Intelligent Systems Program Comprehensive Examination

Intelligent Tutoring Systems

Noboru Matsuda
March 14, 2003

Question from Kevin Ashley:

Write a short essay (no more than 6 pages) on teaching with examples in Intelligent Tutoring Systems. Address the following points.

What uses have intelligent tutoring systems made (or could they reasonably make) of teaching with examples? List as many different uses as you can think of and provide some brief but specific illustrations.

From a pedagogical or cognitive viewpoint, discuss what roles the examples appear to play in the interaction between the ITS and student. For instance, discuss what cognitive or pedagogical roles examples play in the ITS's process of teaching the abstract rules or principles of a domain. Do the students reason by analogy from the examples? If so, what kinds of analogies do/should they draw and how do the analogies relate to the pedagogical goals? Give an example or two. In your opinion, are there any connections between an ITS's teaching with examples and self-explanation? Explain why or why not.

What challenges are presented in designing and implementing ITSs to teach with examples in the various ways you described in your initial list? How could one design an ITS that uses examples to teach students how to apply abstract principles in solving problems? Discuss if and how one could design an ITS that uses examples to encourage students to "discover" some principles (say, of geometry or physics) on their own?

Answer:

Principle of versatile output says that "a tutor must have a wide varieties of instructional actions to provide students better learning" (Ohlsson, 1986). Use of examples are known to be effective hence must be added in a repertory of tutor's instructional strategy. We first discuss use of examples in ITS with an actual system developments. Next, we discuss theoretic background in learning from examples in cognitive studies. Finally, we discuss some challenging issues to design ITS that facilitate learning from examples.

1. Use of examples in ITS

There are several ways to use examples in intelligent tutoring systems as described in this essay. The bottom line is that the examples are provided to students as a model of problem solving. Namely, examples show how to apply domain principles (or, problem-solving rules or operators, if you will). The variations in the use of examples lie in the way that the tutor *displays* examples as well as the way that the students *read* examples.

The most straightforward way to display examples is to simply provide them to students as hints. Namely, when students cannot proceed problem solving, the ITS can simply show a solution for a similar problem. There is nothing fancy to display examples in this manner.

Examples can be decorated so that particular aspects of solution are highlighted. There are several different aspects that can be highlighted. For problem solving, a structure of solution (which is also called a *goal structure*) is one of the most important things to be recognized by students. Indeed, explication (also called "reification" in ITS literature) of a goal structures is know to be effective in cognitive studies on learning, including ITS studies (Koedinger & Anderson, 1993). The model tracing tutors can provide a structured figure of solution (i.e., an example), because individual domain principles can be implemented independent of other rules, called production rules (Anderson, Corbett, Koedinger, & Pelletier, 1995). In CATO, an ITS to support learning argumentation skills, explicate "rhetorical recipe" that is a general structure of argument with cases (Aleven & Ashley, 1997). CATO can generate an example of argument with annotation from the rhetorical recipe, and let students compare their argument with the example argument.

Problem solving strategy (i.e., forward chaining, backward chaining, and back-up) can be explicated in an example. Even when plenty of example solutions are available to students, it is still questionable whether the students can read the problem-solving strategy off the examples. It might be needed to explicate how to build up a solution, namely, the process of problem solving, as opposed to a structure of solution as a final product of problem solving. ANGLE, an ITS for geometry theorem proving, reifies not only domain principles (implemented as “diagrammatic configuration schema”) but also the search process including schema selection, schema application, and subgoaling (Koedinger & Anderson, 1993).

Besides problem solving, examples can be used to facilitate comprehension of complex concepts. Human teachers often say “for example” to clarify complex and abstract concepts. No assigned readings provide any instance of the use of example in this manner. Technically, we need a mechanism to generate an instance of example from domain principles.

Now, for the twist on the use of examples on the student’s side, there are several possibilities for students to read examples more effectively. First, the students can self-explain examples. SE-COACH, a physics tutor, provides students with an opportunity to explain steps in an example solution (Conati & VanLehn, 2000). The geometry explanation tutor asks students to justify their assertions (Aleven, Popescu, & Koedinger, 2001). The latter does not provides examples (but only problems), but it must be a natural extension to modify the geometry explanation tutor to provide a solution and just ask students to justify solution steps like SE-COACH.

Second, the students would benefit by manipulating examples. No instance of such learning environment can be found in the readings, but by chaining values used in the examples, for example, students might see more insight into application of domain principles.

How about to ask students to make examples by themselves? I have not ever read a paper on this issue, but it would be an interesting research. ITS would ask students to provide an application of domain principle(s). ITS then critiques examples generated by students.

2. Cognitive theories of learning from examples

This issue is not covered by the readings in “Intelligent Tutoring Systems,” but do covered by the readings in one of two other topics of interest, namely, “Human Problem Solving and Skill Acquisition.” Hence I refer some studies from readings in that topic. Some of the cited studies, which I did read in the past, come even outside of the whole reading lists assigned for the current comprehensive examination. Appendex shows which paper came from outside of the reading list.

Learning from worked examples appears to be effective especially for novice students. The theory of cognitive load explains that the reason of effectiveness in learning from example is that it requires students less cognitive load (Kalyuga, Chandler, Tuovinen, & Sweller, 2001).

2.1. Use of analogy and learning from examples

In cognitive science, *analogical thinking* is commonly defined as a transfer of problem solving from one situation to another by a process of *mapping* – finding a set of one-to-one correspondences between aspects of one body of information and aspects of another (Gick and Holyoak (1983, p.2) cited in Caplan & Schooler, 1999, p.42). Hence examples must provide students with an opportunity to learn features to be mapped to activate analogical thinking.

So, for the students to use abstract rules (and principles) to solve novel problems by analogy, they must learn not only how to apply rules in the given examples but also underlying aspects of rule applications appeared in examples.

Several studies show that learning by examples would facilitate transfer (i.e., acquisition of abstract rules and principles) when the underlying goal structures are explicated (or, reified, if you will). For example, Catrambone (1996) found that providing a meaningful label to a set of steps, namely, explication of a subprocedure, facilitate learning a statistical procedure. So ITS must explicate goal structures underlying problem solving. This is exactly one of the eight design principles proposed through the experience with the model tracing tutors (Anderson et al., 1995).

In my opinion, it might be interesting to build an ITS to support analogical transfer by explicating mapping between source and target domains. In the learning session, students learn domain principles with worked examples. In the analogical transfer session, ITS provides scaffolding for students to find mapping between old and new situations to make the learned principles applicable in the new situation.

2.2. Learning with examples and self-explanation

It is well known that self-explanation facilitates learning from examples. Chi et al found that during learning with worked examples, good learns generate many explanations which refine and expand the conditions for the action parts of the example solutions, and relate these actions to principles in the text. These self-explanations are guided by accurate monitoring of their own understanding and misunderstanding. Such learning results in example-independent knowledge and in a better understanding of the principles presented in the text (Chi, Bassok, Lewis, Reimann, & Glaser, 1989, p.145).

The above findings in cognitive psychology surely convince me that asking students to self-explain examples would facilitate learning. Indeed, as mentioned above, both SE-COACH and Geometry Explanation Tutor show that eliciting self-explanation on problem solving steps leads students to better learning.

2.3. From learning by examples to learning by problem solving

Kalyuga et al (2001) found that once learners attained certain levels of experience, the worked examples might become redundant hence brings little benefit. Practice at solving the equivalent problems may result in superior outcomes (p.587).

This cognitive principle would encourage ITS designers to design an instructional planner that takes account of student's progress in (or a degree of familiarity to) example learning. The ITS may need a student model that represents student's ability to understand examples. Or, it might be just a matter of time spent on examples (could be number of examples). This is an empirical question.

3. Challenges to implement example-based learning in ITS

Besides the issues of learning abstract principles and discovery learning, which are inquired in the exam question, there are several other interesting issues around the use of examples as a vehicle of learning in ITS.

3.1. Order of examples

Does the order of examples exposed to a student matter? Does some well tailored sequence of examples facilitate learning? Does a flaw in sequence hinder learning and/or even implant misconception that further lead students to negative transfer? Any clues to answer those questions cannot be found in the assigned readings, but my best guess is that a sequence of examples might affect learning. This is because students can overly generalize (or specialize) principles induced from examples.

3.2. Student modeling

To control a sequence of examples and also to switch the mode of learning from example based to problem-solving based, ITS needs to know the level of student's competence, namely the student model.

Inferring student model with the students just reading examples might be challenging due to a limited capacity of information that ITS can read off the students, i.e., the bandwidth (VanLehn, 1988). As is discussed in (Self, 1999), although we do not have to build extremely precise student model, ITS with accurate student model brings better learning than the blind (i.e., no student model at all) ITS.

One way to assess students' understanding and build a student model is to let students make self-explanations for examples they read. The two ITSs mentioned above follow this strategy; SE-COACH and Geometry Explanation Tutor. In SE-COACH, the student model is maintained as Bayesian network representing principle applications, which associate facts and goals. When a student correctly provide an explanation for a *principle application* or for the *goal* accomplished by a principle application, then probability of the corresponding node in Bayesian network is updated (Conati & VanLehn, 2000).

Geometry Explanation Tutor is a version of model tracing tutor (Aleven et al., 2001). The student model is hence represented as production rules. Question then is how to identify a production rule that is most closely related to student's justification. Aleven et al utilizes a natural language understanding technique to solve this problem, which is discussed below.

3.3. Understanding student's understanding (Language Issue)

As mentioned above, self-explanation facilitates learning from examples. So, the need to understand student's explanations will be one of the central issues in ITS research. This is where the techniques for natural language understanding take place. Natural language understanding contributes in two different purposes; (1) to identify a domain principle that corresponds to student's explanation, (2) to assess quality of student's explanation.

An example of the former use is Geometry Explanation Tutor. The goal of the natural language understanding component is to categorize a justification made by a student into one of 140 correct and incorrect description of 25 geometry rules (Aleven et al., 2001). In other words, Geometry Explanation Tutor has a collection of conceivable students' explanations, and identify student's explanation as one of these stored explanations.

There is no preceding study in the assigned readings that utilize natural language understanding to assess *quality* of students' explanations. To my best knowledge, the most suitable technique used in practical ITS toward this branch of learning from examples with self-explanation is latent semantic analysis (LSA) for essay grading (see, for example, Person, Graesser, Kreuz, & Pomeroy, 2001)[1]. The basic idea is to measure a “distance” between the student’s essay and the expert’s ideal one.

3.4. Tutoring strategies for example-based learning

What kind of tutoring strategy would apply for learning from examples? Chi et al (2001) found that providing context free prompting (e.g., “could you explain more?” “why do think so?” etc.) does enhance students’ learning. This strategy fits in a learning environment where ITS ask student to provide an explanation for examples. If ITS can fully understand student’s explanations, then more sophisticated dialogue based tutoring can be apply (Graesser, Person, & Magliano, 1995).

3.5. Use of examples to discover and learn abstract principles

The theory of learning from examples tells that if students eventually read sufficient examples and explain what they read, then they learn underlying principles to solve problems. So, in my opinion, if the issues on design for example-based learning discussed above are automated in ITS, then one can expect that students would eventually “discover” domain principles.

Can students learn “abstract” principles from examples? Although the answer depends on the definition of “abstract,” my answer is rather optimistic. Students do learn principles to solve examples, and that principles can be used to solve novel problems. Hence, some generalization must be occurred, and the result of generalization is, by definition, results in the abstract principles.

[1] This paper was dropped from the reading list in the early elaboration process with the exam committee. However, I actually read this paper, and this paper gives support to my speculation (or, creative imagination, if you will). Hence I cite the paper here.

Reference:

Aleven, V., & Ashley, K. D. (1997). Teaching Case-Based Argumentation Through a Model and Examples: Empirical Evaluation of an Intelligent Learning Environment. In B. d. Boulay & R. Mizoguchi (Eds.), *Proceedings of the World Conference on Artificial Intelligence in Education* (pp. 87-94). Amsterdam: IOS Press.

Aleven, V. A. W. M. M., Popescu, O., & Koedinger, K. R. (2001). A Tutorial Dialogue System with Knowledge-Based Understanding and Classification of Student Explanations. *2nd IJCAI Workshop on Knowledge And Reasoning In Practical Dialogue Systems*.

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. *Journal of the Learning Sciences, 4*(2), 167-207.

Caplan, L. J., & Schooler, C. (1999). On the use of analogy in text-based memory and comprehension: The interaction between complexity of within-domain encoding and between-domain processing. *Journal of the Learning Sciences, 8*(1), 41-70.

Catrambone, R. (1996). Generalizing solution procedures learned from examples. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 22*(4), 1020-1031.

Chi, M. T., Bassok, M., Lewis, M. W., Reimann, P., & Glaser, R. (1989). Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science*, *13*(2), 145-182.

Chi, M. T. H., Siler, S. A., Jeong, H., Yamauchi, T., & Hausmann, R. G. (2001). Learning from human tutoring. *Cognitive Science*, *25*, 471-533.

Conati, C., & VanLehn, K. (2000). Toward Computer-Based Support of Meta-Cognitive Skills: a Computational Framework to Coach Self-Explanation. *International Journal of Artificial Intelligence in Education, 11*, 389-415.

Graesser, A. C., Person, N. K., & Magliano, J. P. (1995). Collaborative dialogue patterns in naturalistic one-to-one tutoring. *Applied Cognitive Psychology*, *9*(6), 495-522.

Kalyuga, S., Chandler, P., Tuovinen, J., & Sweller, J. (2001). When problem solving is superior to studying worked examples. *Journal of Educational Psychology*, *93*(3), 579-588.

Koedinger, K. R., & Anderson, J. R. (1993). Reifying implicit planning in geometry: Guidelines for model-based intelligent tutoring system design. In S. P. Lajoie & S. J. Derry (Eds.), *Computers as cognitive tools* (pp. 15-45). Hillsdale, NJ: Lawrence Erlbaum Associates.

Ohlsson, S. (1986). Some principles of intelligent tutoring. *Instructional Science*, *14*(3-4), 293-326.

Person, N. K., Graesser, A. C., Kreuz, R. J., & Pomeroy, V. (2001). Simulating Human Tutor Dialog Moves in AutoTutor. *International Journal of Artificial Intelligence in Education*, *12*, 23-39.

Self, J. (1999). The defining characteristics of intelligent tutoring systems research: ITSs care, precisely. *International Journal of Artificial Intelligence in Education*, *10*, 350-364.

VanLehn, K. (1988). Student modeling. In M. C. Polson & J. J. Richardson (Eds.), *Foundations of Intelligent Tutoring Systems* (pp. 55-78). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Appendix I: Assigned Reading List:

Design Principles

Collins, A., Neville, P., & Bielaczyc, K. (2000). The Role of Different Media in Designing Learning Environments. International Journal of Artificial Intelligence in Education, 11, 144-162.

Self, J. (1999). The defining characteristics of intelligent tutoring systems research: ITSs care, precisely. International Journal of Artificial Intelligence in Education, 10, 350-364.

Corbett, A., Koedinger, K., & Anderson, J. R. (1997). Intelligent Tutoring Systems. In M. G. Halander & T. K. Landauer & P. V. Prabhu (Eds.), Handbook of Human-Computer Interaction (2nd ed., pp. 849-874). Amsterdam: Elsevier.

Shute, V. J., & Psotka, J. (1994). Intelligent Tutoring Systems: Past, Present, and Future (AL/HR-TP-1994-0005). Brooks Air Force Base, TX: Armstrong Laboratory.

VanLehn, K. (1988). Student modeling. In M. C. Polson & J. J. Richardson (Eds.), Foundations of Intelligent Tutoring Systems (pp. 55-78). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Greeno, J. G., Brown, J. S., Shalin, V., Bee, N. V., Lewis, M. W., & Vitolo, T. M. (1986). Cognitive Principles of Problem Solving and Instruction (Technical Report, NR154-497). Pittsburgh, PA: University of Pittsburgh.

Ohlsson, S. (1986). Some principles of intelligent tutoring. Instructional Science, 14(3-4), 293-326.

Effectiveness of Human Tutoring

Chi, M. T. H., Silera, S. A., Jeonga, H., Yamauchia, T., & Hausmann, R. G. (2001). Learning from human tutoring. Cognitive Science, 25, 471-533.

Graesser, A. C., Person, N. K., & Magliano, J. P. (1995). Collaborative dialogue patterns in naturalistic one-to-one tutoring. Applied Cognitive Psychology, 9(6), 495-522

Merrill, D. C., Reiser, B. J., Ranney, M., & Trafton, J. G. (1992). Effective tutoring techniques: A comparison of human tutors and intelligent tutoring systems. Journal of the Learning Sciences, 2(3), 277-305.

McArthur, D., Stasz, C., & Zmuidzinas, M. (1990). Tutoring Techniques in Algebra. Cognition and Instruction, 7(3), 197-244.

Bloom, B. S. (1984). The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. Educational Researcher, 13(6), 4-16.

Cohen, P. A., Kulik, J. A., & Kulik, C. l. C. (1982). Education outcomes of tutoring: A meta-analysis of findings. American Educational Research Journal, 19(2), 237-248.

Effective Machine Tutors

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. Journal of the Learning Sciences, 4(2), 167-207.

Sleeman, D., Kelly, A. E., Martinak, R., Ward, R. D., & Moore, J. L. (1989). Studies of Diagnosis and Remediation with High School Algebra Students. Cognitive Science, 13, 551-568.

Feedback

VanLehn, K., Lynch, C., Taylor, L., Weinstein, A., Shelby, R., Schulze, K., Treacy, D., & Wintersgill, M. (2002). Minimally invasive tutoring of complex physics problem solving. In S. A. Cerri & G. Gouarderes & F. Paraguacu (Eds.), Proceedings of the 6th International Conference on Intelligent Tutoring Systems (pp. 367-376).

Hume, G., Michael, J., Rovick, A., & Evens, M. (1996). Hinting as a tactic in one-on-one tutoring. Journal of the Learning Sciences, 5(1), 23-47.

McKendree, J. (1990). Effective feedback content for tutoring complex skills. Human Computer Interaction, 5(4), 381-413.

Schooler, L. J., & Anderson, J. R. (1990). The Disruptive Potential of Immediate Feedback. In M. Piattelli-Palmarini (Ed.), Proceedings of the Annual Conference of the Cognitive Science Society (pp. 702-708): Erlbaum.

Schmidt, R. A., Young, D. E., Swinnen, S., & Shapiro, D. C. (1989). Summary knowledge of results for skill acquisition: Support for the guidance hypothesis. Journal of Experimental Psychology: Learning, Memory, and Cognition, 15(2), 352-359.

Kulik, J. A., & Kulik, C. l. C. (1988). Timing of feedback and verbal learning. Review of Educational Research, 58(1), 79-97.

Reification

Koedinger, K. R., & Anderson, J. R. (1993). Reifying implicit planning in geometry: Guidelines for model-based intelligent tutoring system design. In S. P. Lajoie & S. J. Derry (Eds.), Computers as cognitive tools (pp. 15-45). Hillsdale, NJ: Lawrence Erlbaum Associates.

Self-explanation

Aleven, V. A. W. M. M., Popescu, O., & Koedinger, K. R. (2001). A Tutorial Dialogue System with Knowledge-Based Understanding and Classification of Student Explanations. 2nd IJCAI Workshop on Knowledge And Reasoning In Practical Dialogue Systems, Seattle.

Conati, C., & VanLehn, K. (2000). Toward Computer-Based Support of Meta-Cognitive Skills: a Computational Framework to Coach Self-Explanation. International Journal of Artificial Intelligence in Education, 11, 389-415.

Aleven, V., & Ashley, K. D. (1997). Teaching Case-Based Argumentation Through a Model and Examples: Empirical Evaluation of an Intelligent Learning Environment. In B. d. Boulay & R. Mizoguchi (Eds.), Proceedings of the World Conference on Artificial Intelligence in Education (pp. 87-94). Amsterdam: IOS Press.

Appendix II: Supplement Papers Read in the Past

Chi, M. T., Bassok, M., Lewis, M. W., Reimann, P., & Glaser, R. (1989). Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science, 13*(2), 145-182.

Person, N. K., Graesser, A. C., Kreuz, R. J., & Pomeroy, V. (2001). Simulating Human Tutor Dialog Moves in AutoTutor. *International Journal of Artificial Intelligence in Education, 12*, 23-39.