the sub-skills. The bottom-level skills are probabilistically connected with the evidence nodes, which are test questions.

In AdaptErrEx we have followed an approach that is a combination of these two prior approaches. Based on the results of an extensive literature review of students' learning of decimals, we identified the most frequently occurring decimal misconceptions. Then, we organized these misconceptions into a taxonomy based on their differences and similarities [7]. The resultant taxonomy attempts to structure the rationale behind the students' incorrect responses to decimal problems and, thereby, provides a means for diagnosing students' learning difficulties.

In order to account for dependencies between misconceptions, a BN was built, where each misconception is represented by a probabilistic node with two possible alternatives (present/absent). The taxonomic relations between the nodes are accompanied by tables of conditional probabilities. These conditional/hierarchical links mean that if a misconception M2 is a sub-node of a misconception M1, the probability of a student having M1 depends on whether she has M2. Figure 1 shows a fragment of the AdaptErrEx's BN. It presents two misconception nodes, *Regz* and *Megz*, connected with a taxonomic relation, and several problem nodes linked to them. *Regz* represents the misconception "*decimals treated as regular numbers*"; having such a misconception can lead to solutions like this: $0.09 > 0.1$

(ignore leading zeros, as with "regular" integer numbers). *Megz* represents the misconception "*longer decimals are larger*"; this misconception can lead to solutions like this: 0.123 > 0.45. According to this representation, if a student thinks that decimals are the same as integers, she is more likely to believe that longer decimals are always larger.

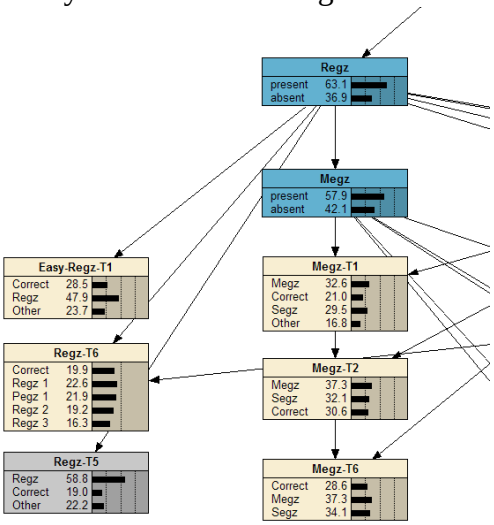

Fig. 1. Bayesian network of students' misconceptions in AdaptErrEx

Another node type in the network is the evidence nodes representing problems (bottom nodes in the taxonomy such as Easy-Regz-T1 and Megz-T2). They can be connected to one or more misconceptions. The evidence nodes contain several alternatives, where each alternative corresponds to a possible answer the student might give to the problem. Every evidence node alternative is probabilistically connected to the corresponding misconception node alternatives. This means that presence/absence of a misconception influences the likelihood of a student giving a certain answer to the problem. Consider, for example, the problem *Megz-T6* (Fig. 2). If a student chooses the second option as an answer she produces evidence of having the misconception *Megz* described above (i.e., thinking that 0.37 > 0.5 because 0.37 is longer than 0.5), whereas the third alternative is evidence for a different misconception – *Segz*, which is present when a student thinks that shorter decimals are larger (e.g., thinking that 0.5 > 0.713 because 0.5 is shorter than 0.713). The node alternative labeled as *Correct* represents the correct answer and contributes negatively to the presence of all the misconceptions, connected to the given problem. Finally, the node labeled as *Other* represents any incorrect answer not known to be connected to any misconception. It is important to mention that the relations between the alternatives of the evidence nodes and the alternatives of the misconception nodes are not deterministic. Multiple misconceptions can be responsible for a student choosing a particular incorrect answer, or the student may have had a slip. This is similar to what mathematics education literature has shown: Often students have multiple interacting misconceptions.

The problems connected to the BN are clustered in 4 types:

- open questions where students input their answers freely (e.g. 1.4 + 0.7 = ?),
- ordering decimals by moving them in a list of numbers or on a number line,
- multiple choice with a story context (such as shown in Figure 2), and
- multiple choice without a story context (e.g., Which is the greatest number? 0.237 0.56 0.9).

These problems were developed based on the analysis of the mathematical educational literature [8]. In particular, we created/selected problems that: (a) have been shown to be difficult to solve in prior studies; (b) received students' answers indicating evidence of misconceptions from the designed taxonomy; (c) require more than procedural knowledge to answer correctly.

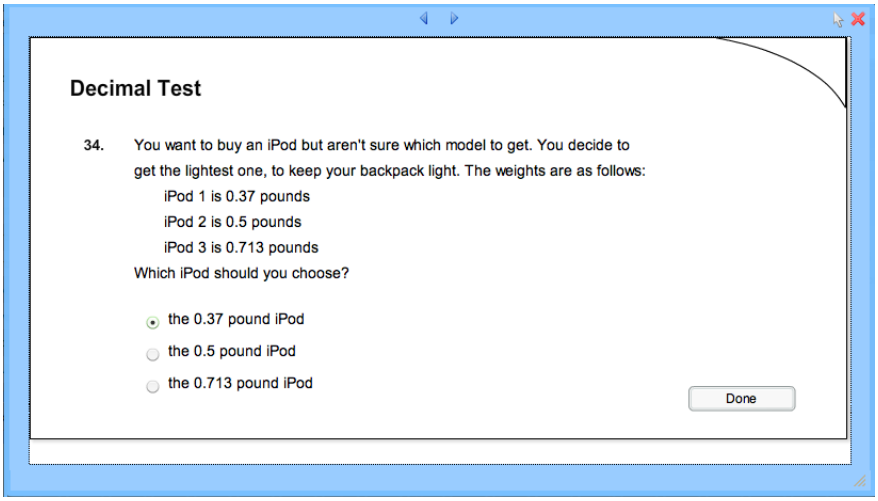

Fig. 2. An example of a problem on decimal arithmetic (*Megz-T6* from Figure 1)

Overall, the developed network contains twelve misconception nodes, where seven nodes represent the most typical decimal misconceptions (such as *Megz* and *Regz*) and the five nodes serve as aggregation nodes. The concrete misconception nodes are connected to 126 evidence nodes representing problems. These problems are divided into three isomorphic problem sets (set A, set B and set C), each set containing 42 problems.

In order to ensure that the results are not driven by subtle differences in the problems, the roles of problem sets A, B and C were counterbalanced across student groups. Each of the problem sets was used either as a pretest, an immediate posttest, or a delayed posttest. In total, there are six possible combinations of the problem sets (*ABC*, *ACB*, *BAC*, *BCA*, *CAB* and *CBA*) depending on the role each set plays for each student. Consequently, students were randomly assigned to one of the six groups, facing one of the six possible sequences of tests.

2. Evaluating The Accuracy of Model's Predictions

This section summarizes our approach to evaluating the capability of the designed BN to predict the effective state of student's learning. The approach consists of three steps:

- **training the domain model**: the initial BN parameters were trained based on the pretest data from the 70% of users; the resulting BN represents the initial probabilities of decimal misconceptions for a student from the target population.
- **learning student models**: the logs of students from the testing set (the remaining 30%) were used to update the domain model; the obtained collection of BNs represent individual SMs each reflecting the assumptions of the system about the misconceptions of the corresponding student from the testing set;
- **evaluating the accuracy of the model's predictions**: based on the individual SMs populated on the second step, three different metrics are computed and evaluated (1) predicting students' next answer; (2) predicting the correctness of the answers; (3) predicting students' misconceptions; each metric evaluates the capability of the model to make a particular prediction.

The rest of this section describes these steps in more details.

2.1 Training the Domain Model

Parameter estimation is a well-known challenge in the field of BNs. In our case, these parameters include prior probabilities for misconception nodes and conditional probability

tables for links between the nodes. For completing this task, we have applied a conventional approach. We supplied the initial estimations of network parameters, following the logical choice of values. For example, if an alternative of a problem node is witnessing an occurrence of a misconception, the conditional probability between this alternative and the "present" alternative of a misconception is high. Then the training algorithm was applied to refine the parameters. In presence of a sufficient amount of training data, the initial estimations of network parameters should not play a significant role for their final values.

For the training set we randomly selected 70% of the students participating in the study. Based on the pretest logs of these students, the prior probabilities for misconception nodes and the conditional probabilities for evidence nodes of all three problem sets A, B and C are computed. Our selection also ensured that (random) representatives of each of the six constellations of tests/problem sets are uniformly distributed in the training data. As our goal was to obtain the domain model, we used only the results of the pretest to train the network. In this way, the resulting BN represents the initial state of knowledge of decimals (more specifically, the state of misconceptions) for a typical student from the target population. The prior probabilities of misconception nodes quantify how likely such an average student is to have a particular misconception. The conditional probabilities encode the strength of a causal relation among misconceptions and between the misconceptions and the problems.

2.2 Learning Student Models for the Testing Set

After the initial training/calibration, the BN was ready to learn the models of individual students. In order to do this, we fed the activity logs of the remaining 30% of the students to the network. Only their answers to the pretest and immediate posttest were used on this step. This evidence back-propagated to the relevant misconception nodes and updated their posterior probabilities, thus individualizing the networks. The resulting collection of BNs contained individual misconception models of every student in the testing set. Each resulting individual SM took into account both the collective traits of the target population and the history of idiosyncratic behavior of the corresponding student.

2.3 Estimating Predictive Validity of the Student Model

The BNs obtained in Step 2 can be used to make individual predictions about students. Those can be predictions of a student having a misconception, or a student answering a problem in a certain way. Based on such predictions, an AES could control the individual learning experiences of its students. We identified three types of these predictions and tried to verify their average accuracy. In order to do so, we compared the predictions made by the individually trained BNs of the students from the training set with their results on the delayed posttest. The three prediction types were: predicting the *actual student answer*, predicting the *correctness of the student answer*, and predicting the *presence of a student misconception*. The notion of accuracy in these three cases was defined as follows:

I. A prediction of the *actual student answer* is accurate (inaccurate otherwise) if the alternative chosen by a student for a posttest problem had the highest probability in the BN trained for this student in Step 2. The corresponding metric is computed as a percentage of accurate predictions.
II. A prediction of the *correctness of the student's answer* is accurate in any of the cases (inaccurate otherwise) if:
   – the student gives the correct answer to a delayed posttest problem and the probability of the correct alternative for this problem's node is maximum in the BN trained for this student in Step 2;

– the student gives an incorrect answer to a delayed posttest problem and the probability of the correct alternative for this problem's node is less than the sum of probabilities of incorrect alternatives in the BN trained for this student in Step 2.

The corresponding metric is computed as a percentage of accurate predictions.

III. A prediction of the *presence of a misconception* is defined as follows. Based on the state of a misconception node, the student is believed to have a corresponding misconception if its probability is greater than 0.5. This prediction is considered accurate if during the delayed posttest the student has shown more evidence of having the misconception than not having it (and vice-versa). The evidence is quantified as an average rate of misconception occurrence in the students' answers in the delayed posttest. The average rate of misconception occurrence is computed in the following way:

– The occurrence of the answer option that represents evidence for the given misconception contributes positively to the occurrence of the misconception;
– The correct answer to the problem whose answer options represent evidence for the misconception contributes negatively to the occurrence of the misconception;
– Incorrect answers not connected to the given misconception do not contribute to the diagnosis.

Let *P(M)* be the probability of the presence of a misconception *M* in a Bayesian model, *Npos(M)* – the number of student's answers that provide evidence for the misconception M, *Nneg(M)* – the number of correct answers to the problems that can diagnose *M*, and *N(M)* – the total number of problems that address this misconception. Then, the model prediction is said to be accurate if and only if:

$$\left(P(M) \geq 0.5\right) \& \left(\frac{Npos(M)}{N(M)} \geq 0.5\right) \vee \left(P(M) < 0.5\right) \& \left(\frac{Nneg(M)}{N(M)} \geq 0.5\right)$$

Note that although it is in general difficult to verify that the student has a misconception by analyzing his answers to problems, the exercises in this experiment are designed specifically for this purpose, so that the answer options have a high diagnostic value and hence the average occurrence rate defined above provides a realistic estimate.

3. Experiment Settings and Evaluation Results

The data for the evaluation came from a classroom empirical study conducted in a Pittsburgh, PA (U.S.A.) middle school during the fall of 2010. Overall, 255 students from 6^{th}-8^{th} grades participated in the study. The study had several sessions, in which the students took a pretest, solved the treatment problems, took an immediate posttest, and (one week later) a delayed posttest. As explained in section 1, the 126 test problems were split into 3 isomorphic problem sets (A, B, and C) and the roles of these problem sets being pretest, posttest or delayed posttest were counterbalanced across student groups. The learning materials came from the domain of decimals. The MathTutor web-based system was used to deliver the materials to the participants [1]. Students took 4 to 5 sessions to complete all of the materials.

MathTutor logs all students' interactions, as well as diagnostic information in the PSLC DataShop storage [9]. After cleaning the DataShop logs (e.g. removing the data of the students who did not complete all tests) and filtering events generated by the system, we obtained the data that served as a basis for our evaluation. In total, we analyzed 31,049 student interaction events which resulted from each of the 255 students solving up to 126 problems. Each of these events contained several parameters, such as an identifier of the user and the problem tackled, as well as the user input and the diagnosis, i.e., the correctness of the answer or an evidence for a particular misconception.

As specified in section 2.2, the accuracy values were calculated for the testing set, comprising 30% of students randomly chosen from the total of 255 students (77 students; data from these students was not used in the training phase). Using the metrics defined in Section 2.3, we evaluated the accuracy of the predictions of our SM. The average accuracies for predicting concrete answers, and the correctness of students' answers were computed in the following way:

- the list of prediction accuracies for all problems was computed for each student, where each accuracy is computed using the corresponding algorithm, described in Section 2.3;
- an average accuracy of predictions for each student was computed by dividing the sum of individual accuracies by the number of problems solved by the student;
- the list of average accuracies for all students has been averaged again by the number of students.

As the result of the calculation, the average accuracy of *predicting the actual answer* of the students in the delayed posttest was *60%*, whereas the average accuracy of *predicting the answer correctness* was *69%*.

The accuracy of predicting the occurrence of (evidence for) misconceptions was computed as follows:

- for each student and each misconception an average accuracy of predicting the evidence occurrence for this misconception is computed using the algorithm described in Section 2.3;
- the list of average occurrences of misconceptions is stored for each student;
- for each misconception, we compute the final average value by taking the average of the values for all students;
- finally, an average of averages for all misconceptions is taken.

The average accuracy of *predicting misconceptions* was *87%*. The accuracy of predictions of each misconception separately does not significantly vary from the total average.

Similar studies on evaluating the accuracy of predictions of a BN student model of the DCT tutor [13], have shown 80-90% accuracy for predicting misconceptions. For comparison, each misconception in our model is represented as a separate node whereas in DCT tutor there are only two nodes for the coarse and fine classes of student misconceptions. Therefore, our model has much less evidence nodes for each misconception which might result into lower precision of predictions.

4. Conclusions and Future Work

We have presented the development and evaluation of a Bayesian approach to modeling student misconceptions. We have computed three different metrics estimating how well the model can make predictions about the student's knowledge state, in particular his misconceptions. The results of this evaluation show that for all three metrics the student model predictions are reasonably high and accurate, yet leaving room for improvement.

Future work is planned in two main directions: improving the *structure of the Bayesian model*, and enhancing the *methods of evaluation* of the model validity. We plan to experiment with different configurations of BNs, derived from the original one, such as dynamic BNs, and the networks with soft evidence nodes.

When adjusting the evaluation method we could experiment with additional parameters of the students such as gender, grade, or general math skills. Difficulty of the problems could be used here as well as an additional parameter in the computation of the accuracy metrics. For example, if the problem is very easy, the student is likely to solve it correctly

even if the probability of having a misconception is high, and the other way round, difficult problems can be solved incorrectly even if the probabilities of misconceptions are low.

Acknowledgements

The U.S. Department of Education Institute of Education Sciences, grant# R305A090460, funded this work.

References

[1] ALEVEN, V., MCLAREN, B., & SEWALL, J., 2009. Scaling up programming by demonstration for intelligent tutoring systems development: An open-access website for middle school mathematics learning. *IEEE Transactions on Learning Technologies*, 2(2), 64-78.

[2] ARROYO, I. & WOOLF, B. P., 2005. Inferring learning and attitudes from a Bayesian Network of log file data. In *Proceeding of the International Conference on Artificial Intelligence in Education*, IOS Press, Amsterdam, The Netherlands, 33-40.

[3] BONEH, T., NICHOLSON, A., SONENBERG, L., STACEY, K., & STEINLE, V., 2003. Decsys: An intelligent tutoring system for decimal numeration. *Tech Report 134*, School of CSSE, Monash Univ., Australia.

[4] BRUSILOVSKY, P., KARAGIANNIDIS, C., AND SAMPSON, D., 2004. Layered evaluation of adaptive learning systems. *International Journal of Continuing Engineering Education and Lifelong Learning* 14 (4/5), 402 – 421.

[5] CONATI, C., GERTNER, A. AND VANLEHN K., 2002. Using Bayesian networks to manage uncertainty in student modeling. *Journal of User Modeling and User-Adapted Interaction*, vol. 12(4), p. 371-417

[6] COLLINS, J., GREER, J., HUANG, S. (1996). Adaptive assessment using granularity hierarchies and Bayesian nets. In*: Lecture Notes in Computer Science*. Vol. 1086. pp. 569–577.

[7] ISOTANI, S., MCLAREN, B., & ALTMAN, M., 2010. Towards intelligent tutoring with erroneous examples: A taxonomy of decimal misconceptions. Proceedings of the International Conference on Intelligent Tutoring Systems. *Lecture Notes in Computer Science 6095*, Springer, 346-348..

[8] ISOTANI, S., ADAMS, D., MAYER, R. E., DURKIN, K., RITTLE-JOHNSON, B., MCLAREN, B., submitted. Can Erroneous Examples Help Middle-School Students Learn Decimals? Submitted to the *European Conference on Technology Enhanced Learning*.

[9] KOEDINGER, K., BAKER, R., CUNNINGHAM, K., SKOGSHOLM, A., LEBER, B., STAMPER, J., 2010. A Data Repository for the EDM community: The PSLC DataShop. In Romero, C., Ventura, S., Pechenizkiy, M., Baker, R.S.J.d. (Eds.) *Handbook of Educational Data Mining*. Boca Raton, FL: CRC Press, 43-53.

[10] MARTIN, B., MITROVIC, A., KOEDINGER, K., & MATHAN, S., 2011. Evaluating and Improving Adaptive Educational Systems with Learning Curves, *Journal of User Modeling and User Adapted Interaction*, 21(3), Springer.

[11] MADHYASTHA, T. M., & HUNT, E., 2010 MINING FOR PATTERNS OF INCORRECT RESPONSE IN DIAGNOSTIC ASSESSMENT DATA. IN ROMERO, C., VENTURA, S., Pechenizkiy, M., Baker, R.S.J.d. (Eds.) *Handbook of Educational Data Mining*. Boca Raton, FL: CRC Press, 427-439.

[12] MILLÁN, E., LOBODA, T., & PÉREZ-DE-LA-CRUZ, J., 2010. Bayesian networks for student model engineering, *Computers & Education*, Vol. 55, Issue 4, 1663-1683.

[13] Nicolson, A., Boneh, T., Wilkin, T. , Stacey, K., Sonenberg, L., & Steinle, V., 2001. A case study in knowledge discovery and elicitation in an intelligent tutoring application. In proceedings of the 17th Conference on Uncertainty in AI, pp. 386-394, Seattle.

[14] PARAMYTHIS, A., & WEIBELZAHL, S., 2005. A decomposition model for the Layered Evaluation of Interactive Adaptive Systems. In Proceedings of the 10th International Conference on User Modeling, Lecture Notes in Artificial Intelligence LNAI 3538, (pp. 438-442). Springer.

[15] SOSNOVSKY, S., & BRUSILOVSKY, P., 2005. Layered evaluation of topic-based adaptation to student knowledge. In Proc. of 4th Workshop on the Evaluation of Adaptive Systems at UM 2005, 47-56.

[16] STACEY, K., SONENBERG, E., NICHOLSON, A., BONEH, T., & STEINLE, V., 2003. A teacher model exploiting cognitive conflict driven by a Bayesian network. In Peter Brusilovsky, Albert T. Corbett, Fiorella De Rosis (Eds), *User Modeling 2003: Proceedings of the Ninth International Conference*. (pp. 352-362) New York: Springer-Verlag (ISBN 3540403817).

[17] YUDELSON, M., MEDVEDEVA, O., AND CROWLEY, R., 2008. A multifactor approach to student model evaluation. *User Modeling and User-Adapted Interaction*, 18(4), 349-382.