A formative evaluation of AnimalWatch

Ivon Arroyo[1], Tom Murray[1,3], Joseph E. Beck[2], Beverly P. Woolf[1], Carole R. Beal[1]
[1]University of Massachusetts, Amherst - [2]Carnegie Mellon University - [3]Hampshire College

Abstract. We present the results of a deep formative evaluation of AnimalWatch, an intelligent tutor for arithmetic. Students learned with AnimalWatch, and had a positive experience with it. Still, we detected AnimalWatch selected too easy problems. We proposed different hypotheses for this behavior and analyzed the performance of each ITS component, by observing the interactions of hundreds of students with the system. We conclude with a report of lessons learned that would make ITS better.

Introduction

Evaluating Intelligent Tutoring Systems (ITS) is not an easy task due to their dynamic nature. Most quantitative ITS evaluations in the past consisted of pre to post test improvements (Koedinger, 97; Luckin, 99]. These evaluations are appropriate for ITS in experimental settings, but do not show how fast students learned, or how efficient the internal components of the ITS were. In this paper, we present the results of a large scale formative evaluation of AnimalWatch, an ITS for arithmetic (Beal, 02], by analyzing tens of thousands of interactions made by 350 students. We analyzed how much mathematics students had learned, and also how properly the ITS adapted to student's needs. These two goals imply different approaches. The first one implies evaluating whether students make fewer mistakes as time progresses, by focusing on how the *student responded to the system*. The second goal was to evaluate whether the adaptive mechanisms had functioned efficiently, by analyzing how properly the *system adapted to the student.*

1. Students' response to the system

When the student enters an incorrect answer, AnimalWatch provides immediate help, with increasing level of information as students continue to make mistakes. A full description of the different kinds of hints available can be found in (Arroyo, 03a; Arroyo, 03b]. This section analyzes students' mistake behavior in relation to the problems and the help provided.

We analyzed students' reaction to help by plotting overall trends of mistake change along the tutoring session. We analyze problem sequences for each student, i.e. the 1st, 2nd, .., 6th problem seen for a *specific topic and difficulty*. Figure 1 shows the average mistake sequences made for different topics and difficulties. These are averages over about 200 students for the first topics, and of about 50 students for the last topics, as only students who started the problem sequence by making mistakes were considered. After six problems, mistakes reduced about 30%, and 50% by the fifth problem of a similar difficulty. When knowing the gender and cognitive development of the student, specific choices of hints in the system would produce a larger improvement than this average [Arroyo, 03b]. In addition, the impact on students' mathematics attitudes was highly positive: student's mathematics self-confidence and liking of mathematics improved significantly [Arroyo, 03a]. At the same time, we found that students made no mistakes in 70% of the problems, and that most of the problems selected for students involved the first topics while saw few problems on the last topics (fractions). We decided to analyze how the system adapted to the student.

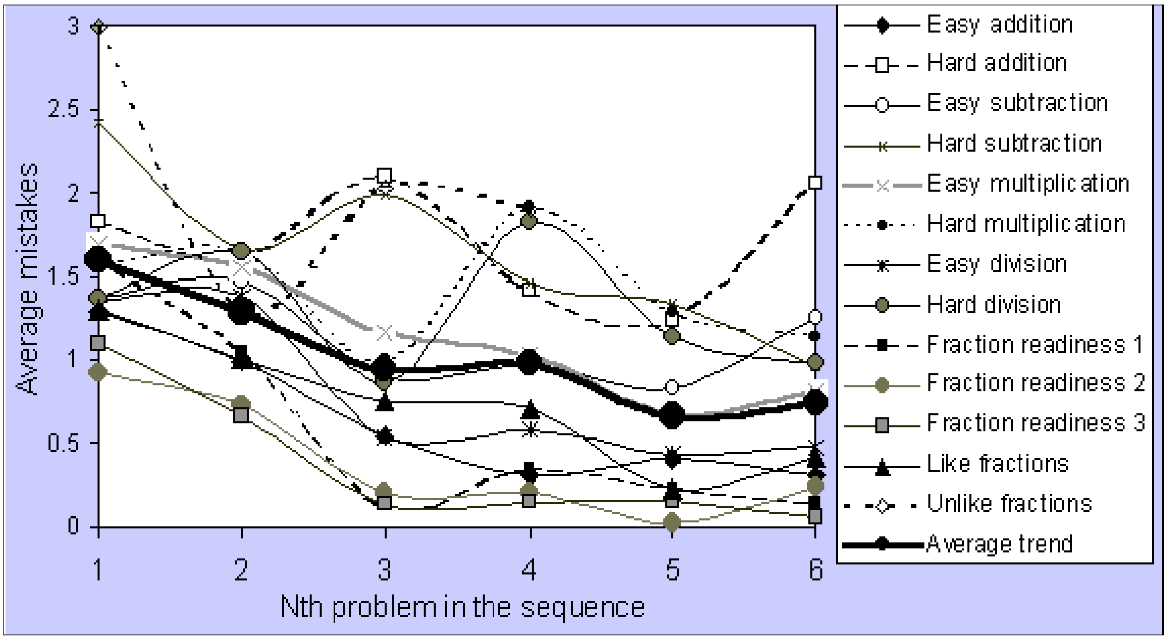

Figure 1. Learning curves for various topics and difficulties

2. System's response to the student

Efficient tutoring implies providing the student with educational activities that are within a zone where problems are not too hard and not too easy [Murray, 03]. We analyzed the pedagogical model's problem selection behavior and the student model's accuracy.

Pedagogical model. We think ITS should present problems where some mistakes will be made, where the difficulty of assigned problems is proportional to the knowledge of the student [Murray, 03]. We found the system had selected problems of a lower difficulty than we originally aimed for. We blame this effect partly to the system's reaction to lack of "ideal" problems. When the system did not find a problem of the difficulty it was looking for, it *relaxed* its constraints until it found a "reasonable" problem. In the end, the system tended to be very sensitive to the content available, and AnimalWatch had more problems of easier difficulty. We learned that the decisions of what to do when the ideal content is not present would be extremely relevant to the pedagogical model's efficiency. We also analyzed AnimalWatch's "review" problems rate. We compared the proportion of review vs. non-review problems. Students got a low percentage of review problems overall, as initially planned. However, the first topics had a higher chance of being reviewed, as they had been mastered for longer time (by the time the last topics were reached, about 40% of the problems seen for the first topic had been review problems). We conclude that the review rate should be "faded" for the earlier topics seen, as more topics get to be mastered.

Student model efficiency. One possibility for AnimalWatch giving too easy problems is that the student model was not accurate at estimating students' knowledge. We thus analyzed how the average mastery level changed for the problem sequences in figure 1. The result was an increasing trend for mastery level as mistakes reduce. Thus, the bayesian update mechanism was sensitive to mistake change. We observed, however, many instances of students with low proficiency levels, which made AnimalWatch give easy problems because it *believed* the student knew little. When plotting average problem difficulty for problem sequences per topic, we observed spikes of problem difficulty in the first five problems of each topic, as if it took the system a while to catch up with the students' actual knowledge level. The student model is thus partly to blame, as it initially assumed the student knew too little to begin with. Initializing the student model to the average level of the class should make the student model more accurate. One last hypothesis was that students were not behaving sincerely with the system. Students may enter wrong answers on purpose to get specific hints, not necessarily because they do not know the topic, but because they are being "sloppy". This will make the student model underestimate student's ability, as the system is very sensitive to student's responses. Actually, a large portion of students spent less than 4 seconds in between responses. This is important to address, as it will eventually generate an underestimation of students' ability. We propose to dynamically adjust the student model's *slip* parameters depending on response time.

3. Conclusions

AnimalWatch was a good system in many senses. Students seemed to learn when the system provided problems where students made mistakes. In addition, their attitudes towards math improved significantly after using the tutor, and students enjoyed working with it. At the same time, we detected that AnimalWatch was selecting "too easy" problems. We find the combination of these results interesting, as the impact on attitudes could have happened *due* to AnimalWatch providing problems that they managed to solve correctly, as students may have liked being successful problem solvers. We described why the pedagogical and student models are to blame for this "too easy" problem behavior. We conclude that how much students learn with an ITS will be affected by many factors, related to the efficiency of the internal components, and on how they interact with each other. The dysfunction of an ITS may not be detected until actual data of students using the system is analyzed, because it is originated not in the accuracy of the intelligent components themselves, but in how they react to unexpected behaviors in the actual environment. This makes formative evaluations important to improve ITS.

1 References

Arroyo, I. (2003a) Quantitative Evaluation of gender differences, cognitive development differences and software effectiveness for an Elementary Mathematics Intelligent Tutoring Systems. Doctoral dissertation. School of Education. University of Massachusetts Amherst.

Arroyo, I.; Beal, C.; Woolf, B; Murray, T. (2003b) Further results on gender and cognitive differences in help effectiveness. Proceedings of the 11th International Conference on Artificial Intelligence in Education.

Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. (1997). Intelligent tutoring goes to school in the big city. International Journal of Artificial Intelligence in Education, 8, 30-43.

Luckin, R. and du Boulay, B. (1999). Ecolab: The development and evaluation of a Vygotskian design framework. International Journal of Artificial Intelligence in Education, 10, 198-220.

Murray, T., & Arroyo, I. (2003). ITS Evaluation Using Process-Oriented ZPD Metrics . Proceedings of the 11th International Conference on Artificial Intelligence in Education.

Beal, C. R., & Arroyo, I. (2002). The AnimalWatch project: Creating an intelligent computer mathematics tutor. In S. Calvert, A. Jordan, & R. Cocking (Eds.), Children in the digital age. Greenwood.