

Evaluating Adaptive Generation of Problems in Programming Tutors – Two Studies

Amruth Kumar

Ramapo College of New Jersey,
Mahwah, NJ 07430, USA
amruth@ramapo.edu

Abstract. We have developed an associative mechanism for adaptive generation of problems in tutors. We evaluated the adaptation using both within-subjects and between-subjects design. In within-subjects evaluation, instead of comparing control and test groups of students, we compared control and test groups of student-concepts: i.e., concepts on which students received practice against those on which they did not receive practice due to limited duration of the practice. We found that adaptation targets the concepts less well understood by students. In between-subjects evaluation, we compared an adaptive tutor against a non-adaptive version, based on the premise that exclusionary adaptation should be compared against the worst-case (all-inclusive case) and inclusionary adaptation should be compared against the best-case (all-exclusive case). We found that students who use the adaptive version learn with fewer problems. We have proposed “gain” of adaptation as the percentage decrease/increase that results from exclusionary/inclusionary adaptation respectively.

1 Introduction

Sequencing is an integral part of tutors. A typical tutor might sequence the topics, or the tasks within a topic. Given problem-solving task, classic tutors sequence the problems based on the domain model [11]. Adaptation has been attempted for various tasks in a tutor, including navigation, feedback and content presentation. The two mechanisms that have been attempted for adaptive problem-sequencing are vector spaces [13] and learning spaces [6].

We have proposed an associative mechanism for adaptive generation of problems in tutors [7]. In this approach, we index problems by concepts. We specify proficiency criteria for each concept in the domain model and maintain the student model as an overlay of the domain model. We use the round-robin algorithm to select the next concept the student has not yet understood and the next problem to be presented on that concept.

Associative adaptation is domain-independent, easier to build and scalable. Unlike vector spaces [13] and learning spaces [6], there is no need to exhaustively enumerate and organize all the problem templates in associative adaptation. New concepts and problem templates can be added to the tutor without affecting any existing template

and/or modifying the previously constructed vector/learning space. This feature permits incremental development of tutors, which is invaluable when developing tutors for large domains.

We have evaluated associative adaptation using both within-subjects and between-subjects designs. In within-subjects evaluation, we compared the concepts on which students received practice against those on which they did not receive practice. In between-subjects evaluation, we compared an adaptive tutor against a non-adaptive version. We will describe both these approaches in this paper, and list the results of evaluation.

2 The Context - Programming Tutors

We have been developing web-based tutors to help students learn C/C++/Java/C# programming language concepts by solving problems. To date, we have developed tutors on expression evaluation, selection statements, loops, pointers in C++, parameter passing mechanisms, scope concepts and their implementation, and C++ classes. Our tutors target program analysis (solving expressions, predicting the output of programs and debugging programs) in Bloom's taxonomy [3] in contrast to program synthesis (writing a program), which has been the traditional focus of intelligent tutors (e.g., LISP Tutor [12], ELM-ART [14]).

Limited problem set has been recognized as a potential drawback of encoding a finite number of problems into a tutor [9]. Therefore, our tutors generate problems as instances of parameterized templates. Each problem template is indexed by one or more concepts, and the template is used to generate problems for only these concepts.

During a typical tutoring session, a student steps through the following stages:

- **Pre-test** – The pre-test consists of a predetermined sequence of problems covering selected concepts. The pre-test is of fixed duration. The tutor administers the pre-test, but does not provide any feedback during the test.
- **Practice** – The tutor adapts the practice session to the needs of the learner, i.e., it generates problems on only those concepts that the student does not already know, as demonstrated by his/her performance on the pretest. The tutor provides detailed feedback for each problem, which includes explanation of the step-by-step execution of the program [8]. The practice session lasts a fixed duration of time or until the student demonstrates proficiency on all the concepts, whichever comes first. Since the tutor is capable of generating problems as instances of parameterized templates ad-infinitum, it never runs out of problems during practice.
- **Post-test** – The post-test consists of a pre-determined sequence of problems, covering concepts in the same order as the pre-test. The post-test is of fixed duration. The tutor administers the post-test, but does not provide any feedback during the test.

The three stages: pre-test, practice and post-test are administered by the tutor back-to-back, with no break in between.

Numerous evaluations have shown that our tutors help students learn, e.g., in one controlled test comparing a tutor with a printed workbook, improvement in learning with the tutor was larger and statistically significant compared to improvement with the printed

workbook [5]. Evaluations have also shown that the explanation of step-by-step execution provided as feedback by our tutors is the key to the improvement in learning [8]. We wanted to evaluate whether associative adaptation of problem generation helped improve the effectiveness of the tutors. The hypotheses for our evaluations were:

1. Associative adaptation targets the concepts less well understood by students.
2. Associative adaptation helps students learn with fewer problems.

Next, we will describe two mechanisms that we used to evaluate these hypotheses in our adaptive tutor.

3 A Within-Subjects Evaluation of the Adaptive Tutor

We propose that if the time allowed for practice with a tutor is limited, instead of categorizing students as control and test groups, we can categorize student-concepts as control and test groups, and compare the two groups to determine whether the adaptation addresses the needs of the student.

In spring and fall 2005, we evaluated our tutor on selection statements. We used the standard pre-test-practice-post-test protocol as described earlier:

- **Pre-test** – The pre-test consisted of a predetermined sequence of 21 problems covering 12 concepts. Students were allowed 8 minutes for the pre-test. The tutor used the pre-test to initialize the student model, as proposed by earlier researchers (e.g., [1,4]). However, the test was not adaptive as proposed by others (e.g., [2, 10]), because we wanted to compare the pre-test score with the score on a similarly constructed post-test to evaluate the effectiveness of the adaptive tutor.
- **Practice** – The practice session lasted 15 minutes or until the student learned all the concepts, whichever came first.
- **Post-test** – The post-test consisted of 21 problems, covering concepts in the same order as the pre-test. Students were allowed 8 minutes for the post-test.

Since the practice provided by the adaptive tutor was limited to 15 minutes, students did not always get practice on all the concepts on which they had not demonstrated proficiency during the pre-test. Therefore, we analyzed the data by student-concepts instead of students or problems. For each student, and each concept, we calculated the problems solved and average score on the pre-test, practice and post-test. Next, we grouped the concepts into four categories:

- **Discarded Concepts:** Concepts on which the student did not attempt any problem during the pre-test or during the post-test because of the time limit on the tests;
- **Known Concepts:** Concepts on which the student demonstrated mastery during the pre-test. The student could demonstrate mastery by solving at least 2 problems on the pre-test and scoring at least 60% on the problems;
- **Control Concepts:** Concepts on which the student solved problems during the pre-test and the post-test, but did not demonstrate mastery during the pre-test and *did not solve any problems during practice* due to the time limit imposed on the practice session – this provided the datum for comparison of test data.
- **Test Concepts:** Concepts on which the student solved problems during the pre-test and the post-test, but did not demonstrate mastery during the pre-test and *did solve problems during practice* – since the tutor provides feedback during practice to help

the student learn, data on test concepts could prove or refute the effectiveness of using the tutor for learning.

Table 1: Classifying student concepts as discarded, known, control or test.

Problems Solved	Pre-Test	Practice	Post-Test
Discarded	0	*	*
Discarded	*	*	0
Known	$A \geq M_1 \ \& \ R / A \geq M_2$	*	*
Control	+	0	+
Test	+	+	+

The four types of student concepts are summarized in Table 1, where * represents 0 or more problems solved, and + represents 1 or more problems solved. For our analysis, we ignored the discarded student-concepts since they represented incomplete data. We ignored the known student-concepts – the tutor cannot be credited for the learning of the concepts that the students already knew during the pre-test. On the remaining student-concepts, since each student served as part of both control group (on concepts for which the student did not get practice) and test group (on concepts for which the student did get practice), we consider this a within-subjects design, based not on the students but rather on the student-concepts. Since the control group of student-concepts represents no practice with the tutor, we cannot use this design to evaluate the effectiveness of adaptation in promoting learning. But, as we will discuss next, we can use this design to evaluate the effectiveness of adaptation in targeting the concepts less well understood by the students.

In Table 2, we have listed the average and standard deviation of the number of problems solved (listed as “Prob.”) and the average score per problem (listed as “Ave.”) on the pre-test, practice and post-test for the 56 control student-concepts and the 135 test student-concepts as defined above. The tutor presented at least two problems on each concept during the pre-test and post-test, and was capable of generating problems on a concept ad-infinitum during practice. The maximum possible value of average was 1.0. Therefore, no ceiling effect was observed.

Note that the average score of the control group remained steady whereas the average score of the test group increased by 48% (from 0.46 to 0.68) and this increase was statistically significant (2-tailed t-test $p < 0.05$). This attests to the effectiveness of the feedback provided by the tutor. A repeated measures one-way ANOVA on the average score, with the treatment (adaptive practice versus no practice) as between-subjects factor and pretest-post-test as the repeated measure showed that there was a significant interaction between the treatment (adaptive practice versus no practice) and time repeated measure [$F(1,189) = 10.211, p = 0.002$]: while the average score with no practice remained the same, with adaptive practice, it showed a significant increase.

What about the effectiveness of adaptation? Note that there is a statistically significant difference between the control and test groups on the number of problems solved and the average score on the pre-test. The average score of the test group of student concepts is significantly lower than that of the control group of student concepts. This can be interpreted to mean that our adaptation targeted the concepts less well understood by the students.

Table 2: Control versus Test Student-Concepts from spring 2005 evaluation of Selection Tutor

Spring 05 Selection	Pre-Test		Practice	Post-Test		<i>p</i> -value	
	Prob.	Ave.	Problems	Prob.	Ave.	Prob.	Ave.
Control (N = 56 student-concepts)							
Average	1.02	0.88	0	1.11	0.87	0.02	0.68
Std-Dev	0.13	0.30	0	0.31	0.31		
Test (N = 135 student-concepts)							
Average	1.07	0.46	1.83	1.35	0.68	0.00	0.00
Std-Dev	0.26	0.47	1.14	0.48	0.43		
<i>p</i>-value	0.05	0.00		0.000	0.00		

We repeated our evaluation of the tutor in fall 2005. When we analyzed the data by student-concepts instead of problems, and divided the set of student-concepts into control and test groups as described earlier, we obtained the results in Table 3. Once again, no ceiling effect was observed. On control student-concepts, the average changed from 0.81 to 0.76 from the pre-test to the post-test, and the change was not statistically significant ($p = 0.55$). On test student-concepts, the average changed from 0.61 to 0.86, and the change was statistically significant ($p < 0.05$). This again attests to the effectiveness of the feedback provided by the tutor, and the result was further borne out by ANOVA analysis.

Table 3: Control versus Test Student Concepts from fall 2005 evaluation of Selection Tutor

Fall 05 Selection	Pre-Test		Practice	Post-Test		<i>p</i> -value	
	Prob.	Ave.	Problems	Prob.	Ave.	Prob.	Ave.
Control (N = 26 student-concepts)							
Average	1.15	0.81	0	1.46	0.76	0.002	0.55
Std-Dev	0.37	0.35	0	0.51	0.40		
Test (N = 87 student-concepts)							
Average	1.00	0.61	1.55	1.15	0.86	0.000	0.00
Std-Dev	0	0.47	1.20	0.36	0.31		
<i>p</i>-value	0.04	0.02		0.006	0.23		

Once again, note that there is a statistically significant difference between the control and test groups on the number of problems solved and the average score on the pre-test. The average score of the test group of student concepts is significantly lower than that of the control group of student concepts. This once again supports our hypothesis that our associative adaptation targets the concepts less well understood by students.

4 A Between-Subjects Evaluation of the Adaptive Tutor

Problem adaptation in our tutors is *exclusionary*, i.e., adaptation results in *excluding* unnecessary problems. In other instances, adaptation may be *inclusionary*, e.g., adaptation may result in *including* additional feedback to be provided to a student. We propose that *exclusionary adaptation should be evaluated by comparing it against*

the worst-case (all-inclusive case) and inclusionary adaptation should be evaluated against the best-case (all-exclusive case).

An adaptive problem-generation tutor differs from a non-adaptive tutor in the following respects:

1. The concepts for which it generates problems – whereas an adaptive tutor generates problems on only the concepts the student has not yet mastered, the non-adaptive tutor generates problems on all the concepts, whether they have been mastered or not;
2. The number of problems it generates back-to-back on each concept – whereas an adaptive tutor can stop generating problems on a concept as soon as evidence is obtained of the mastery of a concept by the student, a non-adaptive tutor would generate a pre-determined number of problems on the concept before moving on to the next concept. We refer to this predetermined number of problems p as persistence [7]. $p = 1$ may not reinforce learning due to rapid switching of concepts. $p > 3$ may make the tutor predictable and boring.

Clearly, adaptive problem-generation is *exclusionary* in nature.

In the worst case, i.e., when the learner does not know any concept and cannot solve any problem correctly, a tutor will present the learner with problems on *all* the concepts, and present p problems on each concept. We evaluated the sequence of problems generated adaptively in our tutor against such a worst-case sequence of problems. When compared against such a worst-case, any exclusion of problems by an adaptive tutor would be a reflection of how much better-prepared a student is compared to the student in the worst-case scenario.

We conducted two between-subjects evaluations of the adaptation in our tutors in spring 2005, using our tutors on arithmetic expressions and relational expressions. We used the pre-test-practice-post-test protocol described earlier:

- **Pretest:** The pre-test consisted of a predetermined sequence of 21 problems covering 17 concepts in arithmetic expressions tutor and 17 problems covering 17 concepts in relational expressions tutor. Students were allowed 7 minutes for the pre-test.
- **Practice:** The practice session lasted 15 minutes. During practice, the test group used the adaptive version of the tutor, whereas the control group used a non-adaptive version of the tutor:
 - The non-adaptive tutor generated problems for all the concepts in a predetermined sequence, and generated p problems for each concept before going on to the next concept ($p = 2$ for arithmetic expressions tutor and $p = 1$ for relational expressions tutor). It repeated concepts and problems for each concept in a round-robin fashion until the expiration of the time allotted for practice.
 - The adaptive tutor generated problems on only the concepts the student had not yet mastered, and generated *up to* p problems on a concept before going on to the next concept, value of p being the same as for the non-adaptive version of the tutor. If the student happened to demonstrate proficiency on a concept with fewer than p problems, the adaptive tutor skipped the remaining problems on the concept and proceeded to the next concept immediately.

In the worst case, i.e., if the student scored zeroes on all the problems on all the concepts in the pre-test, and continued to solve every problem incorrectly during the prac-

tice, the sequence of problems generated by the adaptive tutor would be identical to the sequence of problems generated by the non-adaptive tutor.

- **Post-Test:** The post-test once again consisted of a predetermined sequence of 21 problems covering 17 concepts in arithmetic expressions tutor and 17 problems covering 17 concepts in relational expressions tutor. Students were allowed 7 minutes for the post-test.

All the students whose last name started with A-K constituted the control group for the arithmetic expressions tutor evaluation and test group for the relational expressions tutor evaluation. All the students whose last name started with L-Z constituted the test group for the arithmetic expressions tutor evaluation and the control group for the relational expressions tutor evaluation.

We have listed the results of evaluating the arithmetic expressions tutor in spring 2005 in Table 4. The tutor presented 21 problems during pre-test and post-test, and was capable of generating problems ad-infinitum during practice. The maximum possible value of average was 1.0. Therefore, no ceiling effect was observed. Note that the pre-post improvement was statistically significant for both the control and test groups. The pre-test and post-test averages for the two groups are very similar, and any difference between the two groups is not statistically significant. However, the test group solved 33% fewer problems (25.37 instead of 37.43) during practice than the control group, and this difference is statistically significant. Since the control group got the worst case sequence of problems, students with the adaptive tutor solved 33% fewer problems than the worst case of a student who does not know any concept and solves all the problems incorrectly. *We propose that this decrease is the “gain” of the adaptation – the percentage of problems excluded by the tutor due to the better-preparedness of the students. In inclusionary adaptation, the gain is the percentage increase that results from adaptation.*

Table 4: Control vs Test Groups from spring 2005 evaluation of Arithmetic Expressions Tutor

Spring 05 Arithmetic	Pre-Test		Practice	Post-Test		p-value	
	Prob.	Ave.	Problems	Prob.	Ave.	Prob.	Ave.
Control (N = 21 students)							
Average	10.48	0.46	37.43	14.10	0.61	0.0001	0.0066
Std-Dev	4.21	0.25	19.10	4.95	0.24		
Test (N = 35 students)							
Average	10.11	0.47	25.37	14.23	0.59	0.0000	0.0004
Std-Dev	4.63	0.27	19.39	5.93	0.28		
p-value	0.771	0.927	0.027	0.931	0.851		

We have listed the results of evaluating the relational expressions tutor in spring 2005 in Table 5. The tutor presented 17 problems during pre-test and post-test, and was capable of generating problems ad-infinitum during practice. The maximum possible value of average was 1.0. Therefore, no ceiling effect was observed. Note that the pre-post improvement in average score was statistically significant for both the control and test groups. The pre-test and post-test averages for the two groups are very similar, and any difference between the two groups is not statistically significant. However, the test group solved 69% fewer problems (14.13 instead of 45.05) during practice than the control

group, and this difference is statistically significant. In other words, the gain of the tutor is 69%. The gain of relational expressions tutor is larger than that of the arithmetic expressions tutor presumably because relational expressions are easier than arithmetic expressions. So, students needed fewer practice problems to learn the concepts.

Table 5: Control vs Test Groups from spring 2005 evaluation of Relational Expressions Tutor

Spring 05 Relational	Pre-Test		Practice	Post-Test		p-value	
	Prob.	Ave.	Problems	Prob.	Ave.	Prob.	Ave.
Control (N = 21 students)							
Average	13.90	0.68	45.05	15.00	0.77	0.256	0.020
Std-Dev	3.13	0.21	19.19	3.39	0.16		
Test (N = 16 students)							
Average	14.56	0.73	14.13	15.38	0.82	0.302	0.023
Std-Dev	3.44	0.22	16.12	2.58	0.16		
p-value	0.554	0.474	0.0000	0.705	0.355		

These two evaluations support our hypothesis that associative adaptation of problem sequence helps the student learn with fewer problems.

5 Discussion and Conclusions

We have developed an associative mechanism for adaptive generation of problems in tutors. We evaluated the adaptation using both between-subjects and within-subjects designs. Associative adaptation can be used in any domain by 1) identifying fine-grained concepts in the domain; 2) maintaining the overlay student model in terms of these concepts; and 3) using these concepts to index tutor activities.

In within-subjects evaluation, instead of comparing control and test groups of students, we compared control and test groups of student-concepts, i.e., we compared the concepts on which students received practice against those on which they did not receive practice due to limited duration of practice. We found that adaptation targets the concepts less well-understood by students.

Evaluating a tutor using control and test groups of student-concepts rather than students is a novelty of our work. During the design of the evaluation, student-concepts may be explicitly grouped into control and test groups, so that each student receives differential treatment on some concepts, but not others. The result is either a within-subjects or a partial cross-over design. In our evaluations, we divided student-concepts into control and test groups post-hoc, i.e., after the students had used the tutor, based on whether or not they received practice on each concept. This within-subjects design could be confounded by factors such as the following:

1. Fewer students may get adequate practice on concepts that appear late during the practice session. Therefore, the order in which problems are presented on concepts during the practice session will affect whether a student-concept ends up in the control or test group. Some concepts may be harder than others. Let us consider three scenarios:

- a. If the tutor covered concepts during practice session in increasing order of difficulty, control student-concepts would mostly consist of harder concepts. However, this is belied by the fact that the pre-test average on control student-concepts is significantly higher than that on test student-concepts.
 - b. If the tutor covered concepts during practice session in decreasing order of difficulty, control student-concepts would mostly consist of easier concepts. This would support our claim that adaptation targets the harder concepts.
 - c. If the tutor covered concepts during practice session in a random order that stochastically distributed the level of difficulty of the concepts, the average pre-test score of control and test groups would be comparable. However, this is not the case – in both tables 2 and 3, the difference between control and test group pre-test scores is statistically significant.
2. Fewer students may manage to solve problems on the concepts that appear late in the pre-test or post-test. Therefore, the order in which problems are presented on concepts during the pre-test and post-test will affect the scores of control and test groups on the tests. Each test contained two consecutive sequences of problems, each sequence consisting of one problem per concept. So, students may solve more problems on some concepts than on others. We took several measures to address these confounds in our analysis:
- a. The pre-test and post-test presented problems in the same order of concepts as the practice session.
 - b. We discarded all the student-concepts on which students did not solve any problems either during the pre-test or during the post-test. Therefore, if students consistently failed to solve problems on a concept because of its order on the pre-test/post-test, we simply discarded the concept during analysis. We compared the pre-post change in average score using only those concepts on which a student had solved problems during both pre-test and post-test.
 - c. Finally, we compared average score per problem, not raw score. This factored out the effect of different number of problems solved by students on the various concepts.
3. There is more room for students to improve on the harder concepts, i.e., test student-concepts than on the easier concepts, i.e., control student-concepts. This confounds a between-subjects comparison of the post-test scores of control and test concepts to demonstrate the effectiveness of the tutor. But, we can still use the statistically significant pre-post improvement on the test concepts to back up our claim that the tutor helps students learn even the hard concepts.

In between-subjects evaluation, we compared an adaptive tutor against a non-adaptive version. The premise for the comparison was that exclusionary adaptation should be compared against the worst-case (all-inclusive case) and inclusionary adaptation should be compared against the best-case (all-exclusive case). We found that students who use the adaptive version learn with fewer problems. We proposed “gain” of adaptation as the percentage decrease/increase that results from exclusionary/inclusionary adaptation respectively.

Currently, we are working on exclusionary adaptation of cognitive feedback and inclusionary adaptation of affective feedback in our tutors. We also continue to evaluate the adaptive generation of problems and its effect on student learning in our tutors.

Acknowledgements

Partial support for this work was provided by the National Science Foundation's Educational Innovation Program under grant CNS-0426021.

References

1. Aimeur, E., Brassard, G., Dufort, H., and Gambs, S. CLARISSE: A Machine Learning Tool to Initialize Student Models. S. Cerri, G. Gouarderes, F. Paraguacu (eds.), Proc. of ITS 2002, Springer (2002). 718-728.
2. Arroyo, I., Conejo, R., Guzman, E., & Woolf, B.P. An Adaptive Web-Based Component for Cognitive Ability Estimation., Proc. of AI-ED 2001, IOS Press (2001). 456-466.
3. Bloom, B.S. and Krathwohl, D.R.: Taxonomy of Educational Objectives: The Classification of Educational Goals. Handbook I: Cognitive Domain, New York, Longmans, Green (1956).
4. Czarkowski, M. and Kay, J. Challenges of Scrutable Adaptivity. U. Hoppe, F. Verdejo and J. Kay (eds.), Proc. of AI-ED 2003, IOS Press (2003). 404-406.
5. Dancik, G. and Kumar, A.N., A Tutor for Counter-Controlled Loop Concepts and Its Evaluation, Proceedings of Frontiers in Education Conference (FIE 2003), Boulder, CO, 11/5-8/2003, Session T3C.
6. Kurhila, J., Lattu, M., and Pietila, A. Using Vector Space Model in Adaptive Hypermedia for Learning. Proc. of ITS 2002, Springer (2002). 129-138.
7. Kumar, A.N. A Scalable Solution for Adaptive Problem Sequencing and its Evaluation. In Proceedings of The 2006 International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH 2006), Dublin, Ireland, June 21-23, 2006.
8. Kumar, A.N., Explanation of step-by-step execution as feedback for problems on program analysis, and its generation in model-based problem-solving tutors, Technology, Instruction, Cognition and Learning (TICL) Journal, to appear.
9. Martin, B. and Mitrovic, A. Tailoring Feedback by Correcting Student Answers. Proc. of ITS 2000. Springer (2000). 383-392.
10. Millan, E., Perez-de-la-Cruz, J.L., and Svazer, E. Adaptive Bayesian Networks for Multilevel Student Modeling. Proc. of ITS 2000. Springer (2000), 534-543.
11. Polson, M.C. and Richardson, J.J., (Eds.) 1998. Foundations of Intelligent Tutoring Systems. Hillsdale, NJ: Lawrence Erlbaum.
12. Reiser, B., Anderson, J. and Farrell, R.: Dynamic student modelling in an intelligent tutor for LISP programming, Proc. of IJCAI 1985. Los Altos CA (1985).
13. Salton, G., Wong, A. and Yang, C.S. A Vector Space Model for Automatic Indexing. Communications of the ACM, Vol. 18(11), (1975). 613-620.
14. Weber, G. and Brusilovsky, P. ELM-ART: An Adaptive Versatile System for Web-Based Instruction. International Journal of Artificial Intelligence in Education, Vol 12 (2001). 351-384.