

Intelligent Tutoring Systems: New Challenges and Directions

Cristina Conati

Department of Computer Science
University of British Columbia
conati@cs.ubc.ca

Abstract

Intelligent Tutoring Systems (ITS) is the interdisciplinary field that investigates how to devise educational systems that provide instruction tailored to the needs of individual learners, as many good teachers do. Research in this field has successfully delivered techniques and systems that provide adaptive support for student problem solving in a variety of domains. There are, however, other educational activities that can benefit from individualized computer-based support, such as studying examples, exploring interactive simulations and playing educational games. Providing individualized support for these activities poses unique challenges, because it requires an ITS that can model and adapt to student behaviors, skills and mental states often not as structured and well-defined as those involved in traditional problem solving. This paper presents a variety of projects that illustrate some of these challenges, our proposed solutions, and future opportunities.

1 Introduction

Since the early 1970s, the field of Intelligent Tutoring Systems (also known as Artificial Intelligence in Education) has investigated combining research in Artificial Intelligence, Cognitive Science and Education to devise intelligent agents that can act as tutors in computer-aided-instruction (CAI). Traditional CAI systems support learning by encoding sets of exercises and the associated solutions, and by providing predefined remediation actions when the students' answers do not match the encoded solutions. This form of CAI can be very useful in supporting well-defined drill-and-practice activities. However, it is difficult to scale to more complex pedagogical activities, because the system designer needs to define all relevant problem components, all solutions (correct or incorrect) that the system needs to recognize, and all possible relevant pedagogical actions that the tutor may need to take.

Research in ITS has been investigating how to make computer-based tutors more flexible, autonomous and adaptive to the needs of each student by endowing them with explicit knowledge of the relevant components of the teaching

process and with reasoning capabilities to turn this knowledge into intelligent behavior. There are three types of knowledge that an intelligent tutor (human or artificial) needs to have to be able to aid student learning: (i) knowledge about the target instructional domain, (ii) knowledge about the student, and (iii) knowledge about the relevant pedagogical strategies. In addition, an artificial tutor needs to have *communication knowledge* about how to present the desired information via the computer medium given the available output channels. These different types of knowledge contribute to defining the behavior of a complete intelligent tutor for problem solving activities as follows. The tutor uses pedagogical knowledge represented in the *pedagogical model*, domain knowledge stored in the *domain model* and knowledge about the current state of the student stored in the *student model* to select a suitable new problem for the student. Using the domain knowledge and its communication knowledge, the tutor presents the selected problem to the student in the format most suitable for the student's abilities and preferences. Then it monitors the student's solution to the problem and compares it with its known solution (or set of relevant alternative solutions) to decide whether the student's solution is appropriate or requires pedagogical interventions. A key difference between many intelligent tutors and more traditional CAI systems is that in the ITS the relevant solutions against which to compare the student's input do not need to be predefined by a human author. These solutions are generated in real-time by the ITS itself, given the problem definition and the knowledge in the domain model. The comparison between the student's and the computer's solution(s) is used to both update the ITS's belief regarding the student's relevant domain knowledge and skills (i.e., its *student model*), and to generate an adequate tutorial action (e.g., help with an incorrect solution step, praise for a correct solution).

It should be noted that not all ITS include the four components mentioned above, and that each component can be present at various levels of sophistication. Most ITS, for instance, include fairly rich domain and student models (e.g., Corber and Anderson 1995, Conati *et al.*, 2002), but the pedagogical model may consist of a simple set of heuristics with no explicit communication model. Some ITS, on the other hand, may have a rich communication model that allows the system to interact with the student by using natural

language (e.g. VanLehn *et al.*, 2007). Similarly, ITS vary in the type and sophistication of the pedagogical actions they can perform. Some ITS provide step-by-step monitoring of the student’s solution as it is being generated (e.g., Corber and Anderson 1995, Conati *et al.*, 2002), while others provide feedback on the final solution only (e.g., Mitrovic *et al.*, 2007). Some ITS select the next activity for the students, others let the student select it.

ITS research has successfully delivered techniques and systems that provide adaptive support for student problem solving or question-answering activities in a variety of domains (e.g., programming, physics, algebra, geometry, SQL and introductory computer science). Several of these systems are actively used in real-world settings (e.g., Mitrovic *et al.*, 2007, <http://www.carnegielearning.com/products.cfm>) and have even contributed to changing traditional school curricula (Koedinger *et al.*, 1995).

There are, however, other educational activities that can benefit from individualized computer-based support, such as learning from examples, exploring interactive simulations, playing educational games and learning with a group of peers. Providing individualized support for these activities poses unique challenges, because it requires an ITS that can model domains as well as student behaviors and mental states that may not be as structured and well-defined as those involved in traditional problem solving. For instance, an ITS that provides support for exploration-based learning must be able to “understand” exploratory activities so that it can propose them to students. It also needs to know what it means to explore a given concept or domain effectively, so that it can monitor the student’s exploration process and provide adequate feedback when needed. In recent years, the ITS community has actively taken on these new challenges, aided by advances in AI research. The remainder of this paper will focus on one specific endeavor to push ITS research beyond support for traditional problem solving: devising intelligent tutors that can scaffold meta-cognitive skills.

2 Intelligent tutors that scaffold meta-cognition

Meta-cognition refers to “*one’s knowledge concerning one’s own cognitive processes and products or anything related to them*” (Flavell 1976); more informally, meta-cognition has been referred to as “*thinking about thinking*”. Meta-cognitive skills are therefore domain-independent abilities that are an important aspect of knowing how to learn in general. Examples include, among others, the ability to monitor one’s learning progress (*self-monitoring*), the tendency to explain instructional material to oneself in terms of the underlying domain knowledge (*self-explanation*), the ability to learn from examples (*analogical reasoning*), the ability to appropriately seek tutoring help. Individuals vary significantly in these abilities, and thus several ITS researchers have been investigating how to devise tutors that can help students acquire the relevant meta-cognitive skills. While some researchers have focused on creating tools that can

scaffold meta-cognition by design (e.g., Luckin and Ham-merton 2002, Alevan and Koedinger 2002), others have been investigating how to capture a user’s need for meta-cognitive support in real-time during interaction, to enable the ITS to respond accordingly. Roll *et al.*, for instance, have devised a model that enables an ITS to track and scaffold a student’s tendency to effectively use the available help facilities [Roll *et al.*, 2007]. In our work, we have focused on modeling and scaffolding students’ cognitive skills related to learning from examples, as well as skills related to learning effectively from exploration. Arguably, the higher the level of the user states to be captured, the more difficult they are to assess unobtrusively from simple interaction events. The next two sections briefly describe our progress in this direction.

2.1 An intelligent tutor for example-based learning

Research in cognitive science has provided extensive evidence of the utility of worked-out example solutions as learning aids. (e.g., Anderson *et al.*, 1984, VanLehn 1996). However, this research also indicates that there is great variation in how effectively different students learn from examples, because of individual differences in the meta-cognitive skills relevant to succeeding in this activity. Two of these meta-cognitive skills are *self-explanation* and *min-analogy*. Self-explanation involves elaborating and clarifying available instructional material to oneself (Chi 2000). Min-analogy involves transferring from an example only the minimum amount of information necessary to enable successful problem solving, as opposed to copying indiscriminately from the example (Vanlehn 1998). We have devised ExBL, an ITS that takes into account individual differences in these cognitive skills to provide user-adaptive support to example-based learning (Conati *et al.*, 2006). ExBL complements Andes, an ITS designed to support physics problem solving at the college level (Conati *et al* 2002), and includes two components. The first component, known as the SE (Self-Explanation)-Coach, supports example studying prior to problem solving. The second component, known as the EA (Example-Analogy)-Coach, supports the effective use of examples *during* problem solving (i.e., analogical problem solving, or APS from now on).

In order to tailor its scaffolding to a student’s needs, ExBL must be capable of monitoring and assessing each student’s performance with respect to the target pedagogical tasks. Thus, the framework needs an internal representation of these tasks, against which to compare the student’s problem-solving and example-studying behaviours. It also needs to encode in a student model its assessment of the student’s domain knowledge and relevant meta-cognitive skills.

The above requirements are implemented in the architecture shown in Figure 1. The user interface component provides interactive tools for students to study examples (SE-Coach) and to use examples during problem solving (EA-Coach). All student interface actions are monitored and

assessed against the system’s internal representation of the relevant problem/example solutions. This internal representation, known as the *solution graph*, is automatically built before run-time by the component labelled as *problem solver* in Figure 1 (left) starting from: (i) a knowledge base of physics and planning rules (*Domain and planning rules* in Figure 1) and (ii) a formal description of the initial situation for the examples/problems involved in each task (*Problem definition* in Figure 1) (Conati and VanLehn 2000). Each solution graph is a dependency network that represents how each solution step derives from previous steps and physics knowledge.

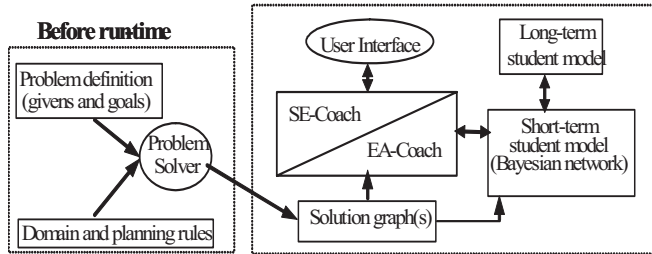


Figure 1: ExBL architecture

Both SE-Coach and EA-Coach use the solution graph to provide feedback on students’ performance during example studying and analogical problem solving, by matching students’ interface actions to elements in the solution graph. In addition to serving as the basis for the ExBL’s ability to provide feedback, the solution graph is used to build its student models. Each time a student opens a new exercise, the corresponding solution graph provides the structure for a Bayesian network that forms the short-term student model for the currently active Coach (see right side of Figure 1). The Bayesian network uses information on the student’s interface actions to generate a probabilistic assessment of the student’s knowledge and relevant meta-cognitive tendencies at any given point during the interaction. This procedure allows the system to generate tailored interventions to foster effective meta-cognitive skills when the model assesses the student as having knowledge gaps or requiring improvement in her meta-cognitive behaviours. The prior probabilities to initialise the rule nodes in the Bayesian network come from the long-term student model (see Figure 1), which contains a probabilistic assessment of a student’s knowledge of each rule in the ExBL’s knowledge base at the time when a new exercise is started, given the student’s performance in all the exercises solved up to that point.

The SE-Coach uses the architecture described above to help a student better understand a given example, based on its current assessment of the student’s knowledge, the student’s reading patterns (tracked via an interface artefact, see Conati and VanLehn 2000 for details) and possible student explanations on the example that the student can generate via dedicated interface tools. Based on this assessment, the SE-Coach guides the student to more carefully explain parts

of the example that may not be fully understood. The SE-Coach also includes a component that automatically generates example solutions at different levels of detail and helps students generate the missing solution steps, in order to support the student in the transition from example-studying to pure problem solving (Conati and Carenini 2001).

The EA-Coach uses the architecture in Figure 1 to support effective analogical problem solving by selecting for each student and current problem an example that maximises both problem solving success and student learning. The example-selection process relies on a decision-theoretic mechanism that, given a problem and a set of examples, computes for each example the probability that it can help the student solve the problem and learn in the process, given the current assessment of the student’s physics knowledge and meta-cognitive skills. The example with the maximum-expected utility in terms of problem-solving success and learning is then presented to the student.

A formal evaluation of the SE-Coach component with adaptive support for example studying showed that it can help students learn more effectively than a version with no adaptive support, when students are in the early stages of learning a new topic (Conati *et al.*, 2006). A formal evaluation of the EA-Coach’s selection process showed that it can significantly increase the number of appropriate student analogical problem-solving behaviours compared with an approach that selects the example most similar to the current problem, as done by other ITS that support example-based problem solving (Muldner and Conati 2007). These results, although obtained in controlled laboratory studies as opposed to classroom settings, represent encouraging evidence that it is feasible to devise intelligent tutors that can model, adapt and support student meta-cognition and subsequent learning. What remains to be seen is whether the meta-cognitive skills themselves are learned in the process, i.e., if a student can retain them when the ITS support is no longer available.

2.2 Supporting user interaction with exploratory learning environments.

This research seeks to provide intelligent support for exploratory learning. The capability to explore effectively is relevant to many tasks involving interactive systems, but not all users possess this capability in equal measure (e.g., Shute and Glaser 1990). We developed a model of exploratory behavior that an ITS can use to improve user exploration via interventions tailored to the user’s needs. This task is challenging because it requires assessing the effectiveness of behaviors for which there is no formal definition of correctness. We tackled the challenge with a probabilistic model that assesses exploration effectiveness by integrating information on user actions, knowledge and whether a user actually reasons about (self-explains) his/her exploratory actions. Self-explanation is a well-known meta-cognitive skill in Cognitive Science, but this work is the first to model self-explanation in the context of exploration-based learning.

We developed the model in the context of ACE (Adaptive Coach for Exploration). ACE is an ITS that supports student exploration of mathematical functions via a set of interactive simulations designed to illustrate function-related concepts, such as the relationship between the input and the output of a function, or between a function's equation and its graph. Figure 2 shows the main interaction window for ACE's Plot Unit, an activity that allows the learner to explore the relationship between a function's plot and equation by moving the plot in the Cartesian plane and observing the effects on the equation (displayed below the plot area in Figure 2).

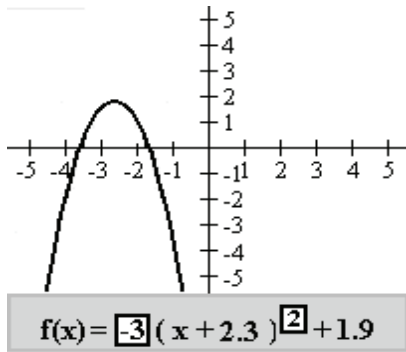


Figure 2: ACE's Plot Unit

The student can also change the equation parameters and see how the change affects the plot. Each function type (e.g., constant, linear and power) has an associated set of 'exploration cases' that together illustrate the full range of function attributes. For example, linear functions are defined by two parameters, the function slope and the y-intercept. Therefore, in order to gain a broad understanding of linear functions, the student should study positive and negative intercepts, and positive, negative and zero slopes.

ACE monitors the student's interaction with its simulations, and generates interventions to improve those behaviors deemed to be suboptimal. For instance, it suggests which further exploratory actions to perform when a student's exploration of a given activity is incomplete (Bunt and Conati 2003). To judge the effectiveness of a student's exploratory behaviors, ACE relies on a probabilistic student model. The first version of the model was a Dynamic Bayesian Network (DBN) that included (i) nodes to represent all possible exploration cases; (ii) nodes to represent student understanding of related mathematical concepts; and (iii) links representing how exploration of relevant cases relates to concept understanding. To assess whether an exercise has been explored effectively, this version of the ACE model just used evidence from the student's interface actions. Initial studies of this version of the system generated encouraging evidence that it could help students learn better from exploration (Bunt and Conati 2003). However, these studies also showed that ACE sometimes overestimated students'

exploratory behavior, because it considered interface actions to be sufficient evidence of good exploration, without taking into account whether a student was reasoning, or *self-explaining* the outcome of these actions. For instance, a student who quickly moves a function plot around the screen, but never reflects on how these movements change the function equation, is performing many exploratory actions but can hardly learn from them. Still, the first ACE student model would likely judge this type of behavior as good exploration.

To circumvent this problem, we devised a new version of the student model that includes assessment of the student's self-explanation behavior during exploration-based learning (Conati and Merten 2007). To assess self-explanation, this model uses evidence derived from both the time spent on each exploratory action and the student attention patterns monitored via an eye-tracking system. This work was one of the first attempts to use eye-tracking information in real-time to assess complex user mental states. We formally evaluated the model using both time and eye-tracking information against (i) a model using only time as a predictor of self-explanation and (ii) the earlier ACE model that ignores self-explanation and uses only the number of user interface actions as a predictor of effective exploration. We found that

- The model including both gaze and time data provides better assessment of student self-explanation than the model using only time. The difference is statistically significant.
- Assessing self-explanation significantly improves the assessment of student exploratory behavior, and the accuracy of the latter increases with increased accuracy of self-explanation assessment. All improvements are statistically significant.

This works shows that it is possible to increase the bandwidth of an ITS that needs to capture high-level user mental states by using information on user attention. Since eye-tracking technology is becoming increasingly more precise and unobtrusive, this opens many opportunities for devising intelligent tutors that can understand and adapt to complex user reasoning processes.

3 Conclusions

Given our society's increasing need for high quality teaching and training, computer-supported education is becoming critical to complementing human tutoring in a large variety of fields and settings. Research in Intelligent Tutoring Systems leverages advances in Artificial Intelligence, Cognitive Science and Education to increase the ability of computer-supported education to autonomously provide learners with effective educational experiences tailored to their specific needs, as good human tutors do.

In this paper, we have provided examples of one current direction of ITS research aimed at extending the reach of this technology toward new forms of computer-based in-

struction beyond traditional problem solving: providing intelligent tutoring for meta-cognitive skills. This endeavor is only one of several new directions in ITS research. Other new forms of intelligent computer-based tutoring that have been actively investigated include, among others: support for collaborative learning (e.g., Isotami and Misogici 2008); emotionally intelligent tutors that take into account both student learning and affect when deciding how to act (e.g., Conati and Maclaren 2009, D’Mello *et al.*, 2008); teachable agents that can help students learn by acting as peers that students can tutor (e.g., Leelawong and Biswas 2008); intelligent support for learning from educational games (e.g., Manske and Conati 2005, Johnson 2007); and intelligent tutoring for ill-defined domains (e.g., Lynch *et al.*, 2008). Providing these forms of intelligent tutoring, like providing intelligent support for meta-cognition, poses unique challenges, because it requires an ITS that can model domains as well as student behaviors and mental states often not as structured and well-defined as those involved in traditional problem solving. Advances in AI techniques for reasoning under uncertainty, machine learning, decision-theoretic planning, as well as the increasing availability of sensors that can help capture the relevant user states, are promising means for the field to face these challenges. Success in these endeavors has the potential to have great impact on our society, and on its ever-increasing need for high quality teaching and training.

Acknowledgments

The research presented in this paper was funded by the National Sciences and Engineering Research Council of Canada (NSERC). We would like to thank all of the researchers who have provided invaluable contributions to this work: Andrea Bunt, Giuseppe Carenini, Christina Merten, Kasia Muldner, David Ternes.

References

[Aleven & Koedinger, 2002] Aleven, V. & Koedinger, K. 2002. An effective meta-cognitive strategy: Learning by doing & explaining with a computer-based cognitive tutor. *Cognitive Science*, 26(2), 147-179.

[Anderson *et al.*, 1984] Anderson, J. R., Farrell, R. & Saurers, R. 1984. Learning to program in Lisp. *Cognitive Science*, 8, 87-129

[Bunt and Conati 2003] Bunt, A. and C. Conati (2003). Probabilistic Student Modeling to Improve Exploratory Behaviour. *Journal of User Modeling and User-Adapted Interaction*, 13(3).

[Chi 2000] Chi, M. T. H. (2000). Self-Explaining Expository Texts: The Dual Processes of Generating Inferences and Repairing Mental Models. *Advances in Instructional Psychology*. R. Glaser. Mahwah, NJ, Lawrence Erlbaum Associates: 161-238

[Conati and VanLehn 2000] Conati, C. and K. VanLehn (2000). Toward Computer-Based Support of Meta-Cognitive Skills: a Computational Framework to Coach Self-Explanation. *Int. Journal of AI in Education* 11.

[Conati and Carenini 2001]. Conati C. and Carenini G. (2001) Generating Tailored Examples to Support Learning via Self-explanation . In *Proceedings of IJCAI '01, the Seventeenth International Joint Conference on Artificial Intelligence*, Seattle, WA.

[Conati et al., 2002] Conati C., Gertner A., VanLehn K., 2002. Using Bayesian Networks to Manage Uncertainty in Student Modeling. *User Modeling and User-Adapted Interaction*. 12(4) p. 371-417

[Manske and Conati 2005] Manske, M. and C. Conati (2005). Modelling Learning in Educational Games in AIED 05, *Proceedings of the 12th International Conference on AI in Education*. 2005. Amsterdam, The Netherlands.

[Conati *et al.*, 2006] Conati, C., Muldner, K. & Carenini, G. 2006. From Example Studying to Problem Solving via Tailored Computer-Based Meta-Cognitive Scaffolding: Hypotheses and Design. *Technology, Instruction, Cognition and Learning (TICL): Special Issue on Problem Solving Support in Intelligent Tutoring Systems*. 4(2), 139-190.

[Conati and Merten 2007]. Conati C. and Merten C. (2007). Eye-Tracking for User Modeling in Exploratory Learning Environments: an Empirical Evaluation. *Knowledge Based Systems*, Volume 20, Issue 6, Elsevier Elsevier Science Publishers B. V. Amsterdam, The Netherlands

[Conati and Maclaren 2009]. Conati C and Maclaren H.. Empirically Building and Evaluating a Probabilistic Model of User Affect. *User Modeling and User-Adapted Interaction* (to appear).

[D’Mello et al., 2008] D’Mello, S.K., Craig, S.D., Witherspoon, A. W., McDaniel, B. T., and Graesser, A. C. Automatic Detection of Learner’s Affect from Conversational Cues. *User Modeling and User-Adapted Interaction*, 18(1), (2008).

[Corbet and Anderson, 1995] Corbet A. T. and Anderson J. R. (1995). Knowledge tracing: Modeling the acquisition of procedural knowledge. *Journal of User Modeling and User-Adapted Interaction* 4(4).

[Flavell, 1976] Flavell, J. H. 1976. Metacognitive Aspects of Problem Solving. *The Nature of Intelligence*. Hillsdale, New Jersey, Lawrence Erlbaum, 231-235.

[Isotani and Mizoguchi 2008] Isotani S. and Mizoguchi R. (2008): Theory-Driven Group Formation through Ontologies. *Intelligent Tutoring Systems 2008*: 646-655

[Johnson 2007] Johnson, W. L. (2007). Serious use for a serious game on language learning. In *Proc. of the 13th Int. Conf. on Artificial Intelligence in Education*, Los Angeles, USA.

- [Koedinger *et al.*, 1995] Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. (1995). Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education* (1997), 8, p. 30-43
- [Leelawong and Biswas 2008] Leelawong, K., & Biswas, G. Designing Learning by Teaching Agents: The Betty's Brain System, *International Journal of Artificial Intelligence in Education*, vol. 18, no. 3, pp. 181-208, 2008.
- [Luckin and Hammerton 2002] Luckin, R. and Hammerton, L. (2002) Getting to Know Me: Helping Learners Understand Their Own Learning Needs through Metacognitive Scaffolding. In: *Intelligent Tutoring Systems: Proceedings of the 6th International Conference, ITS 2002*, Springer Berlin / Heidelberg, pp. 759-771.
- [Lynch *et al.*, 2008], Pinkwart, N., Ashley, K., & Alevén, V. (2008). What do argument diagrams tell us about students' aptitude or experience? A statistical analysis in an ill-defined domain. In V. Alevén, K. Ashley, C. Lynch, & N. Pinkwart (Eds.), *Proceedings of the Workshop on Intelligent Tutoring Systems for Ill-Defined Domains at the 9th International Conference on Intelligent Tutoring Systems* (p. 62-73). Montreal (Canada).
- [Mitrovic *et al.*, 2007] Mitrovic, A., Martin, B., Suraweera, P. Constraint-based tutors: past, present and future. *IEEE Intelligent Systems, special issue on Intelligent Educational Systems*, vol. 22, no. 4, pp. 38-45, July/August 2007
- [Muldner & Conati, 2007] Muldner, K. & Conati, C. 2007. Evaluating a Decision-Theoretic Approach to Tailored Example Selection. In *Proc. of the 20th International Joint Conference on Artificial Intelligence (IJCAI'07)*, 483-489.
- [Roll *et al.*, 2007]. Roll I., Alevén, V., McLaren, B., & Koedinger, K. (2007). Can help seeking be tutored? Searching for the secret sauce of metacognitive tutoring. In R. Luckin, K. Koedinger, & J. Greer (Eds.), *Proceedings of the 13th International Conference on Artificial Intelligence in Education* (pp. 203-210). Amsterdam, the Netherlands: IOS Press.
- [Shute and Glaser 1990]. Shute, V. J. & Glaser, R. (1990). a large-scale evaluation of an intelligent discovery world: Smithtown. *Interactive Learning Environments*, 1, 51-77
- [VanLehn, 1996] VanLehn, K. 1996. Cognitive skill acquisition. *Annual Review of Psychology*, 47, 513-539.
- [VanLehn, 1998] VanLehn, K. 1998. Analogy Events: How Examples are Used During Problem Solving. *Cognitive Science*, 22(3), 347-388
- [Vanlehn *et al.*, 2007] VanLehn, K., Graesser, A.C., Jackson, G.T., Jordan, P., Olney, A., & Rose, C.P. (2007). When are tutorial dialogues more effective than reading? *Cognitive Science*, 31, 3-62