Proceedings of the 12th International Conference on Artificial Intelligence in Education. C.K. Looie, G. McCalla, B. Bredeweg, and J. Breuker, editors. Amsterdam: IOS Press, 2005. pp 33-40.

Inferring learning and attitudes from a Bayesian Network of log file data

Ivon ARROYO, Beverly Park WOOLF
Department of Computer Science University of Massachusetts,
Amherst, MA. Contact: ivon@cs.umass.edu

Abstract. A student's goals and attitudes while interacting with a tutor are typically unseen and unknowable. However their outward behavior (e.g. problem-solving time, mistakes and help requests) is easily recorded and can reflect hidden affect status. This research evaluates the accuracy of a Bayesian Network to infer a student's hidden attitude toward learning, amount learned and perception of the system from log-data. The long term goal is to develop tutors that self-improve their student models and their teaching, dynamically can adapt pedagogical decisions about hints and help improve student's affective, intellectual and learning situation based on inferences about their goals and attitude.

1 Introduction

The advent of the Internet has promoted Web-based learning environments that facilitate collection of enormous student data, as a result of centralized servers and databases. Log data permit the analysis of fine-grained student actions that characterize fading of students' mistakes or the reduction of time on task [1]. This large amount of log-data not only characterizes learning, it has also changed the way we conduct science, in the sense, that the scientific cycle has been altered. We can now start from data rather than from a theory. The analysis of learning curves, may also show how to structure and better understand the domain being taught [2]. Learning to profit from this log file data to enhance our learning environments is one of the next greatest challenges for the AIED community.

Our long-term vision is to create a new generation of tutoring software that self-improves as users interact with it. As a first step towards creating tutors that "learn how to teach," we describe a methodology in which very crude and generic descriptors of students' behavior in a tutoring system are used to predict a students' goals, attitudes and learning for a large database of student actions. We present a model that shows that such dependencies do exist, describe the methodology we used to find a good model, evaluate its accuracy and identify the accuracy of alternative models. The final goal is to use the model to impact students' learning and positive attitudes towards learning, and to eventually create a module in the tutor that recomputes the model as new data arrives.

This community has made recent attempts to link students' attitudes and learning to actual behavior [3, 4, 5, 6]. Aleven proposed a taxonomy of help seeking bugs and possible hints to be given by the tutoring system to encourage positive behaviors. Zhou and Conati built a Bayesian model to infer students' emotions and personality for a mathematics game. Baker observed students' behavior and classified those "gaming" the system. This paper is an integration of that past work; it merges motivation, learning, and misuse of tutoring systems in one single Bayesian model, presenting the

complexity of behaviors linked to students' affect and cognition, advocating for data-driven models that integrate cognition, motivation and their expression with different behavioral patterns.

2 Data sources: Integration of survey and log files data summaries

This section describes the first step in the methodology to use observable student behavior to infer student learning and attitudes, specifically how to identify dependencies between hidden and observable variables. We used log data from Wayang Outpost, a multimedia web-based tutoring system for high school mathematics [7] to predict affective variables, e.g., the student liked the experience, was learning and was trying to challenge himself. Wayang Outpost provides step-by-step instruction to the student in the form of animations, aided with sound, which help students solve the current problem and teach concepts that are transferred to later problems. Problems are presented in a random order (no adaptive problem selection). Every interaction of student and tutor is logged in a server-side relational database, allowing researchers to record variables such as time spent, number of problems seen and speed of response The data used in this study comes from a population of 230 15-17 year-old students from two high schools in rural and urban areas in Massachusetts. Data from an older study of 70 students in fall 2003 and a study of 150 students in spring 2004 providing a large set of data-points. Students took a pretest and then used Wayang Outpost for about 2-3 hours. Students were provided headphones, as the help contains text and animations with audio. After using the tutor, students took a post-test, and answered a survey to identify their hidden attitudes and learning.

Table 1 describes the instruments used at the end of the study, with code names for each question in bold. We used variables of affect that overlapped with those used by de Vicente and Pain as shown in italics. Their motivational variables were created with a similar purpose, i.e., to infer motivation based on students' interaction with a tutor by creating rules from interviews with students.

In addition, we identified observable student behavior, specifically students' ways of interacting with the system, particularly the effort or focus of attention at specific moments. These variables describe generic problem-solving behavior, e.g., mistakes, time, help requests and behavior in problems where the student requests help. This observable behavior falls into four categories: (1) Problem-solving behavior, e.g., average incorrect responses, specifically for those problems where help was requested; average seconds spent in any problem and where help was requested; and seconds spent between making attempts. (2) Help activity, e.g., the moment when students asked for help, including an estimate of the amount of help, how much effort (and focus of attention) the student spent; average hints requested per problem; average hints in helped problems (when a student asks for help, how much help does she request?); average seconds

Student Perceptions of the tutor.
Learned? Do you think you learned how to tackle SAT-Math problems by using the system?
Liked? How much did you like the system? *Satisfaction, Sensory Interest*
Helpful? What did you think about the help in the system? *Satisfaction,*
Return? Would you come back to the web site to use the system again if there were more problems and help for you to see? How many more times would you use it again? *Satisfaction*
Interaction with the tutor.
Audio? How much did you use the audio for the explanations? *Sensory Interest, Effort.*
Attitudes towards help and learning
Seriously try learn. How seriously did you try to learn from the tutoring system? *Effort*
Get it over with (fast). I just wanted to get the session over with, so I went as fast as possible without paying much attention. *Effort*
Challenge. I wanted to challenge myself: I wanted to see how many I could get right, asking as little help as possible. *Independence, Challenge.*
Don't care about help. I wanted to get the correct answer, but didn't care about the help or about learning with the software. *Effort.*
Help fading attitude. I wanted to ask for help when necessary, but tried to become independent of help as time went by. *Independence.*
Other approaches. I wanted to see other approaches to solving the problem, and thus asked for help even if I got it right. *Cognitive Interest*
Fear of Wrong. I didn't want to enter a wrong answer, so I asked for help before attempting an answer, even if I had a clear idea of what the answer could be. *Confidence*

Table 1. Post-test of student attitudes. Students' perceptions about, interaction with and attitudes towards the tutor. Motivation listed in italics from Vincent and Pain (2002)

spent in helped problems (time/effort the student invested when she asked for help); the percentage of helped problems in the tutoring session (how often the student asked for help). (3) Help timing, e.g., the timing of when help was sought as a percentage of all helped problems: *help before making an attempt*; *help after making an attempt*; *help after entering the correct answer*. (These variables paint a good picture of student help seeking and overall behavior in the tutor.) (4) Student descriptors, past experience affects a student's attitudes and learning; correct and incorrect answers in the pre-test; gender, as we had seen gender differences both in attitudes and interactions with the tutors in the past. (5) Other potentially useful variables were not considered, e.g., the standard deviation of mistakes, time, etc. As an example, the standard deviation is inexpensive to compute from relational databases. Last, measures of fading of help and mistakes (some learning curve estimate per student) would be useful as a way to observe learning while the student uses the system. The next section describes an exploratory analysis to find the connection between these concrete observable variables and the more abstract and hidden ones derived from the survey.

3 Identifying dependencies among variables

Bi-variate Pearson correlations were computed to search for links among the hidden and observed variables. Figure 1 shows the high number of significant correlations found among help seeking attitudes, help seeking behaviors, perceptions of the system, gender and other behaviors, such as

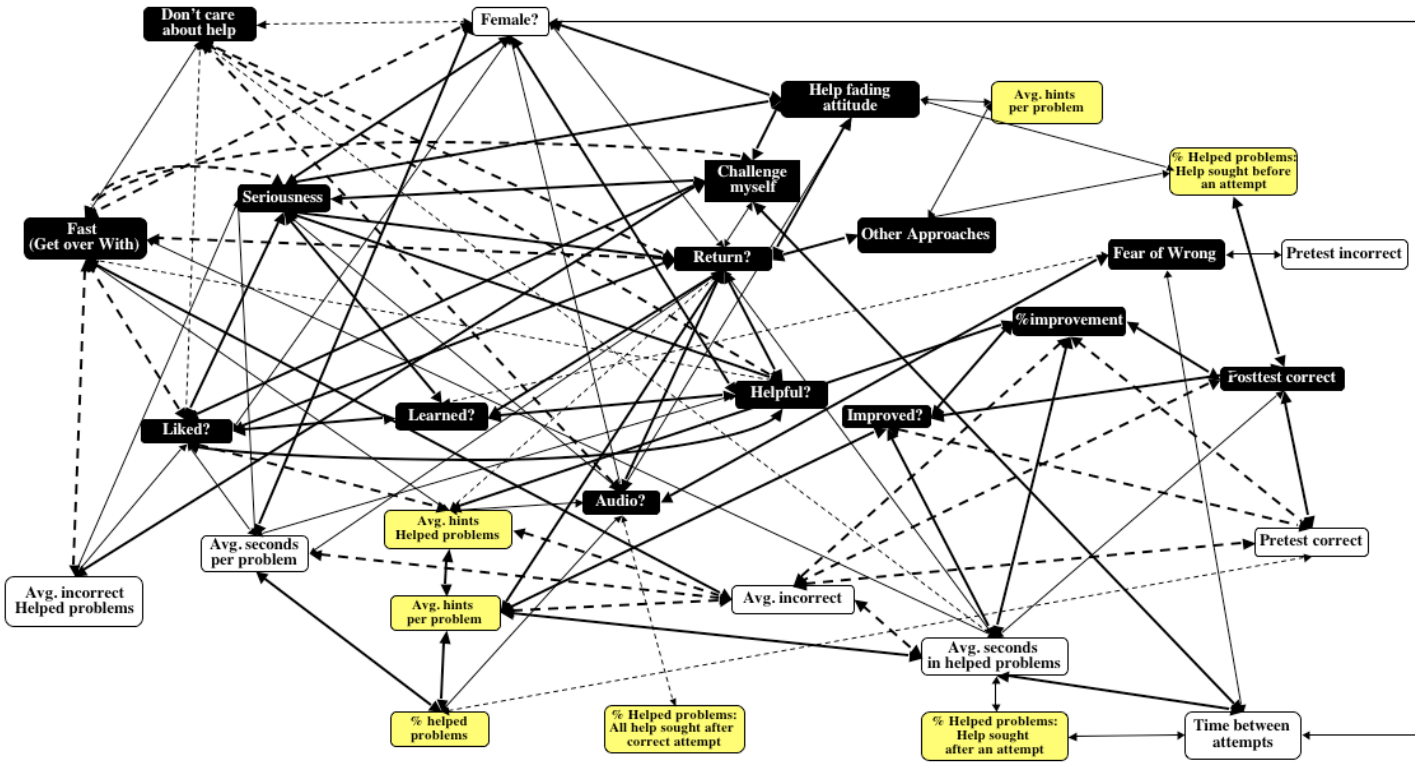

Figure 1. Correlations between hidden and observed variables. Variables that describe a student's observed interaction style (light colored nodes) are correlated with the students' hidden attitudes, feelings and learning (dark nodes) derived from the survey. Line weight indicates correlation: dashed line (- -) indicates a negative correlation; lines (—) indicate a positive correlation; thick lines indicate $p<0.01$ and $R^2>0.09$ - light lines indicate correlations of $p<0.05$

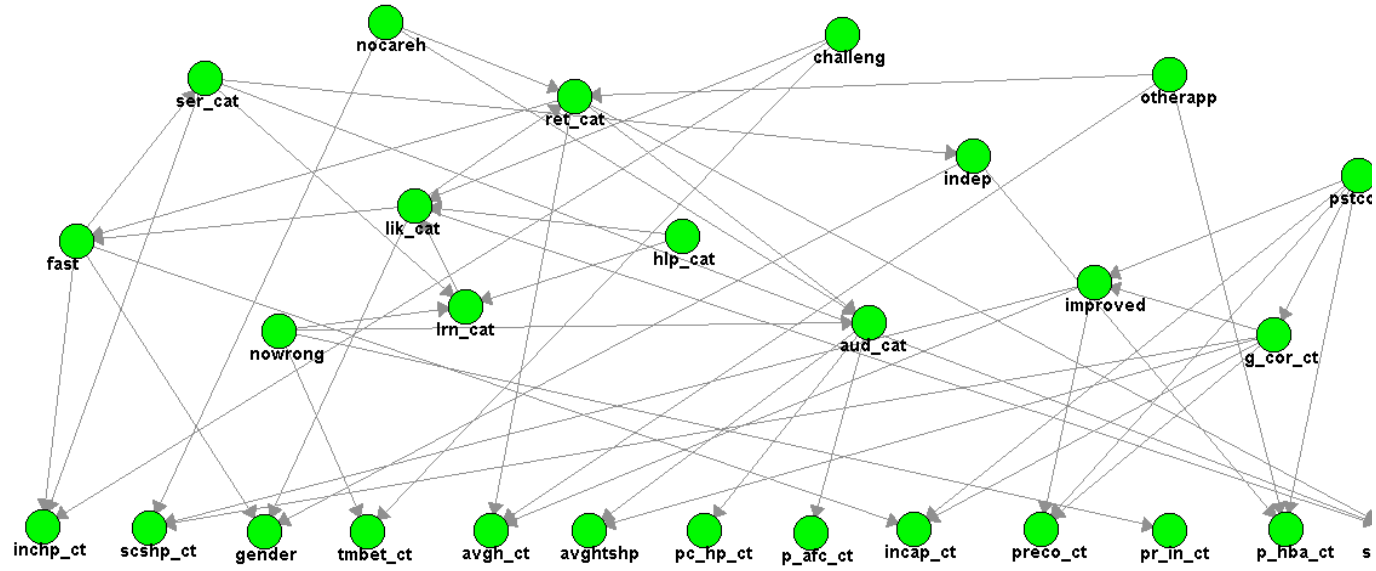

Figure 2 Structure of a Bayesian Network to infer attitudes, perceptions and learning. Nodes are the same as those in Figure 1. The bottom or leaf nodes are observable, all others are hidden. pc=percent; avg=average; no=number; lrn=learn; ret=return; pst=post; h=hint; ct=count; hp=help; co=correct; tm=time.

problems seen and how often a student reported hearing the audio for explanations. Thick lines indicate a significant correlation with $p<0.01$ and an $R2>0.09$, while light lines indicate significant correlations with strength $p<0.05$ and R2 sometimes lower than 0.09. As expected, there are dependencies among variables within a group of hidden variables, such as significant correlations among the variables that describe perceptions towards the system.

Despite type I error, we may attempt to interpret these dependencies among variables to understand students' use of the system. The gain from pre- to post-test (% improvement) is not correlated to 'average hints seen per problem', but it is correlated to 'average hints seen in *helped* problems.' Thus, there is a high probability that students who search deeply for help are more likely to learn. Other variables that relate to % improvement indicate that this relationship is more complex, since learning gain is not positively correlated with 'time spent in a problem,' but it is correlated to 'time spent in those problems where help was seen.' This suggests that spending much time struggling in a problem and not seeing help will produce fewer gains. Learning is inversely correlated to average incorrect attempts per problem, suggesting that students who make many incorrect responses per problem do not display a large improvement. Still, these correlations are not too strong (in general, neither of them by themselves accounts for more than 15% of the variance). Thus, making any definite conclusions about what leads or does not lead to learning is premature. A model that integrates all these independent variables together should allow for a better prediction of the dependent variables that indicate success in a learning environment.

Students' general perceptions and attitudes are also correlated to many concrete behaviors in the tutor. In general, making mistakes while asking for help seems to be a positive action and is correlated to 'seriousness' and 'liking of the system,' though not directly associated to higher learning gains. It is also correlated to the 'challenge' attitude, showing that students might want to make an attempt even if they risk a wrong answer. One interesting dependency is that a high number of mistakes per problem is

correlated to a higher chance of a student saying he/she wants to 'get over with' (probably just clicking through to get the answer). However, making a high number of mistakes in problems where they do request help is linked to a lower likelihood of wanting to 'get over with' the session. Interestingly, there are no strong correlations between a student's perceptions of learning and actual learning. This is consistent with past research reports that students may overestimate or underestimate their learning, and that students' perception of learning may not reflect actual learning. This hypothesis is supported by the fact that hidden positive student attitudes are correlated with behaviors that lead to high learning gains (e.g. 'improved?' and 'return?' are both positively correlated to 'average hints per problem'; 'Get over with' and 'Don't care about help' are negatively correlated to 'average seconds in helped problems' which is positively correlated to '% improvement' and 'post-test correct').

4 Building an integrated model of behavior, attitude and perception

The previous sections described the first step in a methodology to infer student learning gains data: A correlation was identified between hidden and observed variables. The next step is to build a complex Bayesian Network to diagnose a student's hidden variables given only observed variables. If an accurate inference of attitudes and learning can be made while the student is using the system, then the tutor can anticipate a student's posterior answers about perceptions of the system. We created a student model that is informed about past correlations results and can integrate real-time observable behavior of a student with more abstract and hidden attitudes and beliefs.

Bayesian networks that learn from data capitalize on the complex network of dependencies among variables, as they to predict the probability of the truth of some unknown variables, given that a few others have been observed. There are many possible Bayesian networks to build for a set of variables, and well-known algorithms to learn the structure of a BBN. However, these algorithms rely on searching for the best possible network among all possible networks. This makes the search of the best model expensive, increasing exponentially with the number of nodes. The urge to find a methodology to approximate a "good enough" model in non-exponential time led us to construct the Bayesian model shown in Figure 2 that relies on the knowledge gained from correlation analysis in Figure 1, based on the fact that links in a Bayesian net express a dependency and variables that are not correlated are unlikely to be dependent on each other. A directed acyclic graph was created by: 1) eliminating the correlation links among observable variables (a naïve approach); 2) giving a single direction to the links from non-observable to observable variables (the observable variables being the leaf nodes, also known as the "outputs" or the "effects"); 3) for links between non-observable variables, creating intuitive unidirectional links (from the nodes that are more likely "causes" to the nodes that are more likely effects; 4) eliminating links that create cycles, leaving in the links that have a higher correlation strength. This resulted in a directed acyclic graph (DAG) that gave the structure of the Bayesian Network (Figure 2). Next, the parameters to the network were generated by: 1) discretizing all variables in two levels (high/low) with

'Fear of wrong'	'Challenge'	Time between attempts	Cases	Probability	
False	False	Low	43	0.64	(1)
		High	24	0.36	(2)
	True	Low	35	0.42	(3)
		High	48	0.58	(4)
True	False	Low	8	0.50	(5)
		High	8	0.50	(6)
	True	Low	7	0.32	(7)
		High	15	0.68	(8)

Table 2. Learning the conditional probability tables (CPT)
Maximum likelihood to learn conditional probability tables for 'fear of wrong' node from students' data.

a median-split; 2) simplifying the model further by discarding existing links whose connecting nodes do not pass a Chi-Square test (the dependency is not maintained after making the variables discrete); 3) creating conditional probability tables (CPTs) from the cross-tabulations of the students' data. As an example, Table 2 shows the conditional probability table attached to the node 'Time Between Attempts.'

The CPT is built from cases or student instances, with the "maximum likelihood" method for parameter learning in discrete models [8].

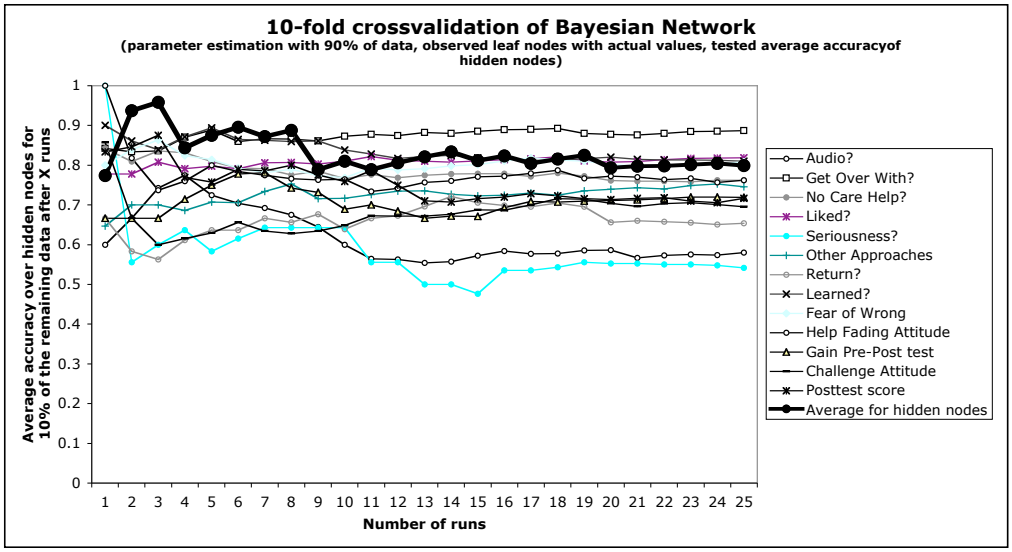

Figure 3. Accuracy of inferred nodes.
This validation test measured the accuracy of the Bayesian network to learn the hidden nodes (attitudes and learning improvement) with 10% of the log-data, after the CPT was learned with 90% of the data. The graph shows the percentage of hits for all hidden nodes.

The CPT table attached to the observable node 'time between attempts' has two parents: 'fear of wrong' and 'challenge,' see Figure 1. Many interesting probabilities result: when a student reports a 'challenge' attitude, the chance of spending a large amount of time between subsequent attempts is higher than when a student does not report wanting to 'challenge' herself (compare (4) to (2) and (8) to (6) in Table 2). When a student reports 'fear of the wrong answer,' there is also higher likelihood of spending a long time between attempts (compare (8) to (4) and (6) to (2) in Table 2). The probability of spending a large amount of time between attempts is highest when the student reported both 'fear of wrong' and 'challenge attitude;' it is lowest when the student did not report 'fear of wrong' or did not want to 'challenge' herself.

5 Model accuracy

A 10-fold cross-validation was performed to test the accuracy of the model. The following process was repeated 25 times: the conditional probability tables were learned from 90% of students' data; the remaining 10% was used to test the model. The model was tested in the following way: the leaf nodes (observable student behavior within the tutor) were instantiated (observed) with the behavior that the student displayed (including gender and pre-test correct and incorrect). Then, the hidden nodes (attitudes, learning improvement, post-test score, perceptions of helpfulness) were inferred with the Bayesian network. If the probability of true was higher than 0.7 and the true value of the inferred node

was 1 (i.e., true, or high, depending on the variable), a "hit" was produced. A hit was also produced for an inference lower than 0.3 and the actual value being a 0 (i.e., false, or low, depending the variable). A "miss" was detected when the inference was higher than 0.7 but the actual value was a 0 (or false, or low). If the inference was within the interval (0.3, 0.7), the inference was considered too uncertain and thus did not "fire." The accuracy for each node was computed as the ratio of hits to the total (hits + misses). Figure 3 shows the percentage of hits for all hidden nodes, after 25 runs. Nodes with higher accuracy also contain less uncertain inferences ; 90% of the 'get-over-with' inferences fell outside the (0.3,0.7) interval, while only 11% of the inferences for the 'seriousness' attitude falls outside of that interval (89% of the inferences were considered uncertain). Because only "certain" inferences were taken into account, the average trend is closer to the most accurate nodes. For some nodes, the accuracy is low, such as for students' reported 'use of audio,' or students' 'seriousness' while using the system. Seriousness relies on 'audio' for its inferences, and audio is hard to predict from observable behaviors (the audio was very embedded in the hints, so seeing or not seeing hints is not a good discriminant to determine whether they are hearing the audio). We believe part of the mistake has to do with the way the question was phrased, which tried to capture students' predisposition to have the headphones on. However, some students did not use the audio because of other causes, such as the wires being uncomfortably short to be plugged at the back of the computer.

6 Understanding the model

We think it is important to produce models that are inspectable. We know this model produces inferences with an average of 80% of accuracy. We may now query the model to gain knowledge and answer questions about students' learning: How does a student who demonstrates a high gain from pre-test to post-test interact with the tutor compared to one who doesn't learn? How does a motivated student behave compared to one who doesn't seem motivated? We may query the model to learn about students' learning.

Table 3 shows how setting some observable-behavior nodes to different values, produces different inferences of '% improvement from pre-test to post-test.' Students with a low pre-test score are more likely to learn, as P (%improvement = high) is always higher than the chance of low improvement. This is not surprising, as students who have a low pre-test score have also more room for improvement. We may note situations that lead to high or low learning with high certainty, by focusing just on the inferences that are >0.7. For low pretest score students, seeking deeply for hints seems important for learning. On the other hand, high pretest score students should avoid spending little time in those problems where help is seen. Probably the current help is not very helpful for these more "expert" students, who should focus on problems, think them out and practice.

Pretest score	Hints in helped problems	Seconds in helped problems	Percentage improvement	Cases (+1)	Probability	
Low	Low	Low	Low	5	0.36	(9)
			High	9	0.64	(10)
		High	Low	6	0.35	(11)
			High	11	0.65	(12)
	High	Low	Low	4	0.24	(13)
			High	13	0.76	(14)
		High	Low	6	0.30	(15)
			High	14	0.70	(16)
High	Low	Low	Low	16	0.84	(17)
			High	3	0.16	(18)
		High	Low	8	0.62	(19)
			High	5	0.38	(20)
	High	Low	Low	12	0.80	(21)
			High	3	0.20	(22)
		High	Low	7	0.37	(23)
			High	12	0.63	(24)

Table 3. A model of learning and attitude that can be inspected. This model may be queried to gain knowledge and answer questions about students' learning learning .

A more detailed analysis of how behavior effects higher-level variables was

carried out, by removing the links to some leaf nodes from the model and seeing how that affects the overall accuracy. Figure 4 shows that when removing links to certain observable nodes, accuracy in predicting other nodes becomes diminished. For instance, we can observe how removing the node called 'incorrect responses in helped problems' (third column from the left) affects the prediction of the 'challenge' attitude, and produces more uncertain inferences. This is important if one intends to understand which behaviors predict attitudes and learning. It may also be used to simplify the model: if an immediate child is removed but the accuracy is not affected, the link to that node can be removed, as it merely promotes over-fitting. One reason may be that another neighbor may capture the same effect. Removing links to other nodes can provide a clear sense of how certain variables affect the prediction of others and provide guidelines to improve the BBN.

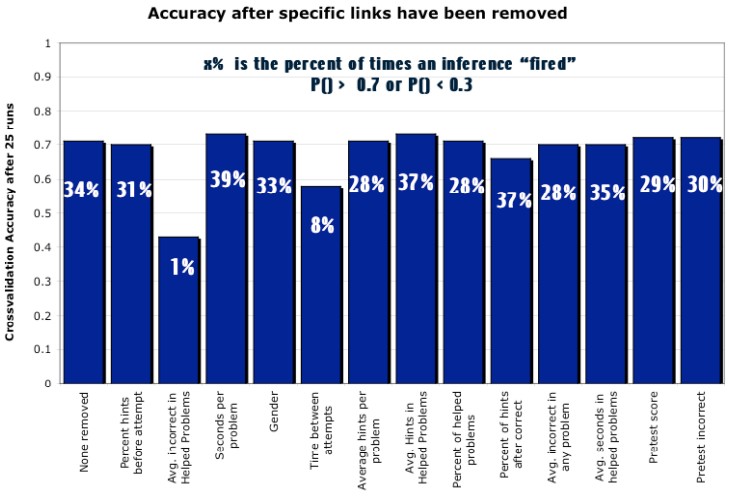

Figure 4. Removing Specific Links. Accuracy of inferences of the "Challenge Attitude" after removing some observable-behavior nodes.

7 Summary and Future Work

We have described a methodology to build a model from log-data that integrates behavioral, cognitive and motivational variables. We showed how the methodology was applied to our bank of data for a tutoring system. We showed how the model captures the complexity of variables that describe the student and capitalize on this dependency structure to infer the students' cognitive and affective state. We highlighted how machine learning methods and a classical statistical analysis can be combined to prune the search of an accurate model in non-exponential time. This is important when considering a large amount of behaviors and other variables, or when thinking about self-improving models that can be re-computed as new users arrive to the system, or after changes to the tutor's interface or any other components. After refining the model, our next step is to implement various forms of remediation that would be triggered in certain "undesirable" situations that are linked to lower learning and negative attitudes. Finally, a module will be added to the tutor that updates and re-computes the model automatically as new data arrives, following the described methodology.

8 Acknowledgements

We gratefully acknowledge support for this work from two National Science Foundation awards: 1) HRD/EHR 012080, Beal, Woolf, & Royer, "AnimalWorld: Enhancing High School Women's Mathematical Competence;" and 2) HRD/HER 0411776, Woolf, Barto, Mahdevan, Fisher & Arroyo, "Learning to Teach: the next generation of Intelligent Tutor Systems."

Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the granting agencies.

9 References

[1] Corbett, A. & Anderson, J. (1992). Knowledge tracing in the ACT Programming Tutor. The Proceedings of the 14th Annual Conference of the Cognitive Science Society. Hillsdale, NJ: Lawrence Erlbaum

[2] Koedinger, K. & Santosh, M (2004). Distinguishing Qualitatively Different Kinds of Learning Using Log Files and Learning Curves. Workshop "Analyzing Student-Tutor Interaction Logs to Improve Educational Outcomes," ITS 2004.

[3] Zhou X. & Conati C. (2003). Inferring User Goals from Personality and Behavior in a Causal Model of User Affect . In Proceedings of the International Conference on Intelligent User Interfaces, pp. 211-218.

[4] Baker, R., Corbett, A.T. and Koedinger, K.R. (2001). Toward a Model of Learning Data Representations. Proceedings of the Twenty-Third Annual Conference of the Cognitive Science Society, 45-50

[5] Aleven, V., McLaren, B., Roll, I. & Koedinger, K. (2004). Toward Tutoring Help Seeking: Applying Cognitive Modeling to Meta-Cognitive Skills. In the *Proceedings of the 7th International Conference on Intelligent Tutoring Systems* (ITS-2004). Springer.

[6] de Vicente, A. & Pain, H. (2002). Informing the detection of the students' motivational state: an empirical study. In Proceedings of the 6th International Conference on Intelligent Tutoring Systems. Lecture Notes in Computer Science. Springer.

[7] Arroyo, I., Beal, C. R., Murray, T., Walles, R., Woolf, B. P. (2004b). Web-Based Intelligent Multimedia Tutoring for High Stakes Achievement Tests. *Proceedings of the 7th International Conference on Intelligent Tutoring Systems*, 468-477, Springer.

[8] Russell, S. & Norvig, P. (2002). Artificial Intelligence: A Modern Approach (2nd Edition). Chapter 14: Probabilistic Reasoning Systems.