Improving Intelligent Tutoring Systems: Using Expectation Maximization To Learn Student Skill Levels

Kimberly Ferguson

August 1, 2005

Abstract

This paper uses Expectation Maximization (EM) to learn the hidden characteristic of a student's mastery of mathematical skills. In particular, we build a Bayesian network (BN) based on student pretests of problems using 12 different skills and then run inference to predict a student's individual mastery of each skill. We utilize the Bayesian Information Criterion (BIC) to evaluate different skill models. This learned knowledge of a student's initial skill levels is essential to the overall effectiveness of the Intelligent Tutoring System (ITS).

1 Introduction

Intelligent Tutoring Systems (ITSs) are developed to give individualized instruction to students based on their personal learning style and knowledge level. We want to gather information about a student prior to the actual tutoring session so that the ITS has a starting point of the student's characteristics on which to build. Each of our experiments includes a pretest, two days of interaction with the tutoring system, and a posttest. We give the student a pretest involving problems that utilize specific skills so we can then, from the student's performance on each problem, learn something about the student's initial level of knowledge. Thus, the ITS will have information about which skills a student has already mastered and what level of improvement is still needed for others. The ITS will then be able to infer how successful the student will be when given a problem involving these sets of skills and will utilize these inferences when choosing a pedagogical action during the tutoring session in order to maximize overall learning. We will show that the ITS successfully teaches students by showing an increase in skill mastery and problems answered correctly from pretest to posttest.

We cannot directly observe student skill levels, so we must infer them from the answers to problems involving these skills [ROLE05]. We construct a graphical model which illustrates the dependencies between problems and skills. In particular, we build a Bayesian network (BN) using the Bayes Net Toolbox [M01] which models the connection between skills (hidden nodes) and the problems (observed nodes). We then use Expectation Maximization (EM), a machine learning technique that deals with missing data, to learn the parameters of the network. We can run inference on this learned model during runtime to predict the skill level of a new student and thus better optimize the student's learning within the tutoring session.

The best way to structure a Bayesian network modeling student skills is not obvious. Structural Learning is often used when the configuration of the model is unknown. Structural Learning uses Bayesian Information Criterion (BIC) to compute the Bayesian score of each model and declares the model with the maximum BIC score the best model [S78]. Structural EM is used for unknown models with missing parameters, but it is computationally expensive to search through all possible models [F97]. However, we already have built the different models we wish to evaluate, and thus can use BIC to determine which of our BNs to incorporate into the ITS. We show various versions of both flat and hierarchical models.

This paper is arranged as follows. Section 2 describes the problem in more detail. Section 3 gives the specifics of

the data used. Section 5 presents our approaches to the problem. Section 6 discusses our results. Section 7 gives a conclusion and Section 8 presents ideas for future work.

2 Problem Definition

The Wayang Outpost is an Intelligent Tutoring System developed by Research on Learning and Education (ROLE) at University of Massachusetts Amherst which emphasizes SAT-math preparation [BARW03]. The idea is to individualize the tutoring of each student based upon his or her specific needs. The tutor will pick an action (i.e. a particular problem to give the student next) based on information it gathers about the student such as what skills/problems he or she tends to get correct/incorrect/skip, motivation level, fact retrieval time, and learning style. The information we know about a student will increase as they use the system, but we also need to gather some student characteristics before the tutoring session begins. We collect this information by giving a pretest to the students before they begin the tutoring session which includes non-multiple choice problems similar to the multiple-choice problems within the tutoring session. We want to determine which skills a student initially has mastered and which the or she does not so that the tutor can use this information to discern the best policy for optimizing the student's learning. We then show that the ITS improves student mastery of the skills by comparing these pretest skill mastery results with those of a similar posttest.

Figure 1: Intelligent Tutoring System Architecture

We cannot observe a student's mastery of a skill directly, so we have to infer this knowledge from their answers to problems involving these skills. First we needed too identify which skills were commonly tested on the math portion of the SAT. We categorized 12 of the basic geometry skills needed to do SAT-math problems as follows:

Skill 1: area of a square
Skill 2: area of a right triangle
Skill 3: properties of an isosceles triangle
Skill 4: identify rectangle
Skill 5: area of a rectangle
Skill 6: perimeter of a rectangle
Skill 7: identify right triangle
Skill 8: area of a triangle
Skill 9: Pythagorean theorem
Skill 10: corresponding angles
Skill 11: supplementary angles
Skill 12: sum of interior angles of a triangle

Then we took those 12 skills and created problems utilizing them in the following fashion: there are 12 one-skill problems each of whichuse only 1 of each of the 12 skills; there are 12 two-skill problems which combine 2 of the 12 skills; there are 4 three-skill problems which use 3 of the 12 skills. Each set of three skills $\{1,2,3\}$, $\{4,5,6\}$, $\{7,8,9\}$, and $\{10,11,12\}$ (see Figure 4) is grouped to make the one-skill, two-skill and three-skill problems as described above. This totals 28 problems for the student pretest, 7 from each set of 3 skills. Some skills are obviously simpler than others (i.e. identify rectangle versus Pythagorean theorem), so we expect to find results that these skills are initially mastered more than others. Similarly, the one-skill problems should generally be easier for students than the two-skill and three-skill problems.

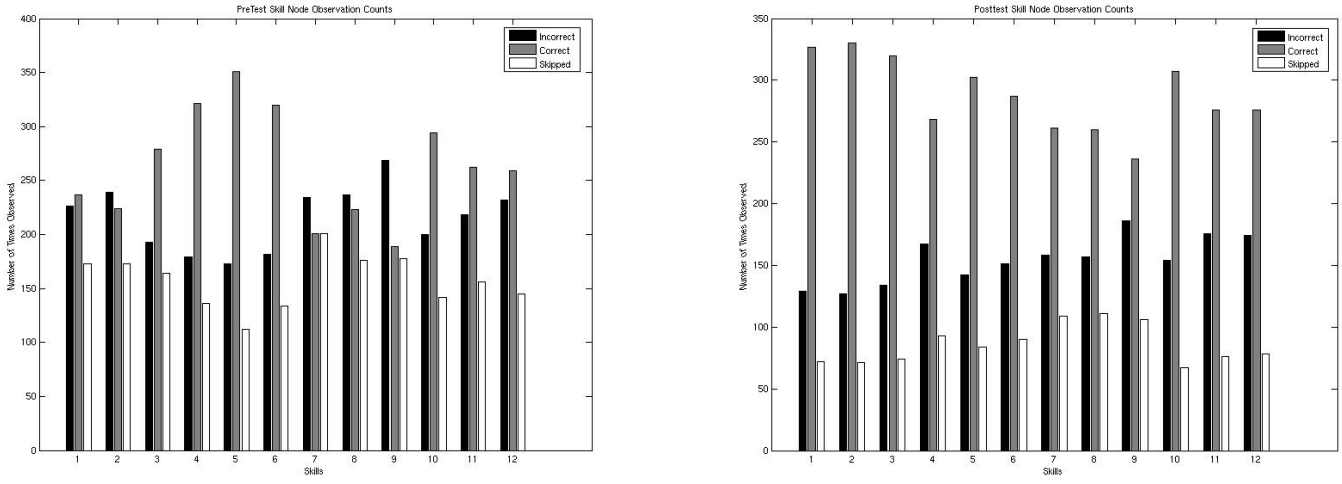
We know exactly which problems involve which skills, and we know how each student answers each problem: incorrect, correct, or skipped. This data is observed. However, the actual mastery of a skill, which is what we really want to know, is a hidden variable. We can treat these hidden skill variables as missing data and use Estimation Maximization (EM) to estimate the parameters of the Bayesian network (BN) in order to learn skill mastery.

Concurrently, we are working on improving the problem and hint selector of the Intelligent Tutoring System. The tutor will choose the best pedagogical approach to increase learning which will aid in the decisions made by the problem and hint selector. Estimating the current skill mastery of each student is essential to the problem selector regardless of which learning philosophy we follow. For example, if the best route is to start on the areas that the student has the most trouble with, then we can use the student's skill levels to determine what areas need the most improvement. Similarly, perhaps a good strategy is to switch between problems that the student has trouble with and problems in which they are already proficient. Again, if we know which skills the student has mastered, the system can select the proper problems with which to challenge or encourage him or her.

3 Data

In Spring 2005 we tracked students of varying ability in Central Massachusetts as they took the pretest, then spent two days using the tutor (50 minutes the first day and 30 minutes the next day) and then took the posttest. We have a total of 159 student pretests from which to learn the pretest BN and 132 student posttest from which to learn the posttest BN. The posttest involves different problems than the pretest, but each problem is associated with the same set of skills as in the pretest. Each problem can be one of 4 observations: incorrect, correct, skipped or left blank. *Left blank* is different than *skipped* since a problem is *left blank* when the student runs out of time before reaching the problem, but a problem tends to be *skipped* because the student doesn't know how to solve the problem and wants to skip to a different problem. Thus, problems that are *skipped* supply us with information about the associated skill(s) being possibly unmastered, as opposed to problems that are *left blank* which we can discount as uninformative.

We can gather some information based on the raw counts of how many student observed answers for problems involving each skill were incorrect, correct, or skipped (See Figure 2).



(a) Pretest Observation Count Across Skills

(b) Posttest Observation Count Across Skills

Figure 2: Improvement in student skill mastery from pretest to posttest

Here we see proof that the Intelligent Tutoring System is teaching students how to do better on these type of

problems. We see that more problems are being answered correctly, and less are being skipped from pretest to posttest. Note the non-uniform distribution of skill levels. This suggests that there is value in modeling and learning the distribution across all students as well as individual students.

4 Models

We have information on which pretest problems are associated with which skills. We then constructed a Bayesian network (BN) linking each of the 12 skills to each of the problems that are associated with it. Each skill is used in four problems: 1 one-skill problem, 2 two-skill problem, and 1 three-skill problem. The BN has hidden nodes representing the mastery of each skill which are modeled by a binomial distribution and observed nodes representing student answers which are modeled by a multinomial distibution. We do not know exactly how the skills should be linked within the network. We have experimented with different structured BN models and use the Bayesian Information Criterion to evaluate them.

We use EM to learn the parameters of each Bayesian network on the pretest data we collected for all students. Given a new student, we can then do inference on this network online to estimate a student's mastery of the skills. This learned network will be included the next experimental study, where each student will take the pretest on which the inference will be done online and immediately used as a factor in the policy selection of the Intelligent Tutoring System.

- *Expectation Maximization:* EM is a framework for maximum-likelihood parameter estimation with missing data [DLR77]. EM finds the model parameters that maximize the *expected* value of the log-likelihood, where the data for the missing parameters are "filled in" by using their expected value given the observed data. In general, EM is trying to learn the pattern that best associates the observed data and the hidden parameters within the context of the specified graphical model. The log-likelihood value maximized though EM is used to calculate the BIC score of that model.

- *Bayesian Information Criterion:* The BIC, also known as Schwarz's Bayesian Criterion [S77], is used to evaluate different models and has been proven to be consistent [HQ79]. Simply put, given a set of data and probability distribution generating the likelihood, the larger the likelihood, the better the model fits. The BIC score is calculated with the following formula: $-2 * ll + npar * log(nobs)$, where ll is the log-liklihood, $npar$ represents the number of parameters and $nobs$ the number of observations in the model. Thus, we will assume that the model with the highest BIC score has the best structure for this task.

- *Inference:* Inference can be thought of as querying the model. We use the junction tree inference algorithm, which is an exact inference engine that uses dynamic programming to avoid redundant computation. We want to ask given a set of observations, what is the probability of other events. For example, when a student answers Problem 1 (using only Skill 1) and Problem 2 (using only Skill 2) correct, how likely is it that he or she will answer Problem 4 (using Skills 1 and 2) correctly as well. We can also use inference to predict a student's overall skill mastery. When we have a pretest for a new student we run inference on the learned model given the new student answers to estimate this student's skill levels. The skill mastery prediction will eventually be done during runtime and used to enhance the ITS's ability to give individualized help and achieve optimal learning for each student.

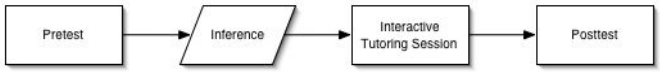


Figure 3: Intelligent Tutoring System Architecture With Inference

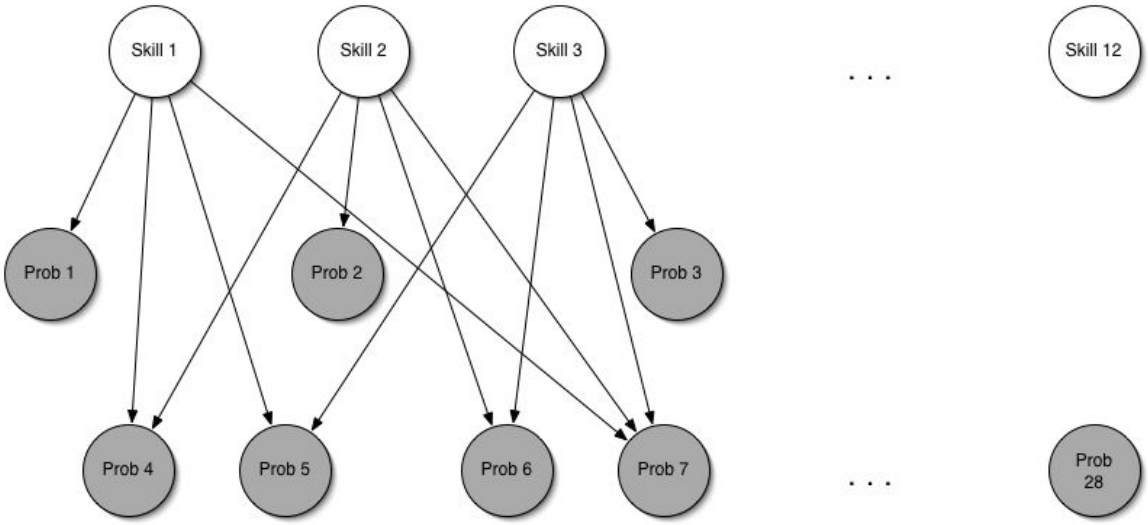


Figure 4: Bayesian network: This identical linking pattern is repeated for each group of 3 skills and 7 problems.

4.1 Flat Skill Models

4.1.1 Flat Skill Model With Uniform Priors

12 Skills (hidden nodes initialized with uniform priors), 28 Problems (observed nodes initialized randomly)
Skills have 2 values: Not Mastered, Mastered
Problems have 3 values: Incorrect, Correct, Skipped

Based on the parameters learned when using uniform priors, we have discovered that it is not the *incorrect* answers from which we infer a lack of mastery, but the *skipped* answers instead (See Results Section Figure 10).

4.1.2 Flat Skill Model With Three Mastery Levels

It is not intuitive to think of skills as being binary values: either mastered or not mastered. There is a lot of middle ground between a skill not being mastered at all and being entirely mastered. This model allows that middle ground for skills that are partially mastered. The graphical model is identical to that of the skill model with uniform priors (Figure 4), only now the Skill nodes have 3 values: Not Mastered, Partially Mastered, and Mastered.

4.1.3 Flat Skill Model With Informed Priors

To increase the accuracy of EM finding the best model to fit the data we used informed prior probabilities on the hidden skill nodes. Some skills are easier for students than others, and the prior probability that a student will have an easier skill mastered before beginning the tutoring session is higher than that of having a harder skill mastered. The graphical model is identical to that of the skill model with uniform priors (Figure 4), only now the 12 Skill nodes are initialized with informed priors. The informed prior probabilities are based on the counts of how many student observed answers for problems involving each skill were incorrect, correct, or skipped (See Figure 2). Thus we have initialized the prior probabilities for each skill using the following equation:

$$P(skill = mastered) = \frac{corr + \frac{1}{2}inc}{total}$$
$$P(skill = unmastered) = 1 - P(skill = mastered) = \frac{skip + \frac{1}{2}inc}{total}$$

where *corr* is the count of how many times a problem which uses this skill was observed as correct across all students, *inc* is the count of how many times a problem which uses this skill was observed as incorrect across all students,

skip is the count of how many times a problem which uses this skill was observed as skipped across all students, and *total* is the total count of all the observed answers for this skill across all students.

We will see that the results improve from those of the model with uniform priors, when setting the priors with this informed estimate approach instead. (See Results and Discussion Section 5.1.4)

4.2 Hierarchical Skill Models

4.2.1 Hierarchical Problem Group Skill Model

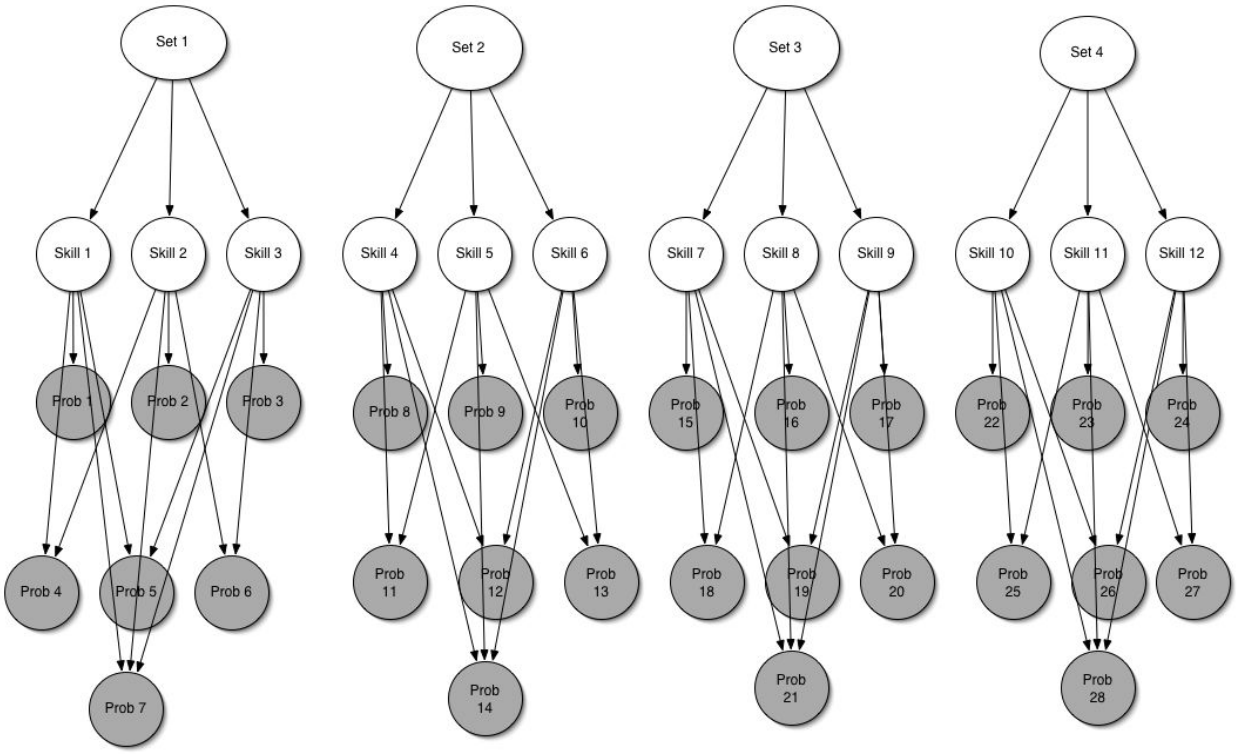


Figure 5: Bayesian network: Hierarchy based on grouping of problems using the same skills

4 Sets (hidden nodes initialized with uniform priors),
12 Skills (hidden nodes initialized with uniform priors), 28 Problems (observed nodes initialized randomly)
Skills have 2 values: Not Mastered, Mastered
Problems have 3 values: Incorrect, Correct, Skipped

Figure 5 captures the idea that the skills are already grouped by sets of problems. The general mastery level of a set of problems may give clues to the mastery level of individual skills within that set.

4.2.2 Hierarchical Skill Group Model

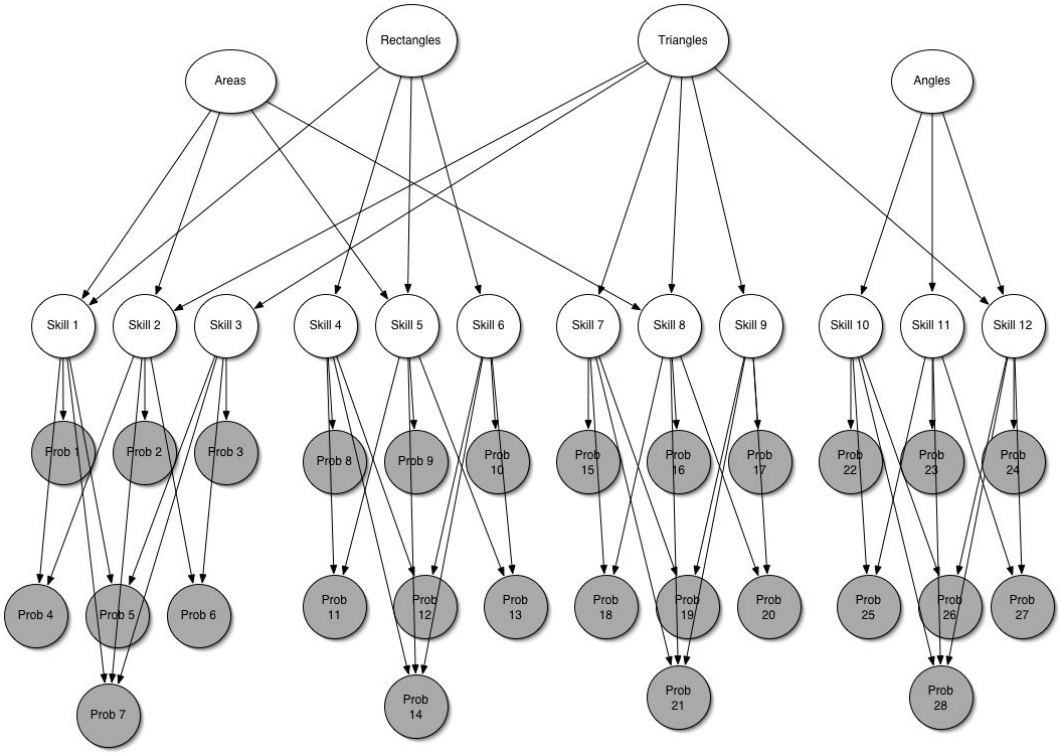


Figure 6: Bayesian network: Hierarchy based on the following categories of skills: Areas, Rectangles, Triangles, Angles

4 Groups (hidden nodes initialized with uniform priors),
12 Skills (hidden nodes initialized with uniform priors), 28 Problems (observed nodes initialized randomly)
Skills have 2 values: Not Mastered, Mastered
Problems have 3 values: Incorrect, Correct, Skipped

Figure 6 captures the idea that the skills have higher level categories in which they can be grouped. For example, it is possible that a student may have skills associated with rectangles mastered, but have trouble with triangles.

4.2.3 Hierarchical Skill Difficulty Model

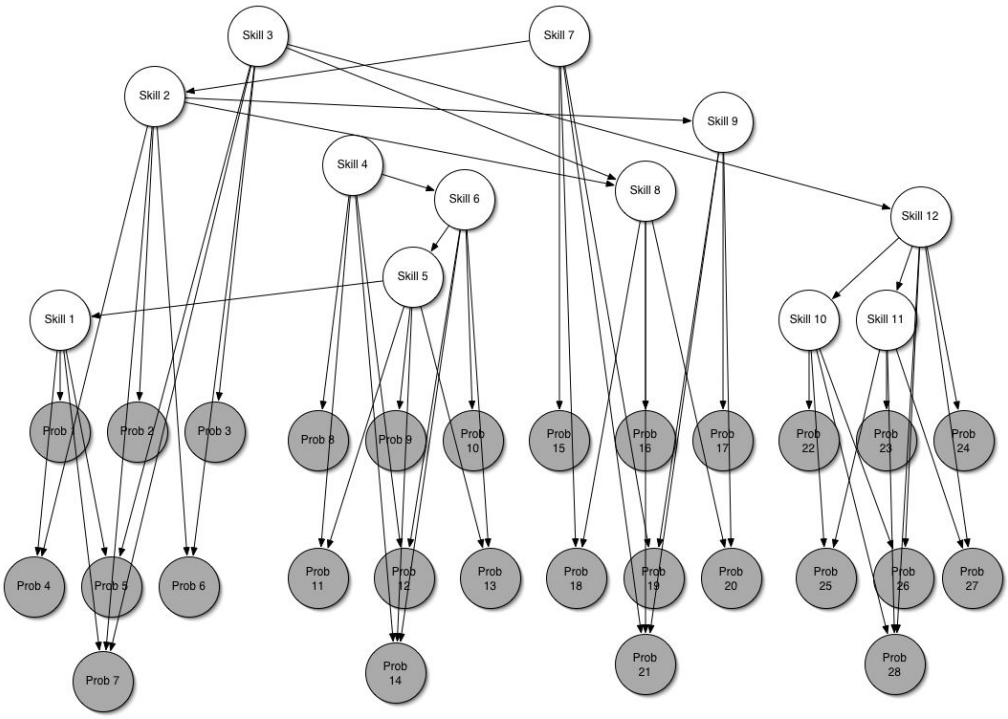


Figure 7: Bayesian network: Hierarchy is based on the order in which skills should be learned (i.e. a student should know how to identify a right triangle (Skill 7) before learning how to take its area (Skill 8) which should all be mastered before learning the Pythagorean Theorem (Skill 9).

12 Skills (hidden nodes initialized with uniform priors, 28 Problems (observed nodes initialized randomly)
Skills have 2 values: Not Mastered, Mastered
Problems have 3 values: Incorrect, Correct, Skipped

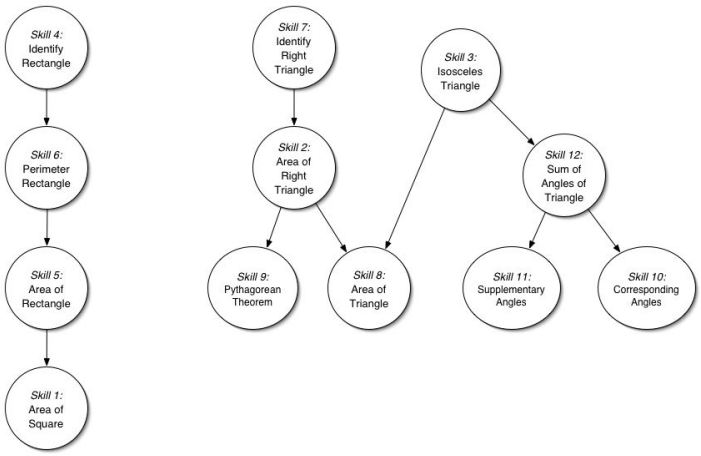


Figure 8: Higher Level Look at Skill Difficulty Hierarchy Model (hidden skill nodes shown only)

Psychology and education researchers studied the pretest and posttest problems and identified these skills. However, these skills are not necessarily mutually exclusive. Figure 7 captures the idea that the skills have different difficulty levels. For example, if the skill of identifying a triangle is not mastered, it seems probable that the skill of finding the area of a triangle is not mastered either. In particular, this Hierarchical Skill Difficulty Model is arranged so that each parent node is a skill which should be mastered before its child node can be mastered.

5 Results and Discussion

5.1 Flat Skill Models

Note that maximum BICs and average BICs are based on 50 random runs.

5.1.1 Flat Skill Model With Simulated Priors

To check the accuracy of the model, we created an experiment using hand-coded priors to get baseline conditional probability Tables (CPTs). The graphical model is identical to that of the skill model with uniform priors (Figure 4) only with user-specified parameters for all 40 nodes. We then sampled from this network to create simulated training data and built an identical model with the skill nodes hidden and initialized randomly. Finally, we found the maximum likelihood estimates of the parameters using the generated data on the model with random parameters. We compared the learned parameters to the parameters set in the initial model using the Kullback-Leibler Divergence (see Figure 9). The KL-Divergence is a distance metric used to measure the difference between two probability distributions. We see that the learned parameters are fairly close to the "true" ones. The divergence decreases as the number of samples is increased, but begins to converge at a reasonably low sample size (400 students).

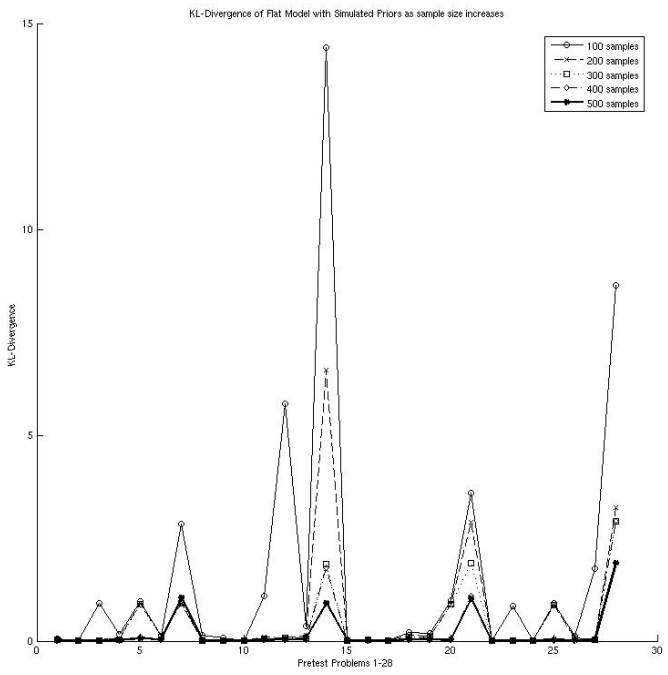


Figure 9: Kullback-Leibler Divergence across 28 problems for increasing sample size

5.1.2 Flat Skill Model With Uniform Priors

- ***BIC Results:***

Max BIC	Average BIC	Standard Deviation	Variance
-5202.4	-5294.7	47.022	2216.7

Table 1: Bayesian Information Criterion For Flat Skill Model With Uniform Priors

This initial experiment is the standard to which we will compare all other models.

- ***Learned Model:***

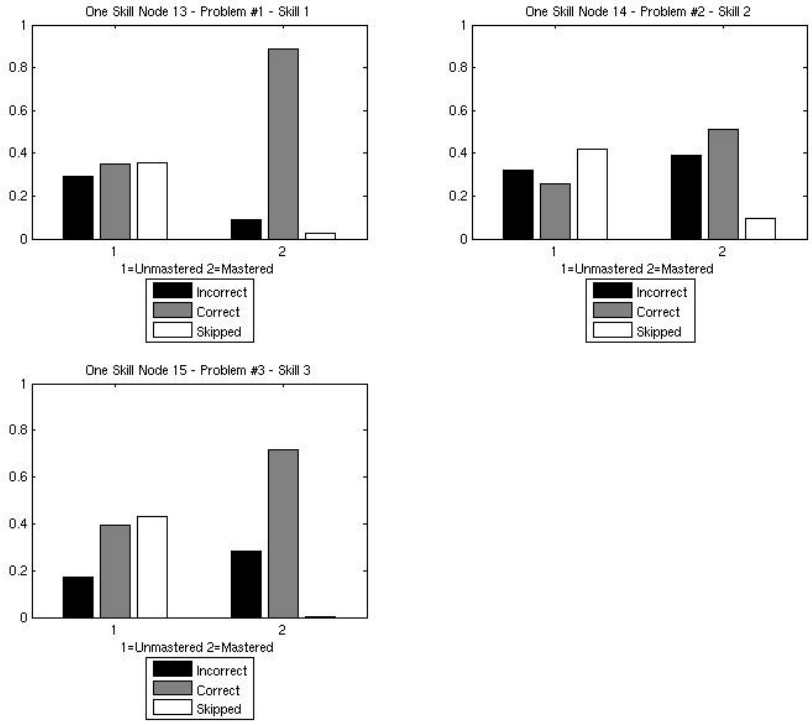


Figure 10: One Skill Problems Learned Conditional Probabilities For Flat Skill Model With Uniform Priors: Skills 1, 2 & 3

In Figure 10, we consider one-skill problems as a simple case. Notice that it is not the *incorrect* answers from which we infer a lack of mastery, but *skipped* instead. This makes sense since the pretest problems are not multiple choice, so if a student has no mastery of the skill needed, he or she will usually skip it. When the observation is *correct* we can usually infer the skill is mastered and when the observation is *skipped* we can usually infer the skill is unmastered, but *incorrect* can mean either. Perhaps, the student does not have to skill fully mastered and thus answers incorrectly, but he or she may make a simple math error or misunderstand the wording of the problem and answer incorrectly even though the skill is mastered.

5.1.3 Flat Skill Model With Three Mastery Levels

- ***BIC Results:***

Max BIC	Average BIC	Standard Deviation	Variance
-5793.2	-5926.4	68.7691	4729.2

Table 2: Bayesian Information Criterion For Flat Skill Model With Three Mastery Levels

The flat skill model with three mastery levels has the lowest maximum and average BIC scores of all the models. Therefore, this model does not fit the data as well as we had assumed.

- ***Learned Model:***

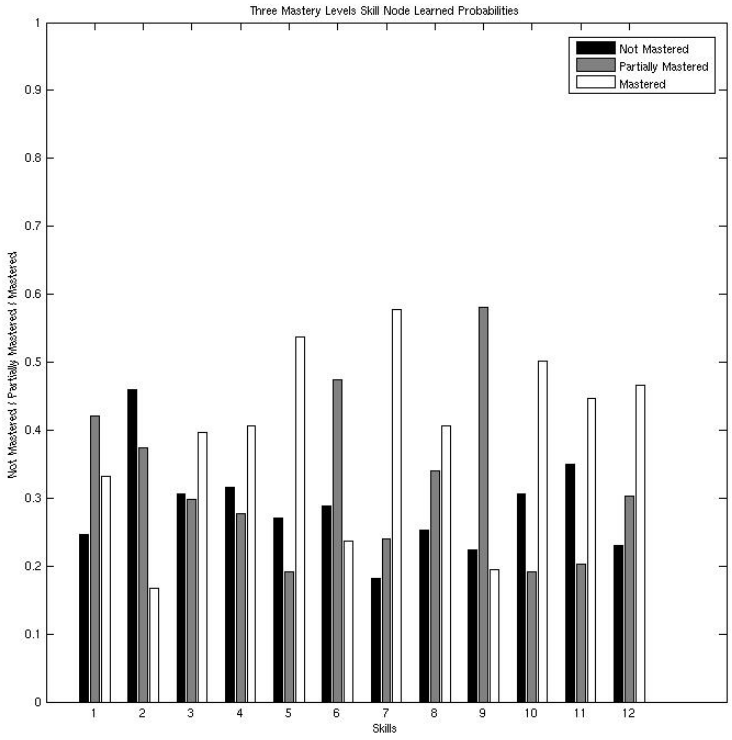


Figure 11: Learned Probabilities For The 12 Skill Nodes For Flat Skill Model With Three Mastery Levels

From the results in Figure 11 we see the prior probabilities of skill mastery based on no observations and can separate skills into the following three groups: 1) skills that are most likely to be not mastered, 2) skills that are most likely to be partially mastered, and 3) skills that are most likely to be mastered.

1) Not Mastered: Skill 2 (area of a right triangle)
2) Partially Mastered: Skill 1 (area of a square), Skill 6 (perimeter of a rectangle), Skill 9 (Pythagorean theorem)
3) Mastered: Skill 3 (properties of an isosceles triangle), Skill 4 (identify rectangle), Skill 5 (area of a rectangle), Skill 7 (identify right triangle), Skill 8 (area of a triangle), Skill 10 (corresponding angels), Skill 11 (supplementary ange Skill 12 (sum of angels in a triangle)

This essentially matches with our notions of easy and hard skills based on the pretest counts in Figure 2(a). However, the conditional probability tables for problems with three mastery levels involving two or three skills are too complicated to analyze by hand.

5.1.4 Flat Skill Model With Informed Priors

- ***BIC Results:***

Max BIC	Average BIC	Standard Deviation	Variance
-5191.1	-5300.4	53.5191	2864.3

Table 3: Bayesian Information Criterion For Flat Skill Model With Informed Priors

As expected, the maximum, as well as the average, BIC score of the flat skill model with informed priors (Table 1) are higher than that of the flat skill model with uniform priors. Thus, informed priors are better for the flat skill model.

- ***Learned Model:***

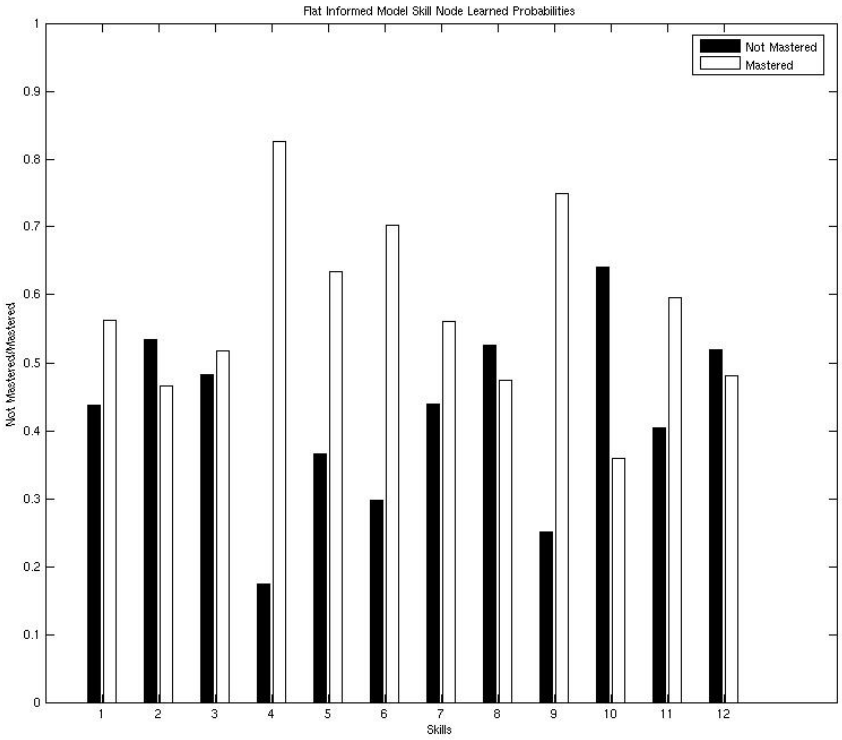


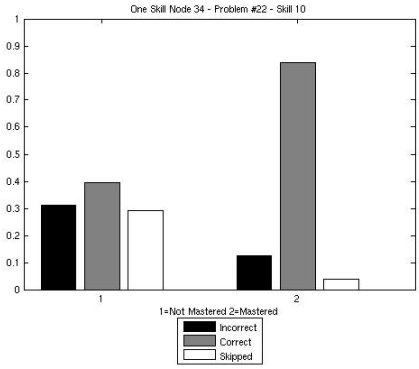
Figure 12: Learned Probabilities For The 12 Skill Nodes For Flat Skill Model With Informed Priors

From the learned model (Figure 12) for the Flat Skill Model With Informed Priors we can see that across all students, Skills 2, 8, 10, and 12 are more likely to be unmastered. We are surprised to see Skills 2 and 8 (area of a right triangle and area of a triangle, respectively) are likely to be unmastered since they are not among the more

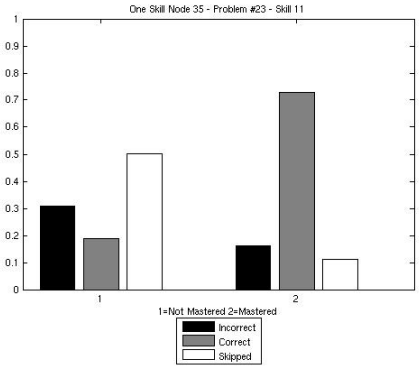
difficult skills. However, it does make sense that these two skills have almost identical learned probabilities, since there is such an overlap in knowledge between them.

The largest peak in mastery level is Skill 4 (identify rectangle) at 82.54% is expected, since this skill should be the easiest of all the skills. Using this reasoning, seeing such a large peak in the mastery of Skill 9 (Pythagorean Theorem), one of the most difficult skills, would normally be an unexpected result. However, from the learned probabilities for Three Mastery Levels Skill Model (see Figure 11), we discovered that a large portion (58.12%) of Skill 9 was actually *partially mastered*. It is understandable, that when we returned to the Two Mastery Level models, most of this was grouped with *mastered*.

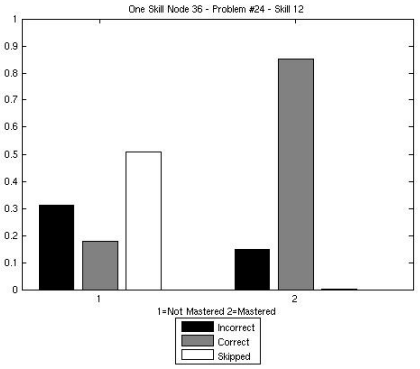
Let us now consider the conditional probabilities learned for problems that only involve one skill: in this case, Skills 10, 11 and 12 (corresponding angles, supplementary angels, and sum of angels in a triangle, respectively).



(a) One Skill Problem Learned Conditional Probabilities: Skill 10



(b) One Skill Problem Learned Conditional Probabilities: Skill 11



(c) One Skill Problem Learned Conditional Probabilities: Skill 12

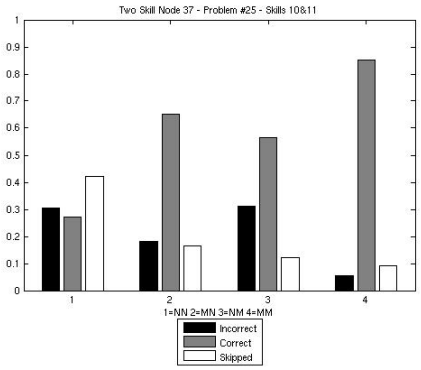
Figure 13: Learned Conditional Probabilities For Flat Skill Model With Informed Priors: One Skill Problems Involving Skills 10, 11, and 12

These graphs align with our preconceptions as follows. Figure 13(a) shows that when we observe a correct answer on Problem 22 there is a 83.68% probability that Skill 10 is mastered. Figure 13(b) shows that when we observe a correct answer on Problem 23 there is a 72.75% probability that Skill 11 is mastered. Figure 13(c) shows that when we observe a correct answer on Problem 24 there is a 85.29% probability that Skill 12 is mastered.

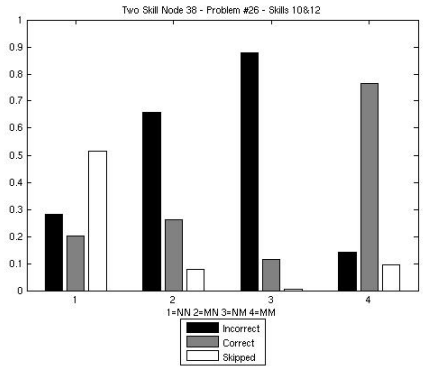
Analysis becomes less straight forward as we look at the conditional probabilities learned for problems that involve two skills.

Figure 14(a) shows that when we observe a correct answer on Problem 25 which involves Skills 10 and 11, that there is a 85.16% probability that both skills are mastered. Additionally, a correct answer shows a 65.23% probability that only Skill 10 is mastered and a 56.54% probability that only Skill 11 is mastered. If we observe a skipped answer we see that both skills are not mastered with probability 42.19%. If the answer is incorrect there is a 30.65% probability that both skills are not mastered, and a 31.24% probability that only Skill 11 is mastered. In summary, this problem is relatively easy for students since is likely to be answered correctly if either or both skills are mastered. However, if Skill 10 is not mastered and Skill 11 is mastered then it is more likely that a student will answer the problem incorrectly than if Skill 10 is mastered and Skill 11 is not mastered. In addition, while it is most likely that the problem will be skipped if both skills are not mastered, it seems that both skills being not mastered was difficult to model, since all observations allow a reasonable probability that this is the case.

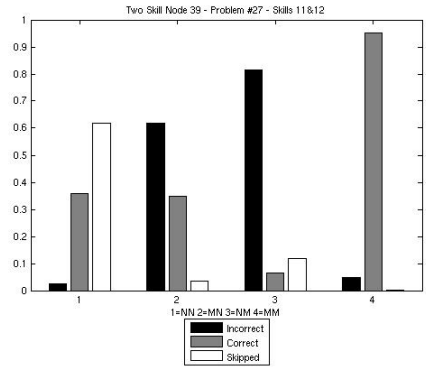
Figure 14(b) shows that when we observe a correct answer on Problem 26 which involves Skills 10 and 12, that there is a 76.42% probability that both skills are mastered. If we observe a skipped answer, we see that both



(a) Two Skill Problem Learned Conditional Probabilities: Skills 10 & 11



(b) Two Skill Problem Learned Conditional Probabilities: Skills 10 & 12



(c) Two Skill Problem Learned Conditional Probabilities: Skills 11 & 12

Figure 14: Learned Conditional Probabilities For Flat Skill Model With Informed Priors: Two Skill Problems Involving Skills 10, 11, and 12

skills are not mastered with probability 51.60%. If the answer is incorrect there is a 87.88% probability that only Skill 12 is mastered, and a 65.88% probability that only Skill 10 is mastered. This is in contrast to Problem 25 (Figure 14(a)) is not surprising since Skill 10 (corresponding angles) and Skill 11 (supplementary angles) are more closely related than Skill 10 and Skill 12 (sum of angles in a triangle). The way in which the skills rely on each other within each problem may be very different. We will see that this theory holds for the grouping of Skill 11 and Skill 12 as well.

Figure 14(c) shows that when we observe a correct answer on Problem 27 which involves Skills 11 and 12, that there is a 95.19% probability that both skills are mastered. If we observe a skipped answer we see that both skills are not mastered with probability 61.97%. If the answer is incorrect there is a 81.54% probability that only Skill 12 is mastered, and a 61.85% probability that only Skill 11 is mastered. This is similar to the findings for the previous problem, except in this problem we see that it is more probable to get the answer correct when only one skill is mastered (Skill 11), although it is still most likely the answer will be incorrect.

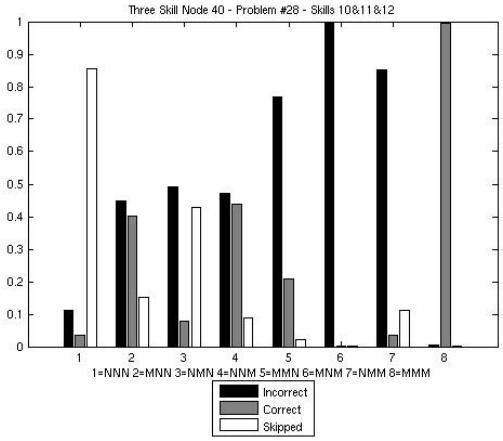


Figure 15: Learned Conditional Probabilities For Flat Skill Model With Informed Priors: Three Skill Problem Involving Skills 10, 11, and 12

The three-skill problem is the most complex. Figure 15 shows that when we observe a correct answer on Prob-

lem 28 which involves Skills 10, 11 and 12, that there is a 99.64% probability that all three skills are mastered. If we observe a skipped answer we see that all skills are not mastered with probability 85.57%. If the answer is incorrect there is a 99.77% probability that only Skill 12 is not mastered, a 85.05% probability that only Skill 11 is not mastered, and a 76.92% probability that only Skill 10 is not mastered. If only two of the three skills are mastered the odds are high that the student will attempt the problem and get the answer incorrect. This may be due to student overconfidence that he or she knows how to do the problem when that is only partially true. If only one of the three skills is mastered, the outcome is not as clear. It is still probable the answer will be incorrect, but there are unexpected peaks that a correct answer may imply that only Skill 10 or only Skill 12 is mastered. However, this is not the case when only Skill 11 is mastered, where the logical outcome is evident: a student is likely to skip the problem or answer it incorrectly. This dichotomy may stem back to the initial estimate of student skill mastery levels where we learned that Skills 10 and 12 are skills which are more likely than not to be not mastered.

- ***Inference:***

  We will look at individual students to show the inference results. We will look at the observations of the first set of 7 problems involving Skills 1, 2 and 3. See Figure 4 for linkings between problems and skills.

Student 1	Skill 1	Skill 2	Skill 3
Mastered	52.87%	11.11%	37.72%
Not Mastered	47.13%	88.89%	62.28%

Table 4: Student 1 Inferred Skill Mastery Given the Following Observed Pretest Answers:
Prob1 = incorrect, Prob2 = skipped, Prob3 = correct, Prob4 = skipped, Prob5 = incorrect, Prob6 = skipped, Prob7 = skipped.

We notice that Student 1 has skipped a majority of the problems. Thus, we are not surprised to see that the inference tends to point to the skills being not mastered.

Student 2	Skill 1	Skill 2	Skill 3
Mastered	32.44%	17.99%	90.12%
Not Mastered	67.56%	82.01%	9.88%

Table 5: Student 2 Inferred Skill Mastery Given the Following Observed Pretest Answers:
Prob1 = correct, Prob2 = correct, Prob3 = correct, Prob4 = correct, Prob5 = correct, Prob6 = incorrect, Prob7 = correct.

We notice that Student 2 has answered Problem 6 incorrectly, which involves skills 2 and 3. Evidently this infers that he or she is likely to have skill 3 mastered, but not skill 2.

The complex linkings of multiple skills to multiple problems coupled with the fact that some skills are simply more difficult than others, means that the inference results may not always make sense to the human eye. We hope to at least see an increase in skill mastery for students from pretest to posttest.

Student 1	Skill 1	Skill 2	Skill 3
Mastered	82.67%	99.72%	73.88%
Not Mastered	17.33%	0.28%	26.12%

Table 6: Student 1 Inferred Skill Mastery Given the Following Observed Posttest Answers:
Prob1 = correct, Prob2 = correct, Prob3 = correct, Prob4 = correct, Prob5 = correct, Prob6 = correct, Prob7 = correct.

Notice that Student 1 has answered every problem in this set correctly on the posttest. The difference in percentage is thus based on the difficulty of the skills based on overall student performance on each skill. Skill 2 is easier than Skill 1 which is easier than Skill 3.

Student 2	Skill 1	Skill 2	Skill 3
Mastered	60.46%	99.68%	51.27%
Not Mastered	39.54%	0.32%	48.73%

Table 7: Student 2 Inferred Skill Mastery Given the Following Observed Posttest Answers:
Prob1 = correct, Prob2 = correct, Prob3 = correct, Prob4 = correct, Prob5 = correct, Prob6 = incorrect, Prob7 = incorrect.

For Student 2, we see a drop in the inferred mastery level percentage of Skill 3 from 90% to 51% from pretest to posttest. This may be because in the pretest Skill 3 is the only skill that is most likely to be mastered, so the percentage may be over estimated to compensate for particular observations given that the other skills are unmastered. In the posttest there is no need to compensate, since all skills are likely to be mastered.

We do see the increase we expected when looking at inference of the same students from pretest to post test. Now all three skills for both students are more likely to be mastered than not.

5.2 Hierarchical Skill Models

Note that maximum BICs and average BICs are based on 50 random runs.

5.2.1 Hierarchical Problem Group Skill Model

- ***BIC Results:***

Max BIC	Average BIC	Standard Deviation	Variance
-5649.5	-5759.9	50.987	2599.7

Table 8: Bayesian Information Criterion For Hierarchical Problem Group Skill Model

The hierarchical skill model based on problem groups has worse BIC scores than the flat skill model with informed priors. This problem group model may not be adding any extra information than the flat model since the 4 sets of skills are already dependent on each other through many of their child nodes.

5.2.2 Hierarchical Skill Group Model

- ***BIC Results:***

Max BIC	Average BIC	Standard Deviation	Variance
-5695.2	-5806.1	59.00	3481.0

Table 9: Bayesian Information Criterion For Hierarchical Skill Group Model

The hierarchical skill model based on skill groups has worse BIC scores than the flat skill model with informed priors. In general, making a hierarchical model based on category seems like a sound idea. However, in this case, it is not likely that students are good at triangle skills but bad at rectangle skills because this is not how

math is taught nor understood. Math skills are learned more in layers, where harder skills build upon easier ones. Thus the Hierarchical Skill Difficulty Model seems like the closest fit, conceptually, to reality.

5.2.3 Hierarchical Skill Difficulty Model

- ***BIC Results:***

Max BIC	Average BIC	Standard Deviation	Variance
-4927.7	-5041.1	47.4178	2248.4

Table 10: Bayesian Information Criterion For Hierarchical Skill Difficulty Model With Uniform Priors

We see that the Hierarchical Skill Difficulty Model yields the maximum BIC, as well as the highest average BIC of all of out models. Thus, we can conclude that this has the best structure of our BNs to fit our data. This may be because some sets of skills/problems which were independent previous are now relevant to each other. For example, Skill 2 (area of a right triangle) is actually a subset Skill 8 (area of a triangle), but in the flat BN these skills are not linked, and are no way dependent on each other. However, in this model, Skill 2 and Skill 8 are conditionally dependent on each other. Recall that the skills and their associated problems were originally split up into the following 4 independent sets of skills: $\{1,2,3\}$, $\{4,5,6\}$, $\{7,8,9\}$, $\{10,11,12\}$. In this model, these sets of skills are no longer independent. Thus, we see that this model is actually more informative than the flat skill model.

Recall that in the flat skill model experiments, we showed that our model with informed priors scored a higher BIC then the model with uniform priors. We would like to see if we can improve upon our current best model, the Hierarchical Skill Difficulty Model with uniform priors, by using informed priors. But, the method described in Section 4.1.3 only works for root nodes and there are only 3 root nodes in this model structure. Nonetheless, it is reasonable to assume that some informed priors is better than none.

Max BIC	Average BIC	Standard Deviation	Variance
-4956.0	-5042.0	44.6162	1990.6

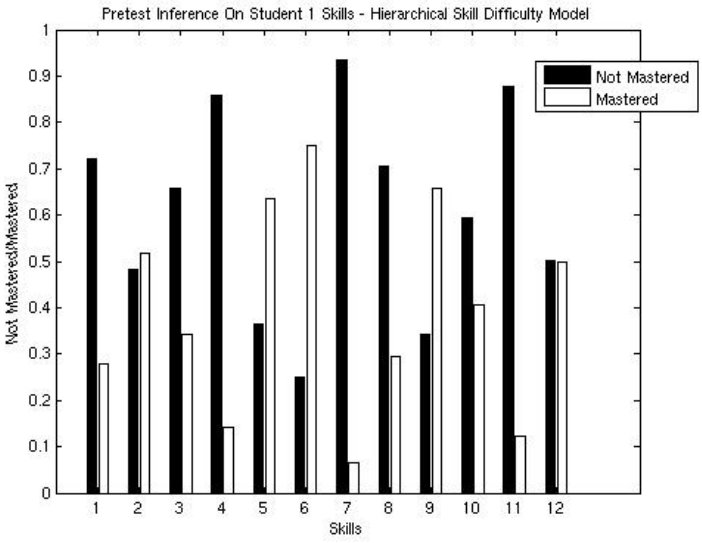
Table 11: Bayesian Information Criterion For Hierarchical Skill Difficulty Model With Partially Informed (the 3 root nodes only) Priors

We see that the difference is small, but partially informed priors do not make the model better. This could be because the majority of the nodes' priors are still set uniformly, and thus this partially informed method is skewing the results because the uniform priors are being misinterpreted as informed.

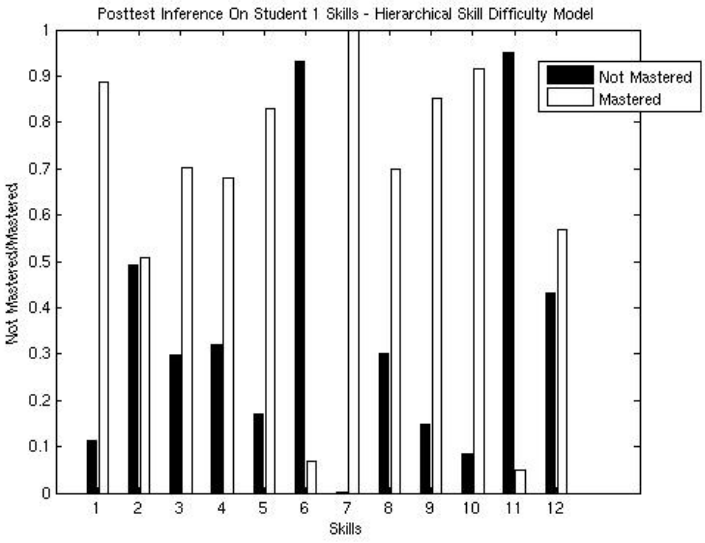
- ***Inference:***
We will look at individual students to show the inference results of the best model: Hierarchical Skill Difficulty Model With Uniform Priors. We will look at the observations of the last set of 7 problems involving Skills 10, 11 and 12. See Figure 7 for linkings between problems and skills.

Problem 1	Problem 2	Problem 3	Problem 4	Problem 5	Problem 6	Problem 7
Incorrect	Skipped	Correct	Skipped	Correct	Skipped	Skipped
Problem 8	Problem 9	Problem 10	Problem 11	Problem 12	Problem 13	Problem 14
Skipped	Incorrect	Skipped	Incorrect	Skipped	Skipped	Skipped
Problem 15	Problem 16	Problem 17	Problem 18	Problem 19	Problem 20	Problem 21
Skipped	Incorrect	Correct	Skipped	Skipped	Incorrect	Skipped
Problem 22	Problem 23	Problem 24	Problem 25	Problem 26	Problem 27	Problem 28
Correct	Correct	Skipped	Incorrect	Skipped	Incorrect	Skipped

Table 12: Student 1 Observed Pretest Answers



(a) Student 1 Pretest Skill Mastery Inferred Probabilities



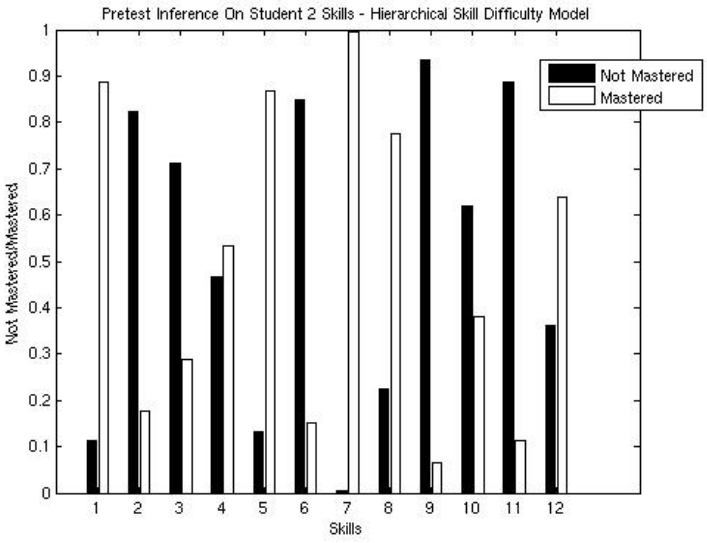
(b) Student 1 Posttest Skill Mastery Inferred Probabilities

Figure 16: Student 1 Improvement in the probabilities of overall student skill mastery from pretest to posttest

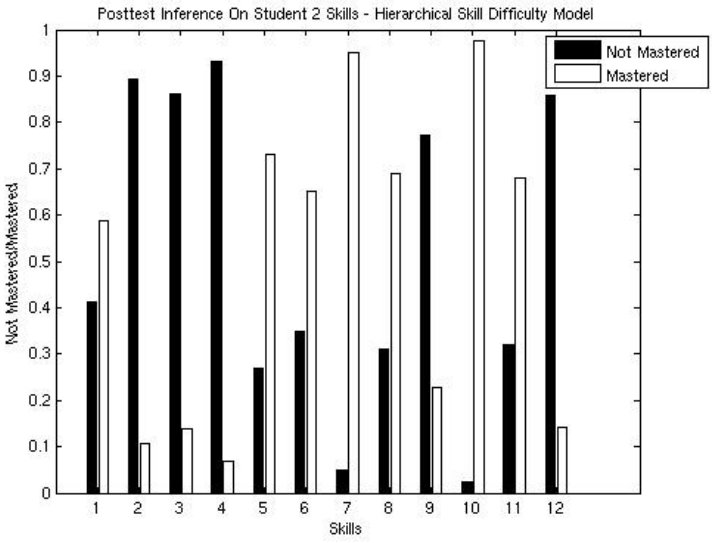
In Figure 16 we do see an overall improvement in skill mastery from pretest to posttest. This aligns with the student's performance on the tests. The test scores are calculated to evaluate individual improvement as follows: $corr/attmp$, the number of problems the students answered correctly divided by the number of problem the student attempted to answer (did not skip). Student 1 got 27% correct on the pretest and 83% on the posttest. For most skills, Student 1 shows a higher mastery in the posttest than in the pretest. However, Skill 6, which was mastered in the pretest, is not mastered in the posttest. This may be because the student did poorly on the posttest problem involving this skill. It is unclear as to whether this assumption should lead to the conclusion of mastery or non-mastery of Skill 6. Notice that Skill 7 was initially the least mastered in the pretest, but is the most mastered in the posttest. If this is a common trend among students, we can assume our tutor does an exceptionally good job of teaching Skill 7. However, Skill 11 actually increases in it's probability of being not mastered from pretest to posttest. This may show that the tutor is not doing a good job of teaching Skill 11. Regardless, these results may also be caused simply by the answers this student supplied on the tests and not the tutor's capability.

Problem 1	Problem 2	Problem 3	Problem 4	Problem 5	Problem 6	Problem 7
Correct	Correct	Correct	Correct	Correct	Incorrect	Correct
Problem 8	Problem 9	Problem 10	Problem 11	Problem 12	Problem 13	Problem 14
Correct	Correct	Correct	Correct	Correct	Incorrect	Incorrect
Problem 15	Problem 16	Problem 17	Problem 18	Problem 19	Problem 20	Problem 21
Correct	Correct	Incorrect	Incorrect	Incorrect	Incorrect	Incorrect
Problem 22	Problem 23	Problem 24	Problem 25	Problem 26	Problem 27	Problem 28
Incorrect	Correct	Correct	Incorrect	Incorrect	Correct	Incorrect

Table 13: Student 2 Observed Pretest Answers



(a) Student 2 Pretest Skill Mastery Inferred Probabilities



(b) Student 2 Posttest Skill Mastery Inferred Probabilities

Figure 17: Student 2 Improvement in the probabilities of overall student skill mastery from pretest to posttest

For Student 2, the number of skills that are mastered does not increase from pretest to posttest. This results is expected since the test score of Student 2 did not improve same from pretest to posttest. Specifically, this student got 51% of the problems correct on both the pretest and the posttest. The inference showing that the student has 6 of the 12 skills mastered on both tests, aligns with this score. Note that Student 2 has not skipped any problems (See Table 13). We have previously learned that skipped problems are more closely correlated to non-mastery than incorrect answers. If many students skip no problems, this idea may need modification. When a student does not skip any problems, then incorrect answers become more meaningful in identifying non-mastery. Future work may include initial clustering of students. Then we must learn different Bayesian networks based on this clustering, and finally, run inference for a new student on which BN their clustering groups him or her.

6 Conclusions

In summary, a successful graphical model can be built to infer student mastery of skills. We used Bayesian Information Criterion to evaluate several models, both flat and hierarchical. Our best model is the Hierarchical Skill Difficulty Model With Uniform Priors. We build a Bayesian network based on the relation between problems (observed data) and the skills (hidden data) associated with each problem. In the best model, related skills are also linked such that parent nodes are more basic skills that should be mastered before their more difficult child node skills are mastered. We learn the hidden parameters of the model using Expectation Maximization (EM) and training data from student pretests as observations. This gives us conditional probability tables over all students. Next we tested the learned parameters by doing inference on new student data, giving us an estimate of a student's initial skill levels. Finally, we ran inference on the student's posttest data to show the improvement in skill mastery, which proves that the Wayang Tutor is succeeding at teaching students these mathematical skills.

We can now make predictions not only of how a student will do on a particular problem, but also of how much a student will improve overall. In addition, we learned, that if a skill is not mastered the student is more likely to skip the problem than answer incorrectly. Furthermore, we illustrated student improvement from pretest to posttest with raw counts, learned parameters, and inference, again highlighting the positive affect of the Intelligent Tutoring System.

7 Future Work

Future work will involve using the parameters learned from this Bayesian network to run inference during runtime to gather prior skill masteries for students, which will be used to enhance the policy selection of the Intelligent Tutoring System. Immediately after the pretest is taken, inference will be done using the student pretest answers as evidence. This will give us an estimate of the student's skill mastery before the tutoring session, which will be utilized by the machine learning algorithm within the ITS to individualize the tutoring session to improve each student's learning. Inference can also be done on the student's posttest to estimate exactly which skills have improved and by how much. Eventually, we would like to take this estimate a step further to predict what score the student will get on the Scholastic Achievement Test (SAT).

Additionally, clustering of the students based on various features may be an important way of separating different types of students whose inference may need to be done on different Bayesian networks. For example, some students never skip problems during the pretest and/or posttest, some students never ask for hints during the tutoring session, and some students actually score worse on the posttest than the pretest. Therefore, information on various students may need to be learned in different ways. We also have many more math-SAT skills to include into the learning portion of the system.

8 References

AMWB04 I. ARROYO, T. MURRAY, B. WOOLF, and C. BEAL, Inferring Unobservable Learning Variables from Students Help Seeking Behavior. *James C. Lester, Rosa Maria Vicari, Fbio Paraguau (Eds.): Intelligent Tutoring Systems, 7th International Conference*, ITS 2004, Macei, Alagoas, Brazil, Proceedings. Lecture Notes in Computer Science 3220 Springer 2004.

BARW03 C. BEAL, I. ARROYO, J. ROYER, and B. WOOLF, Wayang Outpost: An intelligent multimedia tutor for high stakes math achievement tests, *American Educational Research Association annual meeting*, Chicago IL, 2003.

BW00 J. BECK, and B. WOOLF, High-level Student Modeling with Machine Learning, *Proceedings of the International Conference on Intelligent Tutoring Systems*, 5: 584 593, 2000.

BWB00 J. BECK, B. WOOLF, and C.BEAL, ADVISOR: A machine learning architecture for intelligent tutor construction, *Proceedings of the National Conference on Artificial Intelligence*, 17: 552557, 2000.

DLR77 A. DEMPSTER, N. LAIRD, and D. RUBIN, Maximization-likelihood from Incomplete Data via the EM Algorithm, *Journal of Royal Statistical Society, Series B*, 1977.

F97 N. FRIEDMAN, Learning Belief networks in the Presence of Missing Values and Hidden Variables, *Fourteenth International Conference on Machine Learning*, 1997.

HQ79 E. HANNAN, and B. QUINN, The determination of the order of an autoregression, *Journal of the Royal Statistical Society, Series B*, 41, 190195, 1979.

ROLE05 A. JONSSON, J. JOHN, H. MEHRANIAN, I. ARROYO, B. WOOLF, A. BARTO, D. FISHER, and S. MAHADEVAN, Evaluating the Feasibility of Learning Student Models from Data, *AAAI Workshop on Educational Data Mining*, Pittsburgh, PA, 2005.

J01 M. JORDAN, An Introduction to Graphical Models, Unpublished book, 2001.

MM01 M. MAYO, A. MITROVIC, Optimising its behaviour with bayesian networks and decision theory, *International Journal of Artificial Intelligence in Education*, 12: 124 153, 2001.

M01 K. MURPHY, The Bayes Net Toolbox for Matlab, *Computing Science and Statistics*, vol. 33, 2001.

OMM04 S. OSENTOSKI, V, MANFREDI, S. MAHADEVAN, Learning Hierarchical Models of Activity, *IEEE/RSJ International Conference on Robots and Systems*, 2004.

S78 G. SCHWARZ, Estimating the Dimension of a Model, *Annals of Statistics*, 6, 461-464, 1978.