*Tech., Inst., Cognition and Learning*, Vol. 5, pp. 00-00
Reprints available directly from the publisher
Photocopying permitted by license only



# Developing Fine-Grained Transfer Models in the ASSISTment System

LEENA RAZZAQ, NEIL T. HEFFERNAN, MINGYU FENG AND ZACHARY A. PARDOS*

*Worcester Polytechnic Institute*

In Massachusetts, similar to many states, teachers are being asked to use state mandated assessments in a data-driven manner to help their students meet state standards. However, teachers want feedback on student performance much more often than once a year and they also want better feedback than they currently receive. For instance, while the number of Mathematics skills and concepts that a student needs to acquire is on the order of hundreds, the feedback on the Massachusetts Comprehensive Assessment System (MCAS) test to principals, teachers, parents, and students is broken down into only 5 mathematical reporting categories: Algebra, Number Sense, Geometry, Data Analysis and Measurement. In this article, we describe our attempt to create a fine-grained transfer model for 8th grade math based on the skills needed to take the Math MCAS exam and how we use this model in a web-based intelligent tutoring system called the ASSISTment system.

*Keywords: Transfer model, Intelligent tutoring, Mathematics assessment, Skill mapping*

## INTRODUCTION

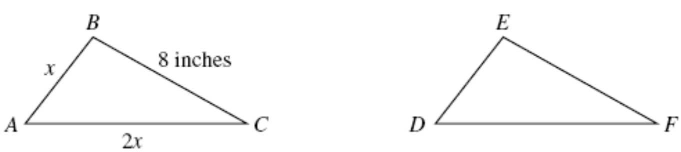
The Worcester Public School District in Massachusetts is representative of the many districts across the country that are trying to use state assessments in a data-driven manner to provide regular and ongoing feedback to teachers and students on progress towards instructional objectives. For instance, the School Improvement Teams at each school review the results from the previous year to analyze which items their students performed particularly poorly on. However,

*Corresponding authors: {leenar|nth|mfeng|zpardos}@wpi.edu

teachers want feedback on student performance much more often than once a year and they do not want to wait six months for the state to grade the exams. Teachers and parents also want better feedback than they currently receive. While the number of Mathematics skills and concepts that a student needs to acquire is on the order of hundreds, the feedback on the Massachusetts Comprehensive Assessment System (MCAS) test to principals, teachers, parents, and students is broken down into only 5 mathematical reporting categories: Algebra, Number Sense, Geometry, Data Analysis and Measurement.

The state of Massachusetts' "Curriculum Framework" breaks the 5 strands into 39 individual "learning standards" for 8th grade math and tags each item with one of the 39 standards. The MCAS reporting system is representative of other states' reporting systems. In 2004, a principal handed us a report, which he received from the state, and asked that we focus efforts on Geometry and Measurement because his students scored poorly in those areas (receiving 38% and 36% correct compared to over 41+% correct in the three other reporting categories). However, a detailed analysis of state tests in Texas concluded that such topic reporting is not reliable because items are not equated for difficulty within these areas (Confrey, Valenzuela, & Ortiz, 2002). To get some intuition on why this is the case, the reader is encouraged to try item 19 from the 2003 MCAS shown in Figure 1. Then ask "What is the most important thing that makes this item difficult?" Clearly, this item includes elements from four of the 5 "strands" (only missing "Data Analysis"). It is Algebra, Geometry (for its use of congruence), Number Sense (for doing the arithmetic operations), or Measurement (for the use of perimeter). Ignoring this obvious overlap, the state chose just one of the 5 strands to classify the item. It turns out, the state classified it as Geometry, but below we show how our methodology is creating evidence to suggest, as you might expect, that there is more to this problem than just Geometry.

**19** Triangles $ABC$ and $DEF$ shown below are congruent.




The perimeter of $\triangle ABC$ is 23 inches. What is the length of side $\overline{DF}$ in $\triangle DEF$?

FIGURE 1
Item 19 from the 2003 MCAS test.

Two years ago, Neil Heffernan and his colleague Ken Koedinger received funding[1] to develop a web-based assessment system, designed to collect formative assessment data on student math skills. Since the assessment is delivered online, students can be tutored on items that they get incorrect. We are currently working with teams of paid and volunteer Worcester Polytechnic Institute (WPI) and Carnegie Mellon University students as well as teacher volunteers to build the ASSISTment system, which is reported on in Razzaq et al (2005).

The ASSISTment system is a web-based assessment system that gives tutoring on questions that students get wrong. Once students log into the system they are presented with math problems. Figure 2 shows a screenshot of an ASSISTment for the 19th item on the 2003 MCAS test for 8th grade math discussed above. The item involved understanding algebra, perimeter, and congruence. If the student had answered correctly, she would have moved on to a new item, but she incorrectly typed 23, to which the system responded, “Hmm, no. Let me break this down for you”, which then engaged the student to give her some questions that would help isolate which of the skills she had an error on, and to give her tutoring so that she could figure out the correct actions. The tutor began by asking a “scaffolding” question that isolated the step involving *congruence*. Eventually she got the scaffolding question correct (i.e., AC), and then was given a question that looked to see if she understood *perimeter*. The figure shows the student selected (1/2) *8*x which the system responded to with a “buggy message” letting the student know she seems to be confusing perimeter with area. The student requested two hint messages, as shown at the bottom of the screen. The tutoring ends with a final question that asks the original question again. Then the student will go on to do another MCAS item, and will again get tutoring if she gets it wrong.

Teachers can send their students to the computer lab for about 20 minutes a week for a cognitively-based formative assessment that will not only identify opportunities for special instruction, but will also provide these students with tutoring as they are assessed. Each week, the website learns more about each student and provides an increasingly accurate prediction of how well each student will do on the MCAS. More importantly, it will indicate what knowledge limitations the teacher could most productively address, of both individual students, as well as the class as a whole. The ASSISTment system’s primary goal

[1]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. Razzaq was funded by the National Science Foundation under Grant No. 0231773. All the opinions in this article are those of the authors, and not those of any of the funders.

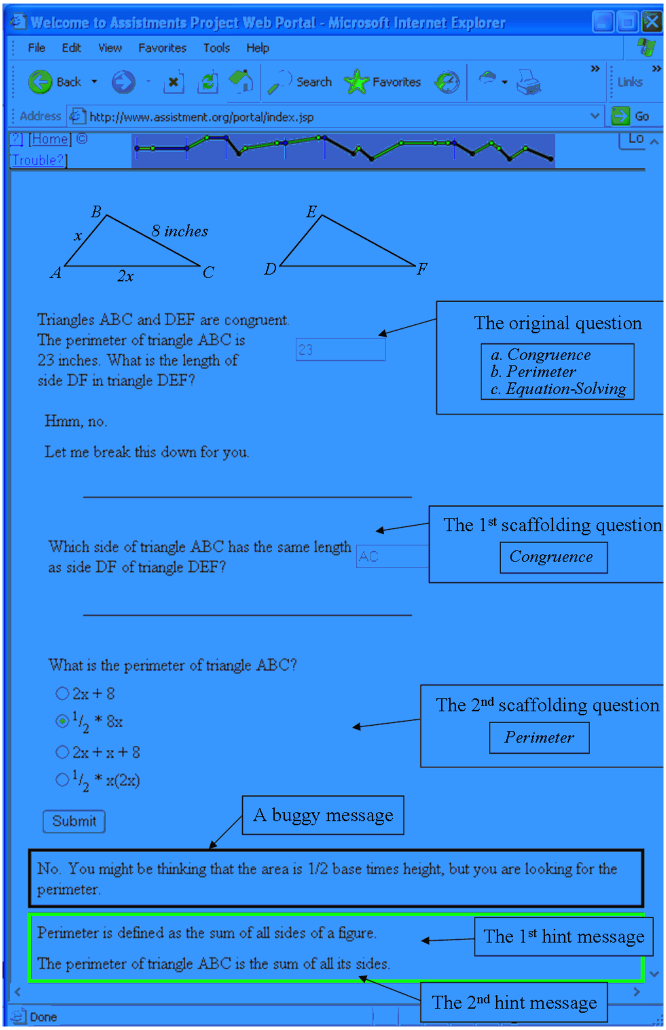



FIGURE 2
An Assistment running.

is to provide feedback to teachers so that they can adjust their classroom strategies based upon the system's fine-grained analysis of student learning.

In addition to testing and teaching at the same time, ASSISTments provide a continuous rather than discrete measure of student performance and understanding by tracking the amount of assistance the student needs to complete a problem. Because the scaffolding questions break the test questions down into a series of simpler tasks that directly assess fewer knowledge components, we believe the ASSISTment system can do a more accurate assessing job. This hierarchal breakdown of knowledge provides a much finer-grained analysis than is currently available procedural.

There were about 300 released test items for us to code. Because we wanted to be able to track learning between items, we wanted to come up with a number of knowledge components that were somewhat fine-grained but not too fine-grained such that each item had a different knowledge component. We therefore imposed upon our subject-matter expert that no one item would be tagged with more than 3 knowledge components. She was free to make up whatever knowledge components 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 knowledge component. She gave the knowledge components names, but the real essence of a knowledge component is what items it was tagged with. The name of the knowledge component served no-purpose in our computerized analysis. When the coding session was over, we had 6, 8 foot-long tables covered with 98 piles of items (see Figures 3 and 4). 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 Sense 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 as well.

We referred to the collection of knowledge components that resulted from our coding session as a “transfer model” (Croteau, Heffernan & Koedinger, 2004). A transfer model is a matrix that relates questions to the skills needed to solve the problem. 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. A transfer model is also referred to as a “Q-matrix” by some AI researchers (Barnes, 2005) and psychometricians (Tatsuoka, 1990), while others call them “cognitive models” (Hao, Koedinger & Junker, 2005).

One might wonder what else developing these transfer models are useful for. If we have a better transfer model we should be able to do a better job of predicting which items a student will get correct in real-time. That means we should be able to do a better job of selecting the next best item for him to work on. In our tutoring system, the next best item will be the one that has the largest ratio of expected test-score gain to expected time to complete the problem. Expected test score gain will be a function that depends upon both the expected

FIGURE 3
Our April 2005 coding session where we tagged all of the existing 8th grade MCAS items with knowledge components.

rise in knowledge components from doing that item at that time, as well as the weight of that knowledge component on the test (i.e., the MCAS). Such a model will allow a teacher who has one week before the 10th grade MCAS to know what topics to review to maximize the class average. We should be able to make a calculation averaging the whole class to suggest what will give the teacher the biggest "bang for the buck".

An example of a useful report that teachers can get using the Assistment system is shown in Figure 5. Teachers can see how their students are doing on each knowledge component and can determine where they need to spend the most time in their classroom instruction.

**How we build ASSISTments**

The main goals of the ASSISTment Builder are ease of use and accessibility during content creation. The web was chosen as the delivery medium to make the tool immediately available to users. The only requirement to use the tool is registration on our website; no software needs to be obtained or installed. Our primary users are middle school and high school teachers in the state of

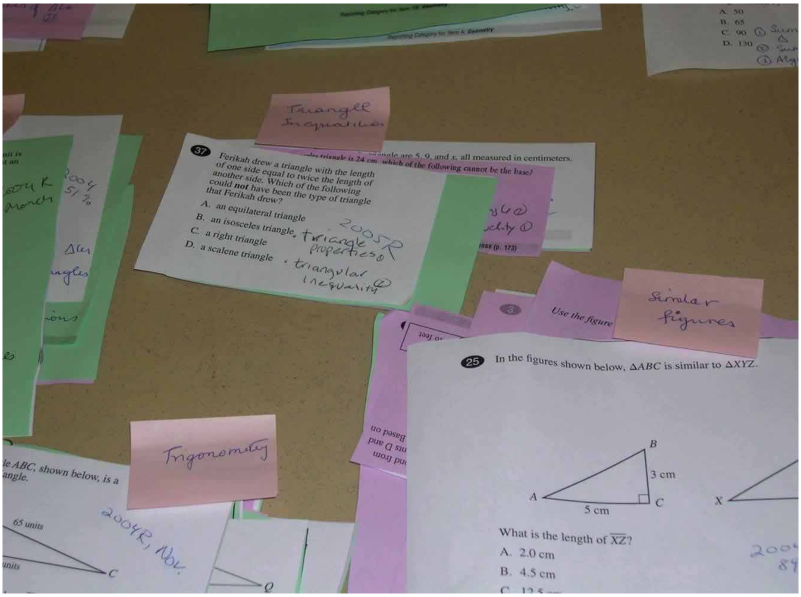



FIGURE 4
MCAS items tagged with knowledge components.

**5 knowledge components your students doing well**

| MCAS-5 | MCAS-39 | WPI-98 | Correct Rate | | |
|---|---|---|---|---|---|
| Number Sense | N.10.8-computing-numbers | Addition | | 86% | 392 |
| | N.1.8-number-representations | Integers | | 85% | 107 |
| | | Ordering-Numbers | | 79% | 312 |
| | | Rounding | | 79% | 164 |
| | N.10.8-computing-numbers | Subtraction | | 76% | 715 |

**5 knowledge components your students need more practice**

| MCAS-5 | MCAS-39 | WPI-98 | Correct Rate | | |
|---|---|---|---|---|---|
| Measurement | M.2.8-concerting-measures | Unit-Conversion | | 44% | 1063 |
| Algebra | A.7.8-solving-equations | Inequality-Solving | | 43% | 593 |
| Geometry | G.1.8-polygon-geometry | Sum-Of-Interior-Angles | | 42% | 124 |
| | G.4.8-pethagorean-theorem | Pythagorean-Theorem | | 37% | 890 |
| Data Analysis | D.2.8-data-representation-tech. | Venn-Diagram | | 37% | 424 |

FIGURE 5
Knowledge component report for one of our participating teachers.

Massachusetts who are teaching the curriculum of the Massachusetts Frameworks; thus, the ASSISTment Builder was designed with an interface simple enough for users with little or no computer science and cognitive psychology background. The ASSISTment Builder also includes other tools to allow teachers themselves to create content and organize it into curriculums and assign to classes, all of which can be done by the teachers themselves. This provides teachers with a total web-based solution for content management and deployment.

**ASSISTments**

An example of a basic ASSISTment is a top-level question that branches into scaffolding problems depending on the student's actions. Scaffolding problems are queued immediately after the behavior consumes an interface action that results in a transition to a state containing scaffolds. One or more scaffolding problems can be mapped to a specified user action. In the ASSISTment Builder an incorrect answer to the top-level problem or a request for hints on the top-level problem will immediately queue a list of scaffolding problems specified by the content creator. Upon answering a scaffolding problem correctly the student is presented with the next one in the queue until it is empty. When an ASSISTment has no more problems in queue it is considered to be finished.

Aside from buggy messages and scaffolds, a problem can also contain hint messages. Hint messages provide insights into methods to solve the given problem. Combining hints, buggy messages, and scaffolds together provides a means to create ASSISTments that are simple but can address complex behavior. Content creators can create complex tree structures of problems each with their own specific buggy messages, hints, and possibly sub-scaffolds.

**ASSISTment Builder Structure**

We constructed the ASSISTment Builder as a web application for accessibility and ease of use purposes. A content creator can build, test, and deploy an ASSISTment without installing any additional software. It is a simple task to design and test an ASSISTment and release it to students. If further changes or editing are needed the ASSISTment can be loaded into the ASSISTment Builder, modified, and saved; all changes will immediately be available in all curriculums that contain the ASSISTment. By making the ASSISTment Builder available over the web, new features are instantly made available to users without any software update. The central storage of ASSISTments on our servers makes a library of content available to teachers which they can easily combine with their own created content and release to their classes organized in curriculums.

The initial view, which has been redesigned based on user input, presents users of the ASSISTment Builder with a top level problem. At the very top of the screen are several links to help manage ASSISTments. The problem is blank and users can enter answers, buggy messages, question text and/or images as well as selecting how the question should be answered (i.e. multiple choice or short answer). A content creator can also add hints. However, hints and scaffolds are mutually exclusive in the top level problem, and a user must select either one for the top level problem. Each section in the problem view is collapsible to allow users to conserve screen space.

The question section is the first section that content creators will usually use. This section allows a user to specify a problem's question text using html and/or images as well as select the interface widget they wish to use and the ordering method used to sort the answers. There are currently three ways to order answers: random, alphabetic, or numeric. This interface is shown in Figure 6.

The answer section of the problem view allows a content creator to add correct answers and expected incorrect answers. Users can map buggy messages to a specific incorrect answer. Users can also edit answers or toggle their correct or incorrect status.

The hint section allows users to enter a series of hints to the applicable problem. Hints can be reordered. This section contains an option to create a bottom out hint for the user that just presents the student with the solution to the problem.

A typical ASSISTment will contain scaffolds and after a user is finished creating the top level problem they will proceed with adding scaffolds. The view

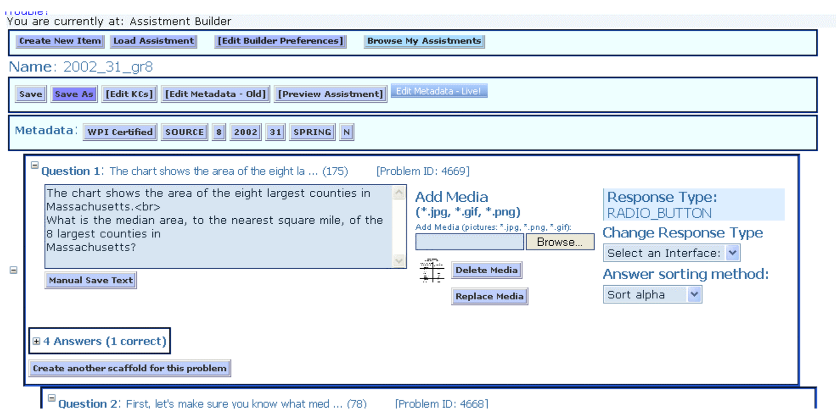



FIGURE 6
Builder interface.

for a scaffolding problem is exactly the same as that for the top level problem, only slightly indented to mark it as a scaffold.

## Knowledge Component Tagging

The ASSISTment Builder supports other applications besides content creation. One of these applications is the mapping of knowledge components, which are organized into sets known as transfer models. This tool, shown in Figure 7, provides a means to map certain skills to specific problems to specify that a problem involves knowledge of that skill. This mapping between skills and problems allows the reporting system to track student knowledge over time using longitudinal data analysis techniques. In a separate paper accepted to WWW2006, we report on the ability to track the learning of individual skills using a coarse-grained model provided by the state of Massachusetts that classifies each 8th grade MCAS math item in one of five categories (i.e. knowledge components in our project): Algebra, Measurement, Geometry, Number Sense, and Data Analysis.

The current system has more than twenty transfer models available, each with up to three hundred knowledge components. In order to more efficiently manage transfer models, the ASSISTment Builder makes use of the preference architecture, allowing users to specify the transfer models they will use. Once those are specified, the user is allowed to browse the knowledge components within each transfer model and to map the ones they select to the problem.

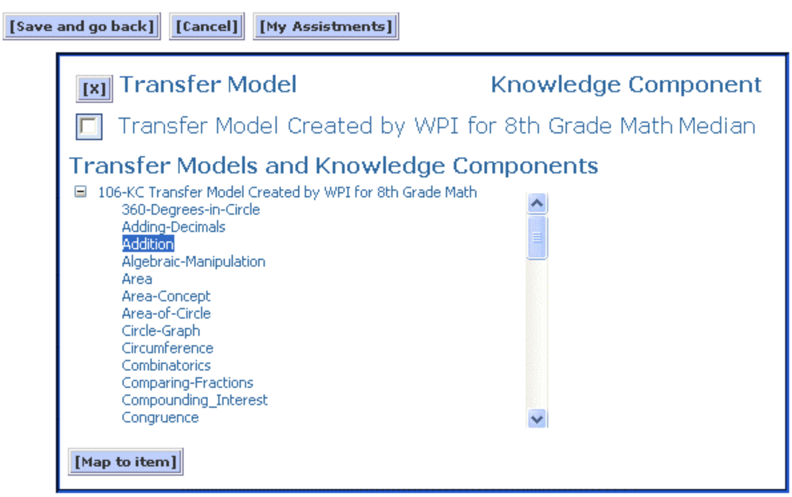



FIGURE 7
Tagging an ASSISTment with skills.

We created ASSISTments for most of the MCAS items that were released since 1998. Every ASSISTment was marked with one or more knowledge component, which is necessary for producing the useful reports for teachers that we described earlier.

**Does our fine-grained model do a good job of assessing?**

How do the finer-grained skill models do on estimating external test scores compared to the skill model with only 5 categories (WPI-5) and the one even with only one category (WPI-1)? We think that if finer-grained models allow for better modeling/prediction, it could have important societal implications (e.g. regarding to tracking student performance.) In Feng et al, 2006 we presented evidence that a finer-grained model performed better.

In the 2004-2005 school year some 600+ students used the system about 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. At the end of that school year, the 8th graders took the 2005 MCAS test. We collected data for students who used our system from Sep. 17, 2004 through May 16, 2005 for on average 7.3 days (one period per day). All these students had worked on the system for at least 6 days. We excluded data from the students' first day of using the system considering the fact they were learning how to use the system at that time. The actual item-level state test report was available for 497 students so that we were able to construct our predictive models on these students' data and evaluate the accuracy on state test score prediction.

The original data set, corresponding to students' raw performance, includes both responses to original questions and to scaffolding questions. It contains about 138 thousand data points, among which around 43 thousand come from original questions. On average, each student answered 87 MCAS (original) questions and 189 scaffolding questions. The data is organized in the way that there can be one or multiple rows for every student response to each single question depending on what skill model we are interested in and how many skills the question is "tagged" with in that particular skill model. For instance, suppose a question is tagged with 2 skills in a model, then for each response made to the question there would be 2 rows in the data set, with skill names listed in a separate column. Students' exact answers are not included. Instead, we use a binary column to represent whether the student answered the specified item correctly or not. No matter what the input type of the item is (multiple-choice or text-input), a "1" indicates a correct response while a "0" means a wrong answer was given.

| Skill Model | MAD | 95% Confidence Interval for MAD | % Error (MAD/34) |
|---|---|---|---|
| WPI-1 | 4.552 | [4.256, 4.848] | 13.39% |
| WPI-5 | 4.343 | [4.066, 4.620] | 12.77% |
| WPI-98 | 4.121 | [3.860, 4.382] | 12.12% |

FIGURE 8
Assessment power of different models.

For every model, we subtracted each student's real test score from his predicted score, took the absolute value of the difference and averaged them to get the Mean Absolute Difference (MAD). We also calculate a normalized metric named % Error by dividing the MAD by 34 to reflect the fact that the MAD is out of a full score 34 points. We compared the three mixed-effects regression models (trained on the "full" data set with scaffolding questions used) fitted using the 3 different skill models. As shown in Figure 8, the WPI-98 model had the best result, followed by the WPI-5, and followed by the WPI-1. % Error dropped down when a finer-grained model was used, from WPI-1 to WPI-5 and then from WPI-5 to WPI-78.

**Does our fine-grained model do a good job of predicting test scores?**

We are engaged in an effort to investigate if we can do a better job of predicting scores on a large scale test by modeling individual skills (Pardos & Heffernan, 2006). Pardos and Heffernan replicated the results discussed above (Feng et al, 2006) by using a different method: Bayesian Networks. In this study, 4 different models were considered, shown in Figure 9: one that is uni-dimensional, WPI-1, one that has 5 skills we call the WPI-5, one that has 39 skills call the WPI-39 and our most fine-grained model has 98 skills we call the WPI-98. The measure of model performance is the accuracy of the predicted MCAS test score based on the assessed skills of the student.

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 using Bayesian Networks. One of the nice properties of Bayesian Nets is that they can help 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 assigned to the other skill. Bayesian Nets allow for an elegant solution and give us the desired qualitative behavior.

| WPI-98 | WPI-39 | WPI-5 | WPI-1 |
|---|---|---|---|
| Inequality-solving<br>Equation-Solving<br>Equation-concept | setting-up-and-solving-equations | Patterns-Relations-Algebra | The skill of "math" |
| Plot Graph | modeling-covariation | | |
| X-Y-Graph<br>Slope | understanding-line-slope-concept | | |
| Congruence<br>Similar Triangles | understanding-and-applying-congruence-and-similarity | Geometry | |
| Perimeter<br>Circumference<br>Area | using-measurement-formulas-and-techniques | Measurement | |

FIGURE 9
Four WPI transfer models for $8^{th}$ grade math.

Figure 10 shows the original question and two scaffold questions from the ASSISTment in Figure 2 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, Scaffold question 3 is tagged with Equation-solving 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 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 before 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. Due to various factors of difficulty and motivation, the priors for various questions should be different and this is why we will attempt to learn the prior probabilities of our skills and question parameters in future research.

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 transfer model, which has been formatted into Bayesian Interchange Format (BIF), 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 transfer model which are appropriately mapped to the over 2,000 question nodes in our system. 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.

When evaluating a student's skill level, both top level questions 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.

After the skill levels of a particular student have been assessed using the

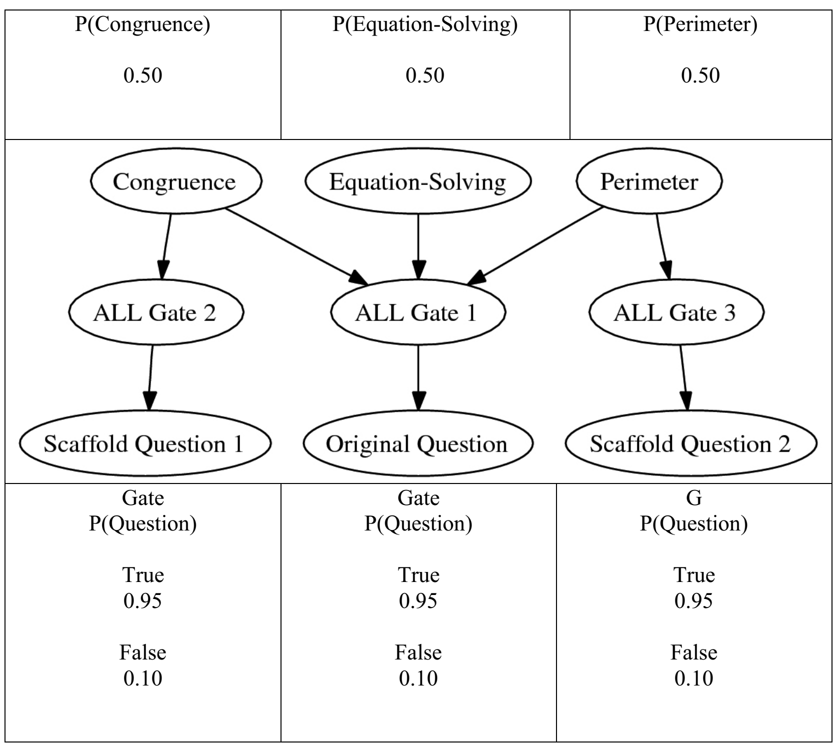



FIGURE 10
Directed graph of skill and question mapping in our model.

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. 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 11. For each model we divided the MAD by the number of questions in the test to get a "% Error" for each model.

| MODEL | Mean Average Deviance (MAD) | % ERROR |
|---|---|---|
| WPI-39 | 4.210 | 14.52 % |
| WPI-5 | 5.030 | 17.34 % |
| WPI-98 | 5.187 | 17.89 % |
| WPI-1 | 7.328 | 25.27 % |

FIGURE 11
Model Performance Results (29 question test model).

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-98 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 98 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 due to less data per skill.

We believe that the ASSISTment system can be a better predictor of state test scores because of this work. Of course, teachers want reports by skills, and this is the first evidence we have that our skill mappings are "good". (We make no claim that our WPI-98 is an optimal fitting mapping.) Now that we are getting reliable models showing the value of these models, we will consider using these models in selecting the next best-problem to present a student with. As part of the future work, we will get our data ready to be shared with other scholars. Researchers interested are welcomed to contact us for details.

## REFERENCES

1. 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.

2. Confrey, J., Valenzuela, A. & Ortiz, A. (2002). Recommendations to the Texas State Board of Education on the Setting of the TAKS Standards: A Call to Responsible Action. At http://www.syrce.org/State_Board.htm

3. 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.

4. 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.

5. Pardos, Z. A., Heffernan, N. T., Anderson, B., & Heffernan C. (2006). Using Fine-Grained Skill Models to Fit Student Performance with Bayesian Networks. Workshop in Educational Data Mining held at the 8th International Conference on Intelligent Tutoring Systems. Taiwan. 2006.

6. Razzaq, L., Feng, M., Nuzzo-Jones, G., Heffernan, N.T., Koedinger, K. R., Junker, B., Ritter, S., Knight, A., Aniszczyk, C., Choksey, S., Livak, T., Mercado, E., Turner, T.E., Upalekar. R, Walonoski, J.A., Macasek. M.A., Rasmussen, K.P. The Assistment Project: Blending Assessment and Assisting. In C.K. Looi, G. McCalla, B. Bredeweg, & J. Breuker (Eds.) *Proceedings of the 12th Artificial Intelligence In Education*, Amsterdam: ISO Press (2005) 555-562.

7. 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.