The Effect of Model Granularity on Student Performance Prediction Using Bayesian Networks

Zachary A. Pardos, Neil T. Heffernan, Brigham Anderson, Cristina L Heffernan

Worcester Polytechnic Institute, Carnegie Mellon University, Worcester Public Schools
{zpardos, nth}@wpi.edu

Abstract. A standing question in the field of Intelligent Tutoring Systems and User Modeling in general is what is the appropriate level of model granularity (how many skills to model) and how is that granularity derived? In this paper we will explore varying levels of skill generality within 8th grade mathematics using models containing 1, 5, 39 and 106 skills. We will measure the accuracy of these models by predicting student performance within our own tutoring system called ASSISTment as well as their performance on the Massachusetts standardized state test. Predicting students' state test scores will serve as a particularly stringent real-world test of the utility of fine-grained modeling. We employ the use of Bayes nets to model user knowledge and for prediction of student responses. The ASSISTment online tutoring system was used by over 600 students during the school year 2004-2005 with each student using the system 1-2 times per month throughout the year. Each student answered over 100 state test based items and was tutored by the system with help questions called scaffolding when they made a mistake. Each student answered on average 160 scaffold questions. Our results show that the finer the granularity of the skill model, the better we can predict student performance for our online data. However, for the standardized test data we received, it was the 39 skill model that performed the best. We view the results as support for using fine-grained models even though the finest-grained sized model did not also predict the state test results the best.

1 Introduction

There are many researches in the user modeling community working with Intelligent Tutoring Systems (ITS) (i.e, Mayo & Mitrovic [12], Corbett, Anderson et al, [6], Conati & VanLehn [5], Woolf [2]) and many who have adopted Bayesian network methods for modeling knowledge [15, 4, 11]. Even methods that were not originally thought of as Bayesian Network methods turned out to be so; Reye [14] showed that the classic Corbett & Anderson's "Knowledge tracing" approach was a special case of a dynamic belief network.

We seek to address the question of what is the right level of granularly to track student knowledge. Essentially this means how many skills should we attempt to track? We will call a mapping of skills to questions a skill model. We will compare different skill models that differ in the number of skills and see how well the different models can fit a data set of student responses collected via the ASSISTment system [7]. We are not the first to do model-selection based on how well the model fits real student data (i.e., [9, 11]). Nor are we the only ones that have been concerned with the question of granularity; Greer and colleagues [10, 15] have investigated method of using different levels of granularity, and different ways to conceptualize student knowledge. We are not aware of any other work where researchers attempted to specifically answer the question of "what is the right level of granularity to best fit a data set of student responses".

1.1 The Massachusetts Comprehensive Assessment System (MCAS)

The MCAS is a Massachusetts state administered standardized test that covers English, math, science and social studies for grades 3rd through 10th. We are focused on 8th grade mathematics only. Our work relates to the MCAS in two ways. First we have built our content based upon ~300 publicly released items from previous MCAS math tests. Secondly, we will be evaluating our models by using the 8th grade 2005 MCAS math test which was taken after the online data being used was collected.

1.2 Background on the ASSISTment Project

The ASSISTment system is an e-learning and e-assessing system [7]. In the 2004-2005 school year more than 600 students used the system about once every two weeks as part of their regular classroom curriculum. Eight math teachers from two schools would bring their students to their computer lab, at which time students would be presented with randomly selected question items. Each tutoring item, which we call an ASSISTment, is based upon a publicly released MCAS item which we have added "tutoring" to. If students get the item correct they are advanced to the next question. If they answer incorrectly, they are provided with a small "tutoring" session where they are asked to answer a few questions that break the problem down into steps. The first scaffolding question appears only if the student gets the item wrong. We believe that the ASSISTment system has a better chance of showing the utility of fine-grained skill modeling due to the fact that we can ask scaffolding questions that break the problem down into parts and allow us to tell if the student got the item wrong because they did not know one skill versus another. Most MCAS questions that were presented as multiple-choice were converted into text-input questions to reduce the chance of guess. As a matter of logging, the student is only marked as getting the item correct if they answer the question correctly on the first attempt.

2 Models

We define a skill model as a set of skill names and a mapping of those skill names to questions and scaffolding in the ASSISTment tutoring system. The single skill in the coarse grain model called the WPI-1 represents all of 8th grade mathematics, while the finest grain model, the WPI-106, breaks the same subject matter into 106 different skills. Bayesian Belief Networks (BBN) provide the framework to represent these skill models in a relatively straight forward fashion. They also provide powerful inference and inspectability which is essential for skill reporting to teachers, students or parents.

2.1 Creation of Fine-Grained Skill Model

In April of 2005, we staged a 7 hour long "coding session", where our subject-matter expert, Cristina Heffernan, with the assistance of the 2nd author, set out to make up skills and tag all of the existing 8th grade MCAS items with these skills. This coding session took place at Worcester Polytechnic Institute (WPI) after most of the tutor interaction had taken place. No student data was used to inform this coding session. There were about 300 released test items to code. Because we wanted to be able to track learning between items, we wanted to come up with a number of skills that were somewhat fine-grained but not too fine-grained such that each item had a different skill. We therefore imposed upon our subject-matter expert that no one item would be tagged with more than 3 skills. She gave the skills names, but the real essence of a skill is what items it was tagged to. To create the coarse-grained models we used the fine-grained model to guide us. For the WPI-5 model we started off knowing that we would have the 5 categories; 1) Algebra, 2) Geometry, 3) Data Analysis & probability, 4) Number Science and 5) Measurement. Both the National Council of Teachers of Mathematics and the Massachusetts Department of Education use these broad classifications as well as a 39 skill classification. After our 600 students had taken the 2005 state test, the state released the items from the test and we had our subject matter expert tag up those test items. Shown bellow, in Figure 1 is a graphical representation of two of the skill models we used to predict the 2005 state test items. The 1 and 5 skills are at the top of each graph and the 29 questions of the test are at the bottom. The intermediary nodes are logic gates which are described in the next subsection.

Fig 1.a – WPI-1 MCAS Model

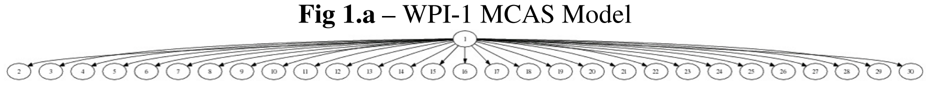
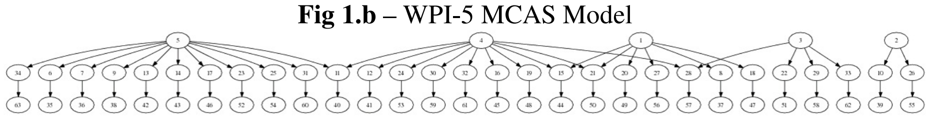


Fig 1.b – WPI-5 MCAS Model



It is the case that with the WPI-39 and WPI-106 models, many of the skills do not show up on the final test since each year only a subset of all the skills needed for 8th grade math are represented.

The WPI-1, WPI-5 and WPI-39 models are derived from the WPI-106 model by nesting a group of fine-grained skills into a single category. This mapping is an aggregate or "is a part of" type of hierarchy as opposed to a prerequisite hierarchy [4]. Figure 2 shows the hierarchal nature of the relationship between WPI-106, WPI39, WPI-5 and WPI-1.

Figure 2. – Skill Transfer Table

WPI-106	WPI-39	WPI-5	WPI-1
Inequality-solving → Equation-Solving → Equation-concept →	setting-up-and-solving-equations	Patterns-Relations-Algebra	The skill of “math”
Plot Graph →	modeling-covariation		
X-Y-Graph → Slope →	understanding-line-slope-concept		
Congruence → Similar Triangles →	understanding-and-applying-congruence-and-similarity	Geometry	
Perimeter → Circumference → Area →	using-measurement-formulas-and-techniques	Measurement	

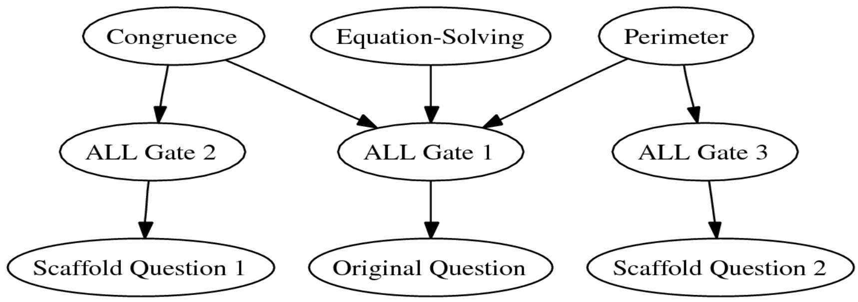
2.2 How the Skill Mapping is Used to Create A Bayes Net

In a typical ASSISTment, an original question will be tagged with a few skills, but if the student answers the original question incorrectly they are given scaffolding questions that are tagged with only a single skill. This gives the system a good chance inspecting which skills a student does not know in the case that they get the original question wrong. Figure 3 shows an example part of the Bayes Net. Each circle is a random Boolean variable. The circles on the top row are variables representing the probability that a student knows a given skill, while the circles on the bottom row are the actual question nodes. The original question in this example is tagged with three skills, *scaffold question 1* is tagged with congruence and *scaffold question 2* is tagged with Perimeter. The ALL[1] gates assert that the student must know all skills relating to a question in order to answer correctly. The ALL gates also greatly simplify the network by reducing the number of parameters specified for the question nodes to just two (guess and slip). The prior probabilities of the skills are shown at the top and the conditional probabilities of getting the questions correct are shown at the bottom of the figure. Note that these parameter values were set intuitively (if a student knows all the skills for an item there will be a 0.95 chance they will get the question correct, but only a 0.10 chance otherwise). This specifies a 10% guess and 5% slip (calculated by 1 - P(Q) | Gate). A prior probability of 0.50 on the skills asserts that the skill is just as likely to be known as not know previous to using the ASSISTment system. When we later try to predict MCAS questions, a guess value of 0.25 will be used to reflect the fact that the MCAS items being predicted are all multiple choice, while the online ASSISTment items have mostly been converted from multiple-choice to “text-input fields”. This model is simple and assumes all skills are as equally likely to be known prior to being given any evidence of student responses, but once we present the network with evidence it can quickly infer probabilities about what the student knows.

[1] The term ‘ALL’ gate is used instead of ‘AND’ gate because our software implementation of Bayesian networks uses AND gates only for nodes with two parents.

Figure 3. – Sample of Bayes Directed Graph with default priors and parameters

P(Congruence)	P(Equation-Solving)	P(Perimeter)
0.50	0.50	0.50



Gate	P(Q)	Gate	P(Q)	Gate	P(Q)
True	0.95	True	0.95	True	0.95
False	0.10	False	0.10	False	0.10

3 Bayesian Network Application

We created a Bayesian framework using MATLAB and Kevin Murphy's Bayes Net Toolkit (BNT) [(http://bnt.sourceforge.net/)] with Chung Shan's BIF2BNT utility. This framework assesses the skill levels of students in the ASSISTment system and measures the predictive performance of the various models. First the skill model, which has been converted into Bayesian Interchange Format from our database, is loaded into MATLAB. A student-id and Bayesian model are given as arguments to our prediction program. The Bayesian model at this stage consists of skill nodes of a particular skill model which are appropriately mapped to the over 1,400 question nodes in our system (300 original questions + 1,100 scaffolds). This can be referred to as the online model. We then load the user's responses to ASSISTment questions from the database and enter their responses into the Bayesian network as evidence. Using join-tree exact inference, a significant improvement over the sampling likelihood-weighting algorithm previously employed [13], posterior marginal probabilities are calculated for each skill in the model for that student.

We now discuss how student performance prediction is done. After the probabilistic skill levels of a particular student have been assessed using the specified skill model, we load a Bayes model of the MCAS test which is also tagged according to the skill model used for the online model. The MCAS test model looks similar to the training model, with skill nodes at top mapped to ALL nodes, mapped to question nodes. In this case we take the already calculated marginal probabilities of the skill nodes from the online model and import them as soft, probabilistic evidence in to the test model. Join-tree inference is then used to get the marginal probabilities on the questions. The probabilities for all 29 questions are summed to produce the final predicted score.

4 RESULTS

An early version of the results in this section (using approximate inference instead of exact inference and without Section 4.2) appears in a workshop paper [13]. Before we present the results we will provide an example, in Table 1, of how we made some of the calculations. To predict each of the 29 questions (rows) we used the skills associated with the question to ask the Bayes Net what the probability is that the user will get the question correct. Question three has two skills, and it consistently viewed as harder by each of the students' (columns). We get a *predicted score* by taking the sum of the probabilities for each question and then taking the ceiling of that to convert it into a whole number. Finally, we find the percent error by taking the absolute value of the difference between predicted and actual score and dividing that by 29. The *Average Error* of 17.28% is the average error across the 600 students for the WPI-5. We repeat this procedure for the WPI-1, WPI-5, WPI-39 and WPI-106 models in Table 2.

Test Question	Skill Tagging (WPI-5)	user 1 P(q)	user 2 P(q)	…	user 600 P(q)	Average Error
1	Patterns	0.2	0.9	…	0.4	
2	Patterns	0.2	0.9	…	0.4	
3	Patterns & Measurement	0.1	0.5	…	0.2	
4	Measurement	0.8	0.8	…	0.3	
5	Patterns	0.2	0.9	…	0.4	
::	::	::	::		::	
29	Geometry	0.7	0.7	…	0.2	
	Predicted Score	14.2	27.8	…	5.45	
	Actual Score	18	23	…	9	
	Error	10.34%	17.24%	…	12.24%	**17.28%**

Table 1. Tabular illustration of prediction calculation and error for the MCAS model.

4.1 MCAS Prediction Results

The prediction results in Table 2 are ranked by error rate in ascending order. The error rate represents how far off, on average, the prediction of student test scores were for each model. The MAD score is the mean absolute deviance or the average raw point difference between predicted and actual score. The under/over prediction is our predicted average score minus the actual average score on the test. The actual average score will be the same for all models. The centering is a result of offsetting every user's predicted score by the average under/over prediction amount for that model and recalculating MAD and error percentage. WPI-5, for example, under predicts student scores by 3.6 points on average. For the centered calculations we add 3.6 points to every predicted score of users in that model and recalculate MAD and error. The choice was made to calculate centered scores for a few reasons: 1) student might take the MCAS test situation more seriously than weekly usage of the ASSISTment system, 2) we would expect to be under-predicting since we are using data from as far back as September to predict a test in May and our model, at present, does not track learning over time. Although the centering method also obscures the differences between models, it is used as a possible score to expect after properly modeling the factors mentioned above.

Model	Error	MAD Score	Under/Over Prediction	Error (After Centering)	Centered MAD Score
WPI-39	12.86%	3.73	↓ 1.4	12.29%	3.57
WPI-106	14.45%	4.19	↓ 1.2	14.12%	4.10
WPI-5	17.28%	5.01	↓ 3.6	13.91%	4.03
WPI-1	22.31%	6.47	↓ 4.3	18.51%	5.37

Table 2. Model prediction performance results for the MCAS test. All models' non-centered error rates are statistically significantly different at the $p<.05$ level.

4.2 Internal/Online Data Prediction Results

To answer the research question of how well these skill sets model student performance *within the system* we measure the internal fit. The internal fit is how accurately we can predict student answers to our online question items, original questions and scaffolds. If we are able to accurately predict a student's response to a given question, this brings us closer to a computer adaptive tutoring application of being able to intelligently select the appropriate next questions for learning and or assessing purposes. Results are shown bellow.

Model	Error	MAD Score	Under/Over Prediction	Error (After Centering)	Centered MAD Score
WPI-106	5.50%	15.25	↓ 12.31	4.74%	12.70
WPI-39	9.56%	26.70	↓ 20.14	8.01%	22.10
WPI-5	17.04%	45.15	↓ 31.60	12.94%	34.64
WPI-1	26.86%	69.92	↓ 42.17	19.57%	51.50

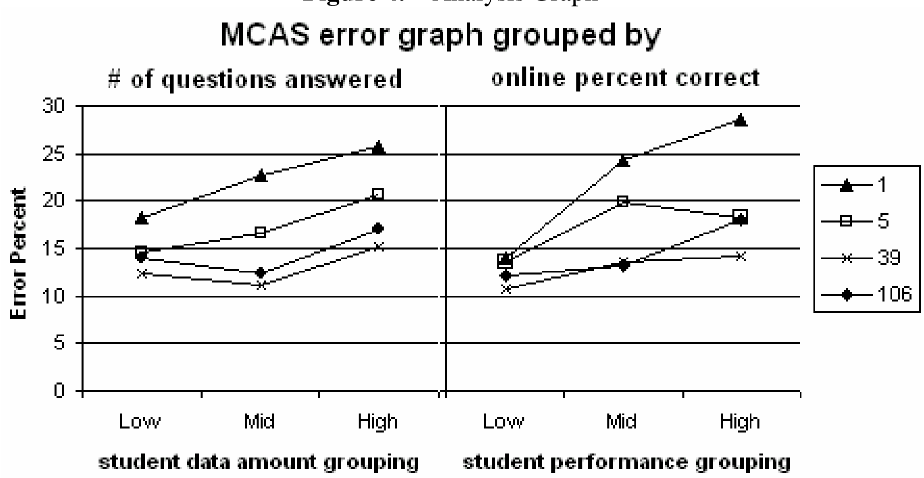
Table 3. Model prediction performance results for internal fit

Like with the MCAS prediction, the internal fit was run on a single student at a time. The calculation of error is the same as for the MCAS test except that the probability of getting the question correct is rounded to 0 or 1. For each question answered by the student, that data point was held out and the rest of the student data was offered to the Bayes net as evidence. The inference was then made on that question giving the probability the student will get the question correct. If the probability of correct was greater than 0.5, 1 point was added to the predicted total point score, otherwise no points were added. The absolute difference between the predicted total point score and actual point score was then divided by the total number of questions answered by the student and that is the error percentage score. This method was employed to maintain symmetry with the methodology from the MCAS test predications in the above section. All the differences between the models in Table 3 were statistically significantly different at the $p < .05$ level.

5 Discussion and Conclusions

The results we present seems to be mixed on first blush. The internal fit of the different models had clear results showing that the finer grained the model, the better the fit to the data collected from the ASSISTment system. This result is in accord with some other work we have done using mixed-effect-modeling rather than Bayes nets [8]. Somewhat surprising, at least to us, is that this same trend did not continue as we expected in the result shown in Table 2. In hindsight, we think we have an explanation. When we try to predict the MCAS test, we are predicting only 29 questions, but they represent a subset of the 109 skills that we are tracking. So the WPI-106, which tries to track all 106 skills, is left at a disadvantage since only ¼ of the skills it is tracking are relevant on the MCAS test. Essentially 75% of the data that the WPI-106 collects is practically thrown out and never used. Whereas the WPI-39, which does the best, can benefit from its fine-grained tracking and almost all of its skills are sampled on the 29 item MCAS test.

Figure 4. – Analysis Graph



In Figure 4 we decided to try to dig into our results so we could better understand how our models perform. Quite surprising to us, we found that the top performing third of students were predicted much worse than the bottom third with all models. Another surprise was that all models predict worse with high amounts of online data versus low amounts. We do not have a firm explanation for this.

As a field we want to be able to build good fitting models that track many skills. Interestingly, item response theory, the dominate methodology used in assessing student performance on most state tests tends to model knowledge as a unidimensional construct, but allowing the items themselves to vary in difficulty (and other properties of items like discrimination and the probability of guessing). Some of our colleagues are pursuing item response models for this very dataset [1, 3] with considerable success, but we think that item response models don't help teachers identify what skills a students should work on, so even though it might be very good predictor of students, it seems to suffer in other ways. We should remind ourselves if you have two models that can predict the data equally well, the finer-grained model is probably the more intepratable and more usefull to use to give reports to teachers.

5.1 Future Work

Our results suggest the 106 skill model as being best for internal fit while 39 skill model is best for the MCAS test, however, a combination of models may be optimal. Building a hierarchy in an aggregate or prerequisite way [4] will likely best represent the various granularities of student understanding and comprehension. These levels of understanding may change over time, so a dynamic Bayes approach will be needed to model these changes as well as model the important variable of learning. This will greatly improve our internal accuracy and will likely show the most benefit to the finer-grained models since the learning of a particular skill will be identifiable. Difficulty is another variable that has the potential to improve model performance. There are many ways to modeling difficulty; the challenge will be to find a method that compliments our current skill models. Additional research into handling the scaffolding selection effect and data filtering will also be explored in future research.

Acknowledgements

This research was made possible by the US Dept of Education, Institute of Education Science, "Effective Mathematics Education Research" program grant #R305K03140, the Office of Naval Research grant #N00014-03-1-0221, NSF CAREER award to Neil Heffernan, and the Spencer Foundation. All of the opinions in this article are those of the authors, and not those of any of the funders. This work would not have been possible without the assistance of the 2004-2005 WPI/CMU ASSISTment Team that helped make possible this dataset, including Ken Koedinger, Brian Junker, Carolyn Rose & Andrea Knight all at CMU, and Mingyu Feng, Abraao Lourenco, Michael Macasek, Goss Nuzzo-Jones, Kai Rasmussen, Leena Razzaq, Steven Ritter at Carnegie Learning, Terrence Turner, Ruta Upalekar, and Jason Walonoski.

REFERENCES

[1] Anozie N., & Junker B. W. (2006). Predicting end-of-year accountability assessment scores from monthly student records in an online tutoring system**.** In Beck, J., Aimeur, E., & Barnes, T. (Eds). Educational Data Mining: Papers from the AAAI Workshop. Menlo Park, CA: AAAI Press. pp. 1-6. Technical Report WS-06-05.

[2] Arroyo, I., Woolf, B. (2005) Inferring learning and attitudes from a Bayesian Network of log file data. Proceedings of the 12thInternational Conference on Artificial Intelligence in Education. 33-40.

[3] Ayers E., & Junker B. W. (2006). Do skills combine additively to predict task difficulty in eighth grade mathematics? In Beck, J., Aimeur, E., & Barnes, T. (Eds). Educational Data Mining: Papers from the AAAI Workshop. Menlo Park, CA: AAAI Press. pp. 14-20. Technical Report WS-06-05.

[4] Carmona1, C., Millán, E., Pérez-de-la-Cruz, J.L., Trella1, M. & Conejo, R. (2005) Introducing Prerequisite Relations in a Multi-layered Bayesian Student Model. In Ardissono, Brna & Mitroivc (Eds) *User Modeling 2005; 10th Internaton Confrence.* Springer. 347-356

[5] Conati, C., Gertner, A., & VanLehn, K. (2002). Using bayesian networks to manage uncertainty in student modeling. *User Modeling and User-Adapted Interaction*, *12*(4), 371–417.

[6] Corbett, A. T., Anderson, J. R. & O'Brien, A. T. (1995) Student modeling in the ACT programming tutor. Chapter 2 in Nichols, P. D., Chipman, S. F. and Brennan, R. L. (eds.) (1995). Cognitively diagnostic assessment. Lawrence Erlbaum Associates: Hillsdale, NJ.

[7] Feng, M., Heffernan, N.T., & Koedinger, K.R. (2006b). Predicting state test scores better with intelligent tutoring systems: developing metrics to measure assistance required. In Ikeda, Ashley & Chan (Eds.). Proceedings of the 8th International Conference on Intelligent Tutoring Systems. Springer-Verlag: Berlin. pp. 31-40. 2006.

[8] Feng, M., Heffernan, N. T., Mani, M., & Heffernan, C. (2006). Using Mixed-Effects Modeling to Compare Different Grain-Sized Skill Models. In Beck, J., Aimeur, E., & Barnes, T. (Eds). Educational Data Mining: Papers from the AAAI Workshop. Menlo Park, CA: AAAI Press. pp. 57-66. Technical Report WS-06-05. ISBN 978-1-57735-287-7.

[9] Mathan, S. & Koedinger, K. R. (2003). Recasting the Feedback Debate: Benefits of Tutoring Error Detection and Correction Skills. In Hoppe, Verdejo & Kay (Eds.), *Artificial Intelligence in Education: Shaping the Future of Learning through Intelligent Technologies., Proceedings of AI-ED 2003* (pp. 39-46). Amsterdam, IOS Press.

[10] McCalla, G. I. and Greer, J. E. (1994). Granularity-- based reasoning and belief revision in student models. In Greer, J. E. and McCalla, G. I., editors, Student Modelling: The Key to Individualized Knowledge--Based Instruction, pages 39--62. Springer--Verlag, Berlin.

[11] Mislevy, R.J., & Gitomer, D. H. (1996). The role of probability-based inference in an intelligent tutoring system. User-Modeling and User Adapted Interaction, 5, 253-282.

[12] Mayo, M., Mitrovic, A. Using a probabilistic student model to control problem difficulty. Proc. ITS'2000, G. Gauthier, C. Frasson and K. VanLehn (eds), Springer, pp. 524-533, 2000.

[13] Pardos, Z. A., Heffernan, N. T., & Anderson, B., Heffernan, C. L.. Using Fine-Grained Skill Models to Fit Student Performance with Bayesian Networks. Workshop in Educational Data Mining held at the Eight International Conference on Intelligent Tutoring Systems. Taiwan. 2006.

[14] Reye, J. (2004). Student modelling based on belief networks. *International Journal of Artificial Intelligence in Education: Vol. 14*, 63-96.

[15] Zapata-Rivera, J-D and Greer, J.E. (2004). Interacting with Inspectable Bayesian Models. *International Journal of Artificial Intelligence in Education*. *Vol. 14*, 127-163.