

Gender Differences and the Value of Choice in Intelligent Tutoring Systems

Derek Green, Thomas J. Walsh,
Paul R. Cohen, Carole R. Beal and Yu-Han Chang*

The University of Arizona, Department of Computer Science, Tucson, AZ 85721-0077

*The University of Southern California, Information Sciences Institute, Marina del Rey, CA

dtgreen@email.arizona.edu

Abstract. Students interacted with an intelligent tutoring system to learn grammatical rules for an artificial language. Six tutoring policies were explored. One, based on a Dynamic Bayes' Network model of skills, was derived automatically from the performance of previous students. Overall, this policy and other intelligent policies outperformed random policies. Among the intelligent policies, some allowed the students to choose one of three problems to work on, while others presented just a single problem at each iteration. The benefit of choice was not apparent in group statistics; however, there was a strong interaction with gender. Overall, women learned less than men, but they learned different amounts in the choice and no choice conditions, whereas men seemed unaffected by choice. We explore reasons for these interactions between gender, choice and learning.

Problem-oriented Intelligent Tutoring Systems (ITS) teach a subject by presenting problems to students to solve (e.g., Woolf, 2008; Beal et al., 2010). One form of intelligence in these systems is the *policies* that decide which problem(s) to present to a student. Generally speaking, subjects comprise skills that should be acquired in a strict or partial order, as some skills will depend on others. Good policies will respect these dependencies, and students who work under good policies will learn more than those who don't. This paper discusses two aspects of policies: whether they offer students choices, and whether they are constructed by hand or learned from data. Our research is primarily concerned with learning policies, and our experiments have been designed primarily to test the efficacy of learned strategies. However, the bulk of this paper is devoted to some surprising and consequential empirical results: Giving students choices has an effect mediated by gender, and when our ITS learned a strategy, it helped men more than women.

Section 1 describes the task domain for which the ITS was developed. Section 2 sketches the ITS itself and summarizes six policies for teaching the domain content, including one learned automatically from observations of students using the ITS. Sections 3 and 4 describe our experiment design and results, respectively.

1 The Task Domain

Our domain focused on the syntax and semantics of a subset of an artificial language. An artificial domain was constructed to allow us to evaluate tutoring policies without

the confounds associated with students' prior knowledge and expectations of familiar domains. The language contains few words, but these can be ordered to construct phrases with very different meanings. There are three types of words: nouns, color modifiers, and quantity modifiers. Each of the nouns (N) refer to a simple geometric shape: "bap" = \square , "muq" = \triangle , "fid" = \circ . The three color modifiers (C) refer to colors ("duq" = orange, etc.). Colors are used as postfix operators on nouns (e.g. "muq duq" = \blacktriangle).

The three quantity modifiers (Q), each of which is polysemous, have the following meanings: "oy" = {small, one, light}, "op" = {large, many, dark}, "ez" = {not, none, non}. The specific meaning of a Q -modifier depends on context. As a prefix to an N (i.e. QN) a Q signifies the size of the noun, (e.g. "op muq" = \blacktriangle "a large triangle"). As a suffix to an N , (i.e. NQ), it signifies the cardinality of the N , (e.g. "muq oy" = \blacktriangle "one triangle"). As a suffix to a C , (i.e. CQ), it signifies the intensity or saturation of the C (e.g. "muq duq op" = \blacktriangle "a very orange triangle"). Multiple Q -modifiers can be used in a single phrase as in "op muq op ne f oy" = $\blacktriangle\blacktriangle\blacktriangle$, or "many large light-green triangles".

The skills in this domain are the abilities to construct or understand 14 legal syntactic forms of phrases up to length 5. That is, the skill set is

$$S = N, C, NC, CQ, NCQ, NQ, NQC, NQCQ, QN, \\ QNC, QNCQ, QNQ, QNQC, QNQCQ.$$

The *dependency structure* for this skill set links every skill s of length l to any skill of length $l - 1$ that is a substring of s .

2 The ITS and its Policies

We built an Intelligent Tutoring System (ITS) called BLAST to teach the language described in the previous section. BLAST presents students with *training problems* or *hints*. Training problems are multiple-choice items (always four choices) for which students have to match a scene (e.g., three triangles) to a sentence in the language. Sometimes the question is a scene and the multiple choice items are sentences, sometimes the items are scenes and the question is a sentence. Roughly 10% of BLAST's actions are hints.

BLAST presents one problem at a time, or it presents a menu of three problems and allows the student to choose which one to solve. These modes of presentation were never mixed for a student; that is, every student was in a *Choice* condition or a *NoChoice* condition. One of our research questions concerns the pedagogical value of choice.

BLAST selects problems (or hints) according to a *policy*. All policies are defined with respect to the skill set S introduced in the previous section.

Random and Expert Policies We constructed two *Random* policies to serve as control conditions. Each selects a problem type from S at random and presents either one problem (*NoChoice*) or three problems (*Choice*) of that type to the student.

Three *Expert* policies were constructed. Each relies on the dependencies between skills in S and on the student's estimated *mastery* of these skills. Briefly, the expert policies don't introduce "the next skill" until the student's problem-solving performance suggests that prerequisite skills have been mastered. The *ExpertNoChoice* policy selects a single problem and presents it to the student. The *ExpertChoice* policy selects three problems at the same skill level and lets the student decide which to solve. The *ExpertChoiceZPD* policy selects two problems at the "current" skill level and one at a higher skill level (although this is not indicated to the student) and lets the student decide which to solve.

Learned Policy After training 75 students with BLAST using the five policies just described, we applied a machine learning algorithm to learn a policy. The algorithm is described in detail in (Green, Walsh, Cohen and Chang, 2011). Very briefly, the algorithm learns which skill to present next given the student's mastery of other skills. The state of the student is encoded as a Dynamic Bayes' Network (Dean and Kanazawa, 1989), so the learned policy is called the *DBN* policy. Whereas the expert policies hard-code the order in which skills should be presented, the *DBN* policy orders skills to optimize skill mastery. Similar ideas have been proposed by (Almond, 2007; Barnes and Stamper, 2008; Beal and Qu, 2007; Beck, Woolf and Beal, 2000).

3 Experiment Protocol

Participants spent roughly one hour working with BLAST. On average, students were able to solve 133 multiple choice *training problems* during this time. Students were given the correct answer after each training problem. A single test, comprising 20 multiple choice problems that tested most of the skills in S , was administered to each student at four points during the hour, one very close to the beginning of the hour, one close to the end, and the others equally spaced between the first and the fourth. Students received no feedback on their answers to test questions. We refer to the first test as the *pretest*, the last as the *posttest*. Each test got a fractional score between 0/20 and 20/20. We define *improvement* to be the difference between the posttest score and the pretest score.

As noted, 75 students (50 men and 25 women) participated in five experimental conditions to collect training data for the *DBN* policy. All were recruited from the Psychology pool at the University of Arizona. An additional 35 students were recruited and distributed between the *DBN* and other expert conditions.

4 Results

Overall, students improve between the pretest and the posttest in the *Expert* and *DBN* conditions but not in the *Random* conditions. Mean scores on each of the four tests are plotted by condition in Figure 1. In the *Random* conditions, students' performance on the tests hovers around chance (0.25) and does not improve. This result establishes that the policy for presenting problems matters: if it is a random policy, students don't learn; if it is an expert or *DBN* policy, they do.

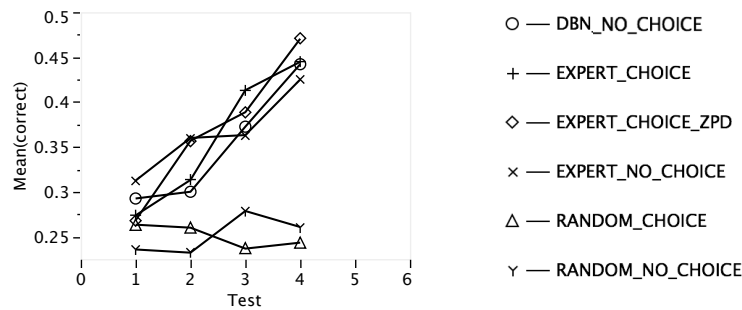


Fig. 1. Mean score on each of four tests by condition.

Comparing DBN with Other Expert Policies The amount and rate at which students learn in the *Expert* and *DBN* conditions do not seem very different. Students in all of these conditions start at roughly chance performance on the first test and are able to answer roughly 45% of the questions correctly on the final test. A two-way analysis of variance comparing *DBN* with the other *Expert* policies crossed with test number shows a main effect of test number ($p < .0001$) and no main effect of *DBN* or an interaction effect. That is, students improve from one test to the next but whether they work under the *DBN* policy or any other expert policy makes no difference to their improvement.

So is the *DBN* policy as good as the other expert policies? This is not a hypothesis testing question, as hypothesis testing can only show that the policies are unlikely to be equal. It is, however, a confidence interval question: We will show that the confidence interval around the difference between the policies is small and contains zero.

Let $\chi_{s,q,t} = [0, 1]$ represent whether student s answered question q on test t correctly (1) or incorrectly (0). Let $\pi_{\bullet,q,t} = (\sum_s \chi_{s,q,t})/N$ be the mean number of correct answers, averaged over N students, for question q on test t , and let $\iota_q = \pi_{\bullet,q,3} - \pi_{\bullet,q,0}$ be the mean *improvement* on question q between the first test (test 0) and the last (test 3). The value of ι_q is quite variable because it is harder to improve on some test questions than on others. The average value of ι_q for *DBN* students is 15% and for the other expert policies is 16%. The confidence interval around ι_q is $[-0.06, 0.09]$. This means that, by question, the difference in improvement under the *DBN* and other expert policies ranges from -6% to 9% with 95% confidence. In fact, by comparing mean squares in a two-way analysis of variance, we estimate that the test questions themselves have roughly ten times the influence on the width of the confidence interval around ι_q than the policy.

In sum, the *DBN* policy is indistinguishable from the other expert policies, at least with respect to improvement from the pretest to the posttest. This result establishes that a problem-oriented ITS, which repeatedly decides which skill to present to a student, can learn how to teach.

Differences due to Choice and Gender *Choice* and *gender* interact in unexpected ways to influence how much students learn. Table 1 shows the mean improvement by condition and gender. Overall, women improve significantly less than men ($p < 0.0045$,

two-tailed t test), and women do better in the two *Choice* conditions than in the two *NoChoice* conditions ($p < 0.052$, two-tailed t test). A two-way analysis of variance with *Choice* and *Gender* as factors shows a strong main effect of *Gender* ($p < 0.003$), no effect of *Choice*, and a marginal interaction effect ($p < 0.18$).

	Female	Male	Mean
DBN	0.046	0.328	0.15
Expert No Choice	0.044	0.19	0.125
Expert Choice	0.145	0.20	0.171
Expert Choice ZPD	0.155	0.25	0.202
Men	0.099	0.236	

Table 1. Mean improvement by condition and gender

These results show that there is not a simple answer to the questions “How does choice affect learning?” and, “How does the DBN strategy perform, relative to the other policies?” Women are clearly at a disadvantage with the *NoChoice* policies, especially the *DBN* policy. Looking at men, only, there is little effect of *Choice*, and the *DBN* policy outperforms the others (though the difference is not statistically significant, due to the small samples). For women, choice matters, though the *DBN* policy is no better or worse than the *Expert-NoChoice* policy.

We can only speculate about the reasons for these gender differences, and our explanations are post-hoc and await prospective experiments to test them. We built regression models to predict improvement and found that factors in these models had different effects depending on gender. The five factors in these models were:

- TotalTime* The total number of seconds spent in the experiment
- NumProblems* The number of training problems done by a student
- Test0Score* The score on the first test
- Success* The fraction of training problems solved correctly
- Test3Time* The number of seconds spent on test 3

We built saturated regression models for men and women, then selectively deleted factors, observing the resulting changes in R^2 , the percentage of variance in improvement accounted for by the models. The saturated models had R^2 of 41% and 54% for women and men, respectively. For men, removing *NumProblems* reduced R^2 by one half, to 27.6%, but for women, removing *NumProblems* had a negligible effect ($R^2 = 40\%$). For men, removing *Test0Score* had a small effect ($R^2 = 46\%$), but for women, removing this factor had a large effect ($R^2 = 27\%$). Removing *Success* had a large effect for women and men, and removing *TotalTime* had essentially no effect at all. Apparently, where a woman starts out (*Test0Score*) influences her improvement but the number of problems she solves has little effect, whereas the opposite is true of men.

The story is very similar when we build models to predict *Success* instead of improvement. For women, the most important factor is the *Test0Score* and *NumProblems* has little effect, whereas this pattern is reversed for men.

Yet, we are not quite ready to say that our tutoring policies do not work for women. One intriguing result is that the *Choice* policies on which women did the best are also

those on which all students, men and women, spent the most time per problem. Perhaps women fared poorly in the *NoChoice* conditions because they felt rushed – women and men both did 138 problems in the *NoChoice* conditions compared with 113 and 105 problems, respectively, in the *Choice* conditions. A three-way analysis of variance with factors *Choice*, *Gender*, and whether the student solved more than 110 training problems yields a marginal three-way interaction effect ($p < 0.06$), suggesting that all three factors contribute to the story of how a tutoring system helps students learn.

5 Discussion

The promise of Intelligent Tutoring Systems (ITSs) is that they will adapt and modify themselves, learning from data provided by previous students, always improving the quality of individualized instruction they give to each student. To the best of our knowledge, the current study is the first to show that ITSs can learn to do the right thing for some students and the wrong thing for others. Perhaps it is not a coincidence that the *DBN* policy, which was trained with data from 50 men and 25 women, helped men more than women. And while it is true that women improved less under all policies than men, this difference was amplified, not diminished, by the learned policy. Our next experiments will prospectively train separate policies on data for women and men, and then test their efficacy in two conditions: “Like” policies, where men and women learn under policies trained on data from their own sex, and “unlike” policies, where women learn on policies trained by men and vice versa. We look forward to a time when ITSs provide high quality, individualized instruction to every student, but now we know that this will not be an inevitable outcome of ITSs learning policies.

References

- Almond, R. G. 2007. Cognitive modeling to represent growth (learning) using markov decision processes. *Technology, Instruction, Cognition and Learning* 5:313–324.
- Barnes, T., and Stamper, J. 2008. Toward automatic hint generation for logic proof tutoring using historical student data. In *ITS*, 373–382.
- Beal, C. R., Arroyo, I., Cohen, P. R., Woolf, B. P. 2010. Evaluation of AnimalWatch: An intelligent tutoring system for arithmetic and fractions. *Journal of Interactive Online Learning*, 9, 65-77.
- Beal, C. R., and Qu, L. 2007. Relating machine estimates of students’ learning goals to learning outcomes: A DBN approach. In R. Luckin, K. R. Koedinger, J. Greer (Eds.), *Artificial intelligence in education: Building technology rich learning contexts that work* (pp. 111-118). Amsterdam: IOS Press.
- Beck, J. E., Woolf, B. P., Beal, C. R. 2000. to teach: A machine learning architecture for intelligent tutor construction. Proceedings of the American Association of Artificial Intelligence 17th National Conference, Austin TX.
- Dean, T., and Kanazawa, K. 1989. A model for reasoning about persistence and causation. *Computational intelligence* 5:142–150.
- Green, D. T., Walsh, T.J., Cohen, P.R., and Chang, Y. 2011. Learning a Skill-Teaching Curriculum with Dynamic Bayes Nets Technical Report, Department of Computer Science, University of Arizona
- Woolf, B. P. 2008. Building intelligent interactive tutors. Morgan Kaufmann, 2008