Using Fine-Grained Skill Models to Fit Student Performance with Bayesian Networks

5 authors, including:

Zachary A Pardos
University of California, Berkeley
97 PUBLICATIONS   **1,749** CITATIONS

Neil T. Heffernan
Worcester Polytechnic Institute
236 PUBLICATIONS   **4,824** CITATIONS

Some of the authors of this publication are also working on these related projects:

Deep Learning in Higher Education Big Data to Explore Latent Student Archetypes and Knowledge Profiles View project

ASSISTments Efficacy Trail with SRI and UMAINE and WPI View project

Using Fine-Grained Skill Models to Fit Student Performance with Bayesian Networks

Zachary A. Pardos & Neil T. Heffernan
Worcester Polytechnic Institute
{zpardos, nth}@wpi.edu

Brigham Anderson
Carnegie Mellon University
brigham@cmu.edu

Cristina L Heffernan
Worcester Public Schools
cris@silverbeach.org

Abstract. The ASSISTment online tutoring system was used by over 600 students during the school year 2004-2005. Each student used the system as part of their math classes 1-2 times a month, doing on average over 100+ state-test items, and getting tutored on the ones they got incorrect. The ASSISTment system has 4 different skill models, each at different grain-size involving 1, 5, 39 or 106 skills. Our goal in the paper is to develop a model that will predict whether a student will get correct a given item. We compared the performance of these models on their ability to predict a student state test score, after the state test was "tagged" with skills for the 4 models. The best fitting model was the 39 skill model, suggesting that using finer-grained skills models is useful to a point. This result is pretty much the same as that which was achieved by Feng, Heffernan, Mani, & Heffernan (in press), who were working simultaneously, but using mized-effect models instead of Bayes networks. We discuss reasons why the finest-grained model might not have been able to predict the data the best. Implications for large scale testing are discussed.

Keywords: Machine Learning, Bayesian Networks, Fine-Grained Skill Models, Inference, Prediction, MATLAB.

INTRODUCTION

Most large standardized tests (like the SAT or GRE) are what psychometricians call "unidimensional" in that they are analyzed as if all the questions are tapping a single underlying knowledge component (i.e., skill). However, cognitive scientists such as Anderson & Lebiere (1998), believe that students are learning individual skills, and might learn one skill but not another. Among the reasons psychometricians analyze large scale tests in a unidimensional manner is that student performance on different skills is usually highly correlated, even if there is no necessary perquisites relationship between these skills. Another reason is that students usually do a small number of items in a given sitting (39 items for the 8th grade Massachusetts Comprehensive Assessment System math test). We are engaged in a effort to investigate if we can do a better job of predicting a large scale test by modeling individual skills. We consider 4 different *skill models*[1], one that is unidimensional, WPI-1, one that has 5 skills we call the WPI-5, one that has 39 skills called the WPI-39 and our most fine-grained model has 106 skills we call the WPI-106. In all cases, a skill model is a matrix that relates questions to the skills needed to solve the problem [Note: we assume that students must know all of the skills associated with a question in order to be able to get the question correct. We do not model more than one way to solve a problem.] The WPI-1, WPI-5, WPI-39, and WPI-106 models are structured with an increasing degree of specificity as the number of skills goes up. The skills of the WPI-5 are far more general than those of the WPI-106. The measure of model performance is the accuracy of the predicted MCAS test score based on the assessed skills of the student.

Modeling student responses data from intelligent tutoring systems has a long history (Corbett, Anderson, & O'Brien, 1995; Draney, Pirolli, & Wilson, 1995) and different skill models have been developed. Our collaborators (Ayers and Junker, 2006, in submission) are also engaged in using a compensatory skill model to predict the state test scores using the same data set that we use in this paper. Though different approaches have been adopted to develop skill models and thus model students' responses, as far as we know little effort has been put to the comparison of different grain-sized skill models in the intelligent tutoring system area. One might think that were would be a great deal of related work in education, but we know of only one

---

[1] A *skill-model* is referred to as a "Q-matrix" by some AI researchers (Barnes, 2005) and psychometricians (Tatsuoka, 1990), while other call them "cognitive models" (Hao, Koedinger & Junker, 2005) and yet other call them "transfer models" (Croteau, Heffernan & Koedinger, 2004).

study where others have tried to do something similar; Yun, Willett and Murnane (2004) showed that they could get a better fit to state test data by using an alternative skill-model that the state government provided.

Bayesian Networks

Bayesian Networks have been used in many intelligent tutoring systems such as Murray, VanLehn & Mostow (2005), Mislevy, Almond, Yan & Steinberg (1999), Zapata-Rivera, D & Greer, (2004) to name just a few instances. One of the nice properties of Bayesian Nets is that they can help us deal with the credit/blame assignment problem. That is, if an item is tagged with two skills, and a student gets the item wrong, which skill should be blamed? Intuitively, if the student had done well on one of the skills previously, we would like to have most of the blame go against the other skill. Bayesian Nets allow for an elegant solution and give us the desired qualitative behavior.

The Massachusetts Comprehensive Assessment System (MCAS)

The MCAS is a Massachusetts state administered standardized test that produces tests for English, math, science and social studies for grades 3 to 10. We are focused on only 8th grade mathematics. Our work related to the MCAS in two ways. First, we have built out content based upon released items as described above. Secondly, evaluating our models by using the 8th grade 2005 test. Predicting students' scores on this test will be our gauge of model performance. We have the results from the 2005 MCAS test for all the students who used our system. The MCAS test consists of 5 open responses (essay), 5 short answers and 29 multiple choice (out of four) questions. The state released items in June, at which time we had our subject matter expert come back to WPI to tag the item before we got individual score reports.

Figure 1. An ASSISTment showing the original question and the first two scaffolding questions.

Background on the ASSISTment Project and Skill Mapping

The ASSISTment system is an e-learning and e-assessing system that is about 1.5 years old. In the 2004-2005 school year, 600+ students used the system about once every two weeks. Eight math teachers from two schools would bring their students to the computer lab, at which time students would be presented with randomly selected MCAS test items. In Massachusetts, the state department of education has released 8 years worth of MCAS test items, totaling around 300 items which we have turned into ASSISTment content by adding "tutoring". If students answered the item correctly they were advanced to the next question. If they answered incorrectly, they were provided with a small "tutoring" session where they were asked to answer a few questions that broke the problem down into steps. A key feature of an ASSISTment is that it provides *instructional assistance* in the process of *assessing students*; the main conference has a paper (Razzaq & Heffernan, in press) on student learning due to the *instructional assistance*, while this paper is focused on assessing *students*.

Each ASSISTment consists of an *original item* and a list of *scaffolding questions*. An ASSISTment that was built for item 19 of the 2003 MCAS is shown in Figure 1. In particular, Figure 1 shows the state of the interface when the student is partly done with the problem. The first scaffolding question appears only if the student gets the item
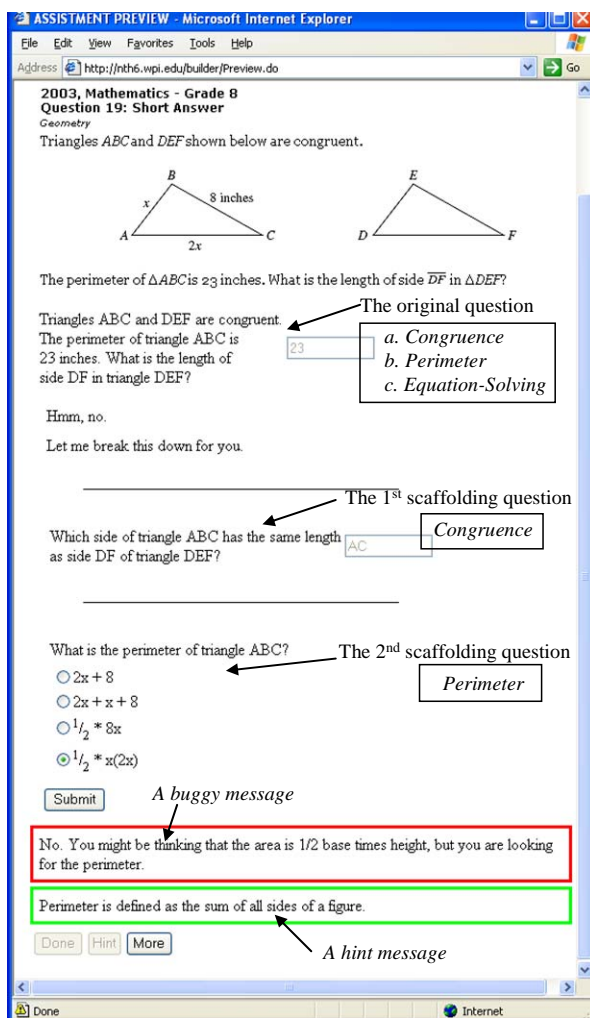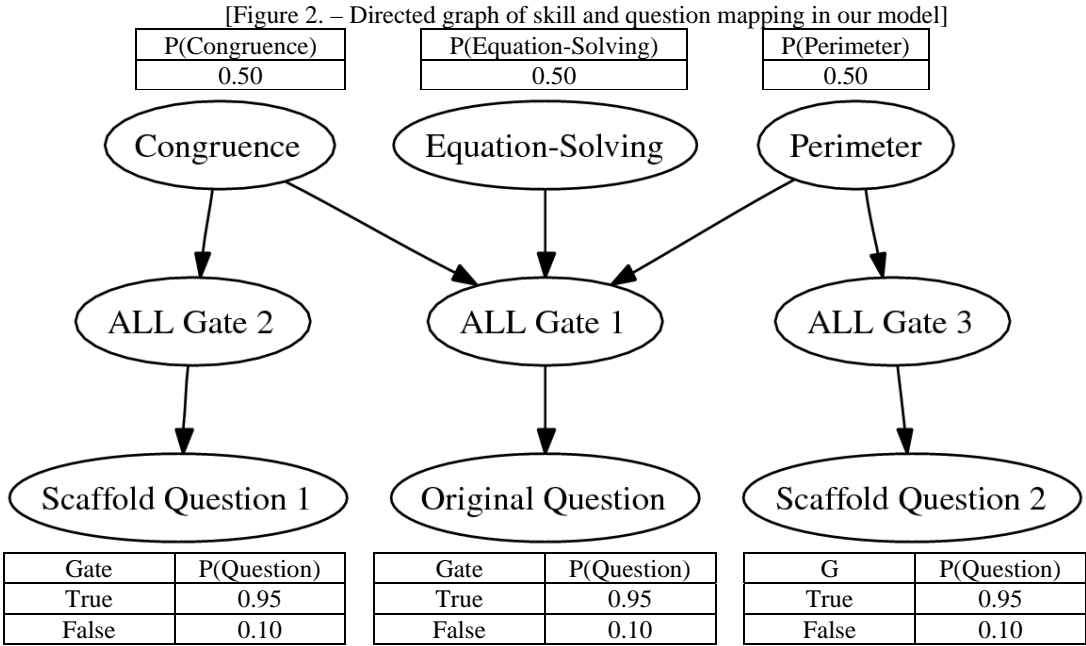
wrong. We see that the student typed "23" (which happened to be the most common wrong answer for this item from the data collected). After an error, students are not allowed to try a different answer to the item but instead must then answer a sequence of scaffolding questions (or "scaffolds") presented one at a time. Students work through the scaffolding questions, possibly with hints, until they eventually get the problem correct. If the student presses the hint button while on the first scaffold, the first hint is displayed, which would be the definition of congruence in this example. If the student hits the hint button again, the second hint appears which describes how to apply congruence to this problem. If the student asks for another hint, the answer is given. Once the student gets the first scaffolding question [tagged with the skill in the WPI-106 of Congruence] correct (by typing *AC*), the second scaffolding question appears. Figure 1 shows a "buggy" message that appeared after the student clicked on "½*x(2x)" suggesting they might be thinking about area. There is also a hint message in a box that gives the definition of perimeter. Once the student gets this question correct they will be asked to solve 2x+x+8=23 for 8, which is a scaffolding question that is focused on equation-solving. So if a student got the original item wrong, what skills should be blamed? This example is meant to show 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 will be able to tell if the student got the item wrong because they did not know congruence versus not knowing perimeter, versus not being able to set up and solve the equation. Most questions' answer fields have been converted to text entry style from the multiple choice style they originally appear as in the MCAS test. As a matter of logging, if the student answered an original question correctly or incorrectly, the student is only marked as getting the item correct if they answered the questions right before asking for any hints or encountering scaffolding.

Figure 3 shows the original question and two scaffold questions from the ASSISTment in Figure 1 as they appear in our online model. The graph describes that Scaffold question 1 is tagged with Congruence, Scaffold question 2 is tagged with Perimeter and the original question is tagged with all three. The ALL gates assert that the student must know all skills relating to a question in order to answer correctly. The ALL gate will be described further in the Bayesian application section. The prior probabilities of the skills are shown at the top and the guess and slip values for the questions are show at the bottom of the graph. These are intuitive values that were used, not computed values. A prior probability of 0.50 on the skills asserts that the skill is just as likely to be known as not known previous to using the ASSISTment system. It is very likely that some skills are harder to learn than others and therefore the actual prior probabilities of the skills should differ. The probability a student will answer a question correctly is 0.95 if they know the skill or skills evolved. Due to various factors of difficulty and motivation, the priors for various questions should be differ. This is why we will attempt to learn the prior probabilities of our skills and question parameters in future papers.

[Figure 2. – Directed graph of skill and question mapping in our model]

Gate	P(Question)
True	0.95
False	0.10

Gate	P(Question)
True	0.95
False	0.10

G	P(Question)
True	0.95
False	0.10

CREATION OF THE 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. There were about 300 released test item for us 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 was free to make up whatever skills she thought appropriate. We printed 3 copies of each item so that each item could show up in different piles where each pile represented a skill. She gave the skills names, but the real essence of a skill is what items it was tagged. The name of the skill served no-purpose in our computerized analysis. When the coding session was over, we had 6, 8 foot-long tables covered with 106 piles of items.[2] To create the coarse-grained models, such as the WPI-5, we used the fine-grained model to guide us. We started off knowing that we would have 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. After our 600 students had taken the 2005 state test, the state released the items from that test, and we had our subject matter expert tag up the items in that test. Shown bellow is a graphical representation of the skill models we used to predict the 2005 state test items. The models are for the MCAS test so you will see the 1, 5, 39 and 106 skills at the top of each graph and the 29 multiple choice questions of the test at the bottom.

[Fig 3.a – WPI-1 MCAS Model]
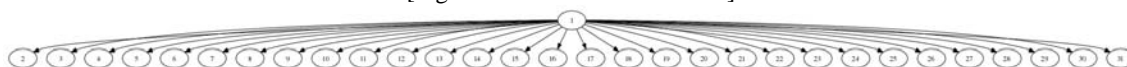
[Fig 3.b – WPI-5 MCAS Model]
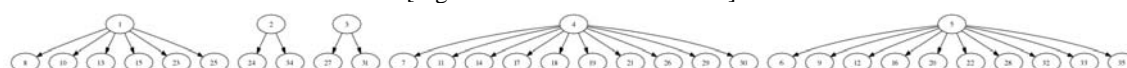
[Fig 3.c – WPI-39 MCAS Model]

[Fig 3.d – WPI-106 MCAS Model]

Figures 3.a and 3.b depict a two layer network where each question node has one skill node mapped to it. Figures 3.c and 3.d introduce multi-mapped nodes, where one question node can have up to three skill nodes mapped to it. The later figures also introduce an intermediary third layer of ALL nodes. You will notice that in the WPI-106 model, many of the skills do not show up on the final test, since each year they decide to test only a subset of all the skills taught in 8th grade math.

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. Figure 4 shows the hierarchal nature of the relationship between WPI-106, WPI39 and WPI-5, and WPI-1. The first column lists just 11 of the 106 skills in the WPI-106. In the second column we see how the first three skills are nested inside of "setting-up-and-solving-equations", which itself is just one piece of "Pattern-Relations-algebra", which itself is one of the 5 that comprise the WPI-1.

---

[2] In Feng, Heffernan, Mani & Heffernan (in press) we called this model the WPI-78 because the dataset that was used included fewer items.

[Fig 4 – 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	

Consider the item in Figure 1, which had the first scaffolding question tagged with "congruence", the second tagged with "perimeter", the third tagged with "equation-solving". In the WPI-39, the item was therefore tagged with "setting-up-and-solving-equations", "understanding-and-applying-congruence-and-similarity" and using-measurement-formulas-and-techniques". The item was tagged with three skills at the WPI-5 level, and just one skill of "math" at the WPI-1.

BAYESIAN NETWORK APPLICATION

Representing the Skill Models

Bayesian networks consist of nodes which have conditional probability tables. These tables indicate the prior probability of an event, given another event. In our three tier model, skill nodes are mapped to ALL nodes, which are mapped to question nodes. Our models allows for a question to be tagged with up to three skills. Any skill that a question has been tagged with is determined to be essential to solving the problem. The assertion here in the model is that in the case of a question mapped with two skills, both those skills must be known in order for the question to be solved by the student. The implementation of this assertion is the ALL gate nodes. The ALL gates also help to simplify the Bayesian network by limiting all question node conditional probability tables to a guess and slip parameter.

Assessing Student Skill Levels

Using MATLAB and the Bayes Net Toolkit as a platform, an architecture was developed to assess the skill levels of students in the ASSISTment system and to test the predictive performance of the various models. First, the skill model, which has been formatted into Bayesian Interchange Format (BIF), is loaded into MATLAB, i.e. bnet39. 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 2,000 question nodes in our system. This can be referred to as the online model. We then load the user's responses to ASSISTment questions from our log file and then enter his/her responses into the Bayesian network as evidence. Using inference, dictated by the CPD tables of the questions, the skill level posterior marginal probabilities are calculated using likelihood-weighting inference which is an approximate inference sampling engine.

Scaffold credit compensation

When evaluating a student's skill level, both top level question and scaffold responses are used as evidence. Scaffolds and top level questions have the same weight in evaluation. If a student answers a top level question incorrectly, it is likely they will also answer the subsequent scaffold questions incorrectly. However, if a student answers a top level question correctly, they are only credited for that one question. In order to avoid this selection effect, scaffolds of top level questions are also marked correct if the student gets the top level question correct. This provides appropriate inflation of correct answers, however, this technique may cause overcompensation when coupled with learning separate parameters for the original and scaffold questions.

Predicting MCAS scores

After the skill levels of a particular student have been assessed using the specified skill model, we then load a model of the actual MCAS test. The MCAS test model looks similar to the training model, with skill

nodes at top mapped to AND 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 evidence in to the test model. Join-tree exact inference is then used to get the marginal probabilities on the questions. That probability is then multiplied by the point value for that question, which is 1 for multiple choice and short answer questions. For example, if the marginal on a question marked with Geometry is 0.6, then 0.6 points are tallied for that question. The same is done for all 29 questions in the test and then the ceiling is taken of the total points giving the final predicted score.

Prior Probabilities

Priors are required for the skill and question nodes in both the training and test models. The priors used for the skills in the training model are set at 0.50 for each skill in the training model. This makes the assumption that it is equally likely a student will know or not know a certain skill when they start using the system. The questions in the training model are given a 0.10 guess and 0.05 slip values. That is, if they do not know the skill(s), there is a 10% probability that they will get the question correct and a 5% probability that they will get it wrong if they do know the skill(s) tagged with the question. For the test model, the questions are given a 0.05 slip and 0.25 guess. The guess value is increased because the MCAS test questions used are multiple choice, out of four.

Software Implementation

The main model evaluation and prediction routine was written in MATLAB by the first author using routines from Kevin Murphy's Bayes Net Toolkit (BNT). Perl scripts were written for data mining and organization of user data as well as the conversion of database skill model tables to Bayesian Interchange Format (BIF) and then to loadable MATLAB/BNT code. Portions of the BIF to BNT conversion was facilitated by Chung Shan's script. MATLAB was setup and results were run on a quad AMD Opteron system running the GNU/Linux platform.

OUTPUT from a single user's run through the evaluation routine is show bellow:
```
[+] Loading bayesian network of transfer model WPI-106 (Cached)
        - Knowledge Components in model: 106
        - Questions in model: 2568
[+] Running User Data Miner to retrieve and organize response data for user 882
        - Number of items answered by student: 225
[+] Crediting scaffold items of correctly answered top level questions
        - Items correct before scaffold credit: 109
        - Items correct after scaffold credit: 195
[+] Loading inference engine (likelihood weighting)
[+] Entering user answers as evidence in bayesian network
[+] Calculating posterior probability values of Knowledge Components
[+] Loading bayesian network of MCAS Test model
        - Knowledge Components in model: 106
        - Questions in model: 30
[+] Loading inference engine (jtree)
[+] Entering posterior probability values as soft evidence in MCAS Test network
[+] Predicting MCAS Test score from posterior values of question nodes
[+] Running User Data Miner to tabulate actual MCAS Test score for user 882
[+] Results:
        Predicted score: 18
        Actual score: 16
        Accuracy: 93%
```

RESULTS

For each student and for each model, we subtract the student's real test score from our predicted score. We took the absolute value of this number and averaged them to get our Mean Absolute Differences (MAD) for each model, shown in Figure 5. For each model we divided the MAD by the number of questions in the test to get a "% Error" for each model.

[Figure 5 – Model Performance Results (30 question test model)]

MODEL	Mean Average Deviance (MAD)	% ERROR
WPI-39	4.500	15.00 %
WPI-106	4.970	16.57 %
WPI-5	5.295	17.65 %
WPI-1	7.700	25.67 %

Does an error rate of 15% on the WPI-39 seem impressive of poor? What is a reasonable goal to shoot for? Zero percent error? In Feng, Heffernan & Koedinger (2006a) we reported on a simulation of giving two MCAS in a row to the same students and then used one test to predict the other and got an approximate 11% error rate, suggesting that a 15% error rate is looking somewhat impressive.

DISCUSSION

It appeared that the WPI-39 had the best results, followed by the WPI-106, followed by the WPI-5, followed by the WPI-1. To see if these "% Error" numbers were statistically significantly different for the different models we compared each model with each other model. We did paired-t-tests between the "% Error" terms for the 600 students. We found that WPI-39 model is statistically significantly better (p<.001) than all the other models, and the WPI-1 is statistically significantly worse than the three other models.   When we compared WPI-106 with WPI-5, we got a p-value of 0.17 suggesting that WPI-106 was not as significantly better (at the p=.05 level) than the WPI-5, but it might be different if we had more data.

After doing this testing we realized that had made a mistake by included one short answer question along with the other 29 multiple choice questions. The inclusion of this item was not a conceptual problem, but we re-ran our analysis to see if we would get similar results, and it would also serve as a sensitivity analysis. Table 6 summarizes the new results

[Figure 6 – Model Performance Results (29 question test model)]

MODEL	Mean Average Deviance (MAD)	% ERROR
WPI-39	4.210	14.52 %
WPI-5	5.030	17.34 %
WPI-106	5.187	17.89 %
WPI-1	7.328	25.27 %

In the 29 question test model the WPI-39 maintains its top standing with a MAD of 4.21 and the WPI-1 remains the lowest performing model with a MAD of 7.328. The subtraction of the short answer test question resulted in slightly better performance of 5.030 for the WPI-5 and slightly lower performance of 5.187 for the WPI-106, compared to the 30 question model. The volatility in relative performance of the WPI-5 vs. WPI-106 reflects the p value of 0.17 calculated above which tells us that these two models' results can not be claimed as statistically different, given these tests.

CONCLUSION

It appears that we have found good evidence that fine-grained models can produce better tracking of student performance as measured by ability to predict student performance on a state test. We hypothesized that the WPI-106 would be the best model, but that was not the case, and instead the WPI-39 was the most accurate. We explain our result by first noting that the WPI-39 is already relatively fine- grained, so we are glad to see that by paying attention to skill model we can do a better job.   On the other hand, the finest grained model was not the best predictor. Given that each student did only a few hundred questions, if we have 106 skills, we are likely to have only a few data points per skill, so we are likely to be seeing a trade-off between finer-grained modeling and a declining accuracy in prediction do to less data per skill.

We think that this work is important, in that using fine-grained models is hard, but we need to be able to show that using them can result in better prediction of things that others care about, such as state test scores. The are still several good reasons for psychometricians to stick with their uni-dimensional models, such as the fact that most tests have a small number of items, and they don't have scaffolding questions that can help deal with the hard credit-blame assignment problems implicit in allowing multi-mapping (allowing a single question to be tagged with more than one skill).

FUTURE WORK

We also want to use these models to help us refine the mapping in the WPI-106. Furthermore, now that we are getting reliable results showing the value of these models, we will consider using these models in selecting the next best-problem to present a student with.   There are many ways we could improve this prediction. Using "time" would be an obvious extension, since we are treating all students' answers, whether collected in September or one week before the real test in May, equally (See Feng, Heffernan, Koedinger, (2006b) for some initial work on using "time".   Learning parameters of all our models and evaluating performance gain could also be productive as will exploring the best hierarchy configuration for prediction.

ACKNOWLEDGEMENTS

REFERNCES

Anderson, J. R. & Lebiere, C. (1998). The Atomic Components of Thought. LEA.

Ayers, E. & Junker, B. (in press). "Do skills combine additively to predict task difficulty in eighth-grade mathematics?" To appear in AAAI-06 Workshop on Educational Data Mining, Boston, 2006.

Barnes,T., (2005), Q-matrix Method: Mining Student Response Data for Knowledge. In the Technical Report (WS-05-02) of the AAAI-05 Workshop on Educational Data Mining, Pittsburgh, 2005.

Corbett, A. T., Anderson, J. R., & O'Brien, A. T. (1995) Student modeling in the ACT programming tutor. Chapter 2 in P. Nichols, S. Chipman, & R. Brennan, Cognitively Diagnostic Assessment. Hillsdale, NJ: Erlbaum.

Draney, K. L., Pirolli, P., & Wilson, M. (1995). A measurement model for a complex cognitive skill. In P. Nichols, S. Chipman, & R. Brennan, Cognitively Diagnostic Assessment. Hillsdale, NJ: Erlbaum.

Embretson, S. E. & Reise, S. P. (2000). Item Response Theory for Psychologists. Lawrence Erlbaum Associates, New Jersey.

Feng, M., Heffernan, N.T, Koedinger, K.R., (in press, 2006a). Predicting State Test Scores Better with Intelligent Tutoring Systems: Developing Metrics to Measure Assistance Required, The 8th International Conference on Intelligent Tutoring System, 2006, Taiwan.

Feng, M., Heffernan, N.T, Koedinger, K.R., (in press, 2006b). Addressing the Testing Challenge with a Web-Based E-Assessment System that Tutors as it Assesses. Accepted to WWW2006, Edinburgh, Scotland.

Feng, M., Heffernan, N. T., Mani, M. & Heffernan, C. L. (in press) Using Mixed-Effects Modeling to Compare Different Grain-Sized Skill Models. To appear in the AAAI 2006 workshop on Educational Datamining. Boston.

Hao C., Koedinger K., and Junker B. (2005). Automating Cognitive Model Improvement by A*Search and Logistic Regression. In the Technical Report (WS-05-02) of the AAAI-05 Workshop on Educational Data Mining, Pittsburgh, 2005.

Mislevy, R.J., Almond, R.G., Yan, D., & Steinberg, L.S. (1999). Bayes nets in educational assessment: Where do the numbers come from? In K.B. Laskey & H.Prade (Eds.), Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence (437-446). San Francisco: Morgan Kaufmann

Murray, R.C., VanLehn, K. & Mostow, J. (2004). Looking ahead to select tutorial actions: A decision-theoretic approach. International Journal of Artificial Intelligence in Education, 14(3-4), 235-278.

Razzaq L., Heffernan, N.T. (2006, in press). Scaffolding vs. Hint in the ASSISTment System. *The 8th International Conference on Intelligent Tutoring Systems*, 2006, Taiwan.

Tatsuoka, K.K. (1990). Toward an integration of item response theory and cognitive error diagnosis. In N. Frederiksen, R. Glaser, A. Lesgold, & M.G. Shafto, (Eds.), Diagnostic monitoring of skill and knowledge acquisition (pp. 453-488). Hillsdale, NJ: Lawrence Erlbaum Associates.

Yun, J T., Willet. J. & Murnane, R. (2004) Accountability-Based Reforms and Instruction: Testing Curricular Alignment for Instruction Using the Massachusetts Comprehensive Assessment System. Paper presentation at the Annual American Educational Research Association Meeting. San Diego, 2004. Achived at http://nth.wpi.edu/AERAEdEval2004.doc

Zapata-Rivera, D & Greer, J. (2004). Interacting with Inspectable Bayesian Student Models. International Journal of Artificial Intelligence in Education, Vol 14. pg., 127 – 168