Evaluating the Effectiveness of Integrating Natural Language Tutoring into an Existing Adaptive Learning System

Benjamin D. Nye, Alistair Windsor, Phillip Pavlik, Andrew Olney,
Mustafa Hajeer, Arthur C. Graesser, Xiangen Hu

Institute for Intelligent Systems, University of Memphis
Memphis, TN 38152
benjamin.nye@gmail.com

Abstract. This paper reports initial results of an evaluation for an ITS that follows service-oriented principles to integrate natural language tutoring into an existing adaptive learning system for mathematics. Self-explanation tutoring dialogs were used to talk students through step-by-step worked solutions to Algebra problems. These worked solutions presented an isomorphic problem to a preceding Algebra problem that the student could not solve in an adaptive learning system. Due to crossover issues between conditions, experimental versus control condition assignment did not show significant differences in learning gains. However, strong dose-dependent learning gains were observed that could not be otherwise explained by either initial mastery or time-on-task.

Keywords: Intelligent Tutoring Systems, Natural Language Tutoring, Mathematics Education, Worked Examples, Isomorphic Examples

1 Overview

Future intelligent tutoring systems (ITS) will need to integrate with other learning systems, particularly other intelligent systems. The **S**hareable **K**nowledge **O**bjects as **P**ortabl**e** **I**ntelligent **T**utors (SKOPE-IT) system was designed to integrate natural language tutoring dialogs into an existing learning environments. In this study, we combined the AutoTutor Conversation Engine (Nye et al., 2014) with the ALEKS (Assessment and Learning in Knowledge Spaces) commercial mathematics system (Falmagne et al., 2013).

AutoTutor and ALEKS have complementary strengths: AutoTutor focuses mainly on help during a problem (micro-adaptivity) and ALEKS focuses on macro-adaptivity, such as problem selection. Based on Knowledge Space Theory, students in ALEKS can only attempt a problem after mastering all of its prerequisites (Falmagne et al., 2013). Conversely, the AutoTutor Conversation Engine (ACE) directs conversations with one or more conversational agents (Nye et al., 2014). While ACE can be integrated with outer-loop models, each tutoring dialog adapts to the student's free-text input and other session events.

When integrating these systems, the goal was to combine worked examples (Schwonke et al., 2009), self-explanation (Aleven et al., 2004), and impasse-driven learning (VanLehn et al., 2003). The integration point between AutoTutor and ALEKS was the "Explain" page for ALEKS items. The ALEKS Explain page presents a worked solution to the specific problem that a learner could not solve. The SKOPE-IT system integrated AutoTutor dialogs by presenting a tutoring-enhanced worked solution for an isomorphic problem, with a series of small dialogs covering key principles. After each dialog finishes, more HTML for the worked example (including any images) is dynamically rendered until the next step-specific dialog is delivered. Figure 1 shows the first dialog of a tutored example, with most of the solution still hidden.

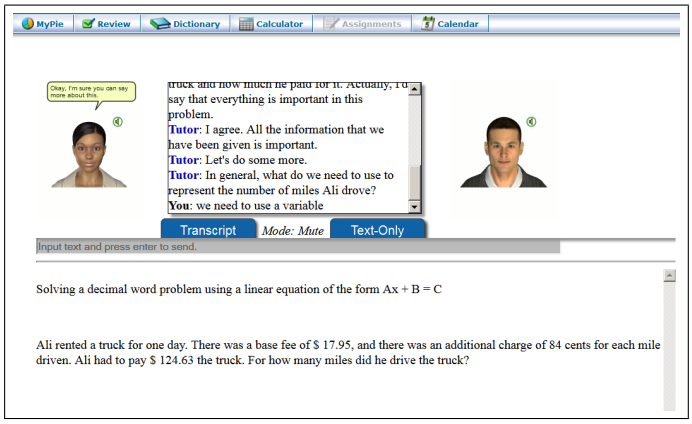

Fig. 1. Integration of a Tutored Worked Solution into ALEKS

2 Study Methodology and Population

50 ALEKS worked solutions drawn from items aligned to the Common Core were enhanced using AutoTutor dialogs. For each worked solution, 5 to 12 brief dialogs were authored (407 dialogs in total). Dialogs focused on Algebra concepts, such as representation mapping, systems of equations, or units of measurement. Two types of dialogs were authored: trialogs (75% of dialogs) and vicarious tutoring (25% of dialogs). In trialogs, the human student answered a conceptual question, with feedback and support from the tutor and peer student agents. In vicarious tutoring, the peer student agent modeled an explanation with the tutor.

Three sections of a mid-south college algebra class participated (112 students), which was a class for students with the lowest math placement scores. SKOPE-IT randomly assigned each student to an experimental condition with tutoring-enhanced items or to a control where ALEKS presented its usual non-interactive solutions. Unfortunately, due to a glitch in authentication, the control condition was presented with tutoring for 3 weeks out of the 12-week course, making the control condition a lower-dose treatment. The following analyses are based on the ALEKS course and AutoTutor interactions. ALEKS data included

course mastery levels from adaptive assessments and time spent in ALEKS. ALEKS assessments determined course grades, so students were presumably motivated. The SKOPE-IT system collected dialog interaction data of the student with the AutoTutor system (e.g., # of inputs, # of hints given).

3 Results

Results from ALEKS assessment scores are presented in Table 1, with standard deviations in parentheses. Due to random chance, the experimental condition contained less subjects at both the start ($N_{E,0}$) and end ($N_{E,f}$) of the study. The experimental subjects slightly outperformed the control (+3.3 points learning gain), but this difference was not statistically significant (Cohen's d=0.2, p=0.45). Attrition rates for both conditions were high (and are generally high for that course), but were not significantly different.

Condition	Initial Score (N_0=103)	Final Score (N_f=76)	Learning Gain (N_f=76)	Effect Size (N_f=76)
Experimental ($N_{E,0}$=42, $N_{E,f}$=28)	20.5 (5.5)	52.6 (18.9)	31.7 (19.4)	d=2.3
Control (N_{C0}=61, $N_{C,f}$=48)	23.2 (7.3)	51.8 (17.1)	28.4 (15.5)	d=2.1

Table 1. Assessment Outcomes by Assigned Condition

The dosage of AutoTutor interactions was a confound for comparing conditions. Since students took different paths through the ALEKS adaptive system, they encountered different numbers of tutoring dialogs (μ=24 and σ=27 among students with at least one dialog). Since each example had an average of 8 dialogs, students who received dialogs saw only about 3 worked examples out of 50. Also, due to crossover issues, the "experimental" subjects only averaged four more dialogs than the "control" subjects.

To look at dose-dependent effects, a linear regression used to model the learning gain as a function of the logarithm of the time spent in ALEKS and logarithm of the number of AutoTutor dialogs interacted with (Table 2). Logarithmic transforms were applied because diminishing learning efficiency was observed for a subset of students who overdosed on the combined system (7 students spent 80+ hours in ALEKS, σ=1.5 above the mean). The regression improved the model fit (R^2=0.54) when compared to a model with only time spent studying (R^2=0.49). Dialog dosage was significant even after accounting for time on task (including time on dialogs). Including a term for dialogs that the learner encountered but ignored (e.g., returned to problem solving instead) did not improve the model fit (t=-0.32, p=0.75).

Factor (N=76)	Coefficient	P-value
Log_{10}(# Hours in ALEKS)	43.0	<0.001 (t=6.6)
Log_{10}(# AutoTutor Dialogs Interacted With)	8.0	0.009 (t=2.7)
Intercept	-41.1	<0.001 (t=-4.2)

Table 2. Learning by Time and Tutoring Dialogs (R^2=0.54, R^2_{cv}=0.54)

4 Discussion and Conclusions

Due to insufficient differences in dosage, the main conditions showed no significant differences in learning gains. With that said, the dosage of tutoring dialogs was strongly associated with learning gains. Moreover, no other explanatory factor was found that captured this difference. Student prior knowledge did not correlate with dialog interaction (Pearon's R=-0.03). Also, dialogs were only associated with learning when the learner interacted with them, making it unlikely that higher-achieving students simply encountered more dialogs. Finally, regressions found that AutoTutor dialogs predicted learning even after accounting for all time studying in the combined system. However, the available data cannot fully eliminate the possibility that students prone to interact with dialogs also learned faster due to mutual causation. A follow-up study that controls for dialog time-on-task would be needed to address this question.

A second issue was that some learners reported that they did not understand the relevance of isomorphic examples. These comments expressed that explaining concepts on a similar problem does not show them "how to get the correct answer" for their earlier problem. This implies that these students may perceive that reaching the answer to the current problem is equivalent to learning. Prior research found that self-explanation can improve deep understanding, such as identifying when a problem cannot be solved (Aleven et al., 2004). Unfortunately, even though self-explanation can improve learning efficiency (Aleven et al., 2004), students may believe the opposite because they do not complete as many problems, which may be their internal barometer for learning. Such students may need interventions to convey the role of self-explanation and dialog in learning. Tutoring such metacognition may be a key future direction.

Acknowledgments This work was supported by the Office of Naval Research Grant N00014-12-C-0643. All views expressed are those of the authors alone.

References

Aleven, V., Ogan, A., Popescu, O.: Evaluating the effectiveness of a tutorial dialogue system for self-explanation. In: Intelligent Tutoring Systems (ITS) 2004. pp. 443–454. Springer (2004)

Falmagne, J.C., Albert, D., Doble, C., Eppstein, D., Hu, X. (eds.): Knowledge Spaces. Springer, Berlin, Germany (2013)

Nye, B.D., Graesser, A.C., Hu, X., Cai, Z.: AutoTutor in the cloud: A service-oriented paradigm for an interoperable natural-language its. Journal of Advanced Distributed Learning Technology 2(6), 35–48 (2014)

Schwonke, R., Renkl, A., Krieg, C., Wittwer, J., Aleven, V., Salden, R.: The worked-example effect: Not an artefact of lousy control conditions. Computers in Human Behavior 25(2, SI), 258–266 (2009)

VanLehn, K., Siler, S., Murray, C., Yamauchi, T., Baggett, W.B.: Why do only some events cause learning during human tutoring? Cognition and Instruction 21(3), 209–249 (2003)