Do Micro-Level Tutorial Decisions Matter: Applying Reinforcement Learning To Induce Pedagogical Tutorial Tactics

Min Chi[1], Kurt VanLehn[2], and Diane Litman[3]

[1] Machine Learning Department, Carnegie Mellon University, PA, 15213 USA
minchi@cs.cmu.edu
[2] School of Computing and Informatics, Arizona State University, AZ, 85287 USA
Kurt.Vanlehn@asu.edu
[3] Department of Computer Science & Learning Research Development Center, University of Pittsburgh, PA, 15260 USA litman@cs.pitt.edu

Abstract. Pedagogical tutorial tactics are policies for a tutor to decide the next action when there are multiple actions available. When the contents were controlled so as to be the same, little evidence has shown that tutorial decisions would impact students' learning. In this paper, we applied Reinforcement Learning (RL) to induce two sets of tutorial tactics from pre-existing human interaction data. The NormGain set was derived with the goal of enhancing tutorial decisions that contribute to learning while the InvNormGain set was derived with the goal of enhancing those decisions that contribute less or even nothing to learning. The two sets were then compared with human students. Our results showed that when the contents were controlled so as to be the same, different pedagogical tutorial tactics would make a difference in learning and more specifically, the NormGain students outperformed their peers.

Keywords: Reinforcement Learning, Human Learning, Intelligent Tutoring Systems, Pedagogical Strategy.

1 Introduction

Human one-on-one tutoring is one of the most effective educational interventions in that tutored students often perform significantly better than students in classroom settings[4]. One hypothesis as to the effectiveness of human one-on-one tutoring comes from the detailed management of "micro-steps" in tutorial dialogue[12, 13]. A typical Intelligent Tutoring System (ITS) is step-based[20]. Once a student enters a step, then the tutor gives feedback and/or hints. For example, in order to solve a physics problem, the student need to apply several domain principles, some of which may need to be applied multiple times. Each principle application can be seen as a step in the ITS. In a physics tutor, for example, applying the definition of Kinetic Energy ($KE = \frac{1}{2}mv^2$) to solve for the kinetic energy of a falling rock at T_0 is a step. Human tutors, by contrast, often scaffold students via a series of micro-steps leading to the full step. In the step mentioned above, for instance, a human tutor can take the following **micro-level** steps: selecting the principle to apply; writing the corresponding

1. **T:** So let's start with determining the value of KE_0.
2. **T:** Which principle will help you calculate the rock's kinetic energy at T0? Please provide the name of the principle, not an equation. **{ELICIT}**
3. **S:** Definition of kinetic energy
4. **T:** Yes, I agree. Now I will write the equation for applying the definition of kinetic energy to the rock at T0: KE0 = 1/2*m*v0^2 **{TELL}**

Fig. 1: An example Dialog

equation; solving the equation; and engaging in some qualitative discussion about the principle.

Fig. 1 shows a sample dialog for two micro-steps. In Fig. 1, each numbered line represents a dialog turn. The labels **T** and **S** designate tutor and student turns respectively. In this example, the tutor and the student first select a principle (lines 2 & 3) and then write the corresponding equation (line 4). Some of the tutor turns in Fig. 1 are labeled **{ELICIT}** or **{TELL}**. This label designates a ***tutorial decision step*** wherein the tutor has to make a tutorial decision deciding whether to elicit the requisite information with a question or to tell the student the information. For example, in line 2, the tutor chooses to *elicit* the answer from the student by asking the question, "Which principle will help you calculate the rock's kinetic energy at T0? Please provide the name of the principle, not an equation." If the tutor elected to tell the students, however, then he or she would have stated, "To calculate the rock's kinetic energy at T0, let's apply the definition of Kinetic Energy." Both actions cover the same target knowledge.

If the effectiveness of human one-on-one tutoring lies in tutors' ability to scaffold a series of micro-steps leading to a step entry, then we would expect human tutors to be more effective than step-based tutors as both require students to enter the same major steps. In several tests of this hypothesis, neither human tutors nor Natural Language (NL) tutoring systems designed to mimic human tutors, outperformed step-based systems[10, 22]. All three types of tutors, however, were more effective than no instruction (e.g., students reading material and/or solving problems without feedback or hints). One possible conclusion is that tutoring is effective, but that the micro-steps of human tutors and NL tutoring systems provide no additional value beyond conventional step-based tutors[21].

On the other hand, such a conclusion would be premature. It could simply be that neither human tutors nor their computer mimics are good at making micro-step decisions. That is, the use of micro-steps is good, but human tutors (and their mimics) lack the effective pedagogical skills to select appropriately. Indeed, although it is commonly assumed that human expert tutors have effective pedagogical skills, little evidence has been presented to date demonstrating that. In order to execute pedagogical skills effectively, it is assumed that tutors should adapt their behaviors to students' needs based upon students' current knowledge level, general aptitude, emotional state and other salient features. However, pre-

vious research has cast doubt on the assumption. Chi, Roy, and Hausman[8] found that human tutors do not seem to maintain an accurate model of student's knowledge level during the tutoring process. Similarly, Putnam[17] found that experienced tutors did not attempt to form detailed models of the students' knowledge before attempting remedial instruction. Rather, each teacher appeared to move through a general curricular script irrespective of the student's state. For the purposes of this paper the term *"pedagogical tutorial tactics"* will be used to refer to the policies for selecting the tutorial action at each micro-step level when there are multiple actions available.

In this study, our primary research question is whether pedagogical tutorial tactics would impact students' learning. We focus on two types of tutorial decisions, Elicit vs.Tell (ET) and Justify vs. Skip-Justify (JS). When making ET decisions the tutor decides whether to *elicit* the next step from the student or to *tell* them the step directly. The JS decisions address points where the tutor may optionally ask students to *justify* a step they have taken or entry they have made. Neither decision is well-understood. There are many theories, but no widespread consensus on how or when an action should be taken[1, 7, 9, 14].

In order to investigate our research question, we applied a general data-driven methodology, Reinforcement Learning (RL), to induce pedagogical tutorial tactics directly from pre-existing interactivity data. We used an NL Tutoring System called Cordillera[23]. In order to avoid confounds due to imperfect NL understanding, we replaced the NL understanding module with a human wizard. During tutoring, the wizard's sole task was to match students' answers to one of the available responses. The wizard made no tutorial decisions.

Previously we investigated whether the RL-induced pedagogical tutorial tactics would improve students' learning[6]. This was done by first collecting an Exploratory dataset in 2007. 64 college students, the Exploratory group, were trained on a version of Cordillera, called random-Cordillera, where both ET and JS decisions were made randomly. From the Exploratory corpus, we applied RL to induce a set of policies, named DichGain policies. They were named after the fact that when applying RL, we dichotomized the reward function so that there were only two levels of reward. The induced DichGain policies were implemented back to Cordillera and the new version of Cordillera was named DichGain-Cordillera. Apart from following the policies (random vs. DichGain), the remaining components of Cordillera, including the GUI interface, the same training problems, and the tutorial scripts, were left untouched. DichGain-Cordillera's effectiveness was tested by training a new group of 37 college students in 2008. It was shown that no significant overall difference was found between the two groups on the pretest, posttest, or the NLGs[6, 5]. There were at least two potential reasons for such lack of difference. First, it might be caused by limitations in our RL approach; for example, in order to induce the DichGain policies, we defined only 18 features and used a greedy-like procedure to search for a small subset of it as the state representation[6]. Second, rather than randomly assigning students into the two groups, the Exploratory data was collected in 2007 while the DichGain data was collected in 2008.

Therefore, in this study we included multiple training datasets, a larger feature set and more feature selection approaches in our RL approach and run a full comparison by random assignment of students to two comparable groups.

More specifically, we induced two sets of tutorial tactics: the Normalized Gain (NormGain) tactics were derived with the goal of making tutorial decisions that contribute to students' learning, while the Inverse Normalized Gain (InvNormGain) tactics were induced with the goal of making less beneficial, or possibly useless, decisions. The two sets were then compared by making all students studying the same materials and training on the Cordillera that covered the same subject matter and training problems, and used the same tutorial scripts and user interface. If our application of RL to induce pedagogical tutorial tactics is effective, then we expect that the NormGain students will outperform their InvNormGain peers. This would occur if the micro-level decisions on ET and JS impact learning. In the following, we will briefly describe how we applied RL to induce the pedagogical tutorial tactics and then describe our study and finally present our results.

2 Applying RL to Induce Pedagogical Tutorial Tactics

Previous research on using RL has typically used Markov Decision Processes (MDPs)[18]. MDP is a formal state model, commonly used to model dialogue data. Formally, an MDP is a 4-tuple (S, A, T, R), where: $S = \{S_1, \cdots, S_n\}$ is a state space; $A = \{A_1, \cdots, A_m\}$ is an action space represented by a set of action variables; $T : S \times A \times S \rightarrow$ *[0, 1]* is a set of transition probabilities between states describing the dynamics of the modeled system (e.g. $P(S_j|S_i, A_k)$ is the probability that the model will transition from state S_i to state S_j by taking action A_k); $R : S \times A \times S \rightarrow R$ is a reward model that assigns reward values to state transitions and models payoffs associated with the transitions. Finally, $\pi : S \rightarrow A$ is a policy.

The central idea behind this approach is to transform the problem of inducing effective pedagogical tactics into computing an optimal policy for choosing actions in an MDP. Inducing pedagogical tutorial tactics can be easily represented using an MDP: the states are vector representations composed of relevant student-tutor interaction characteristics; $A = \{Elicit, Tell\}$ for inducing ET policies and $\{Justify, Skip - Justify\}$ for inducing JS policies, and the reward function is calculated from the system's success measures and we used learning gains. Given that a student's learning gain will not be available until the entire tutorial dialogue is completed, only terminal dialogue state has non-zero reward. Once the S, A, R has been defined, the transition probabilities T are estimated from the training corpus, which is the collection of dialogues, as: $T = \{p(S_j|S_i, A_k)\}_{i,j=1,\cdots,n}^{k=1,\cdots,m}$. More specifically, $p(S_j|S_i, A_k)$ is calculated by taking the number of times that the dialogue is in state S_i, the tutor took action A_k, and the dialogue was next in state S_j divided by the number of times the dialogue was in S_i and the tutor took A_k. The reliability of these estimates clearly depends upon the size and structure of the training data. Once a complete MDP is constructed, a dynamic programming approach can be used to learn the optimal control policy π^* and here we used the toolkit developed by Tetreault and Litman[19]. The rest of this section presents a few critical details of the process, but many others must be omitted to save space.

In this study, the reward functions for inducing both the NormGain and the InvNormGain sets were based on Normalized Learning Gain (NLG). This is

because NLG measures a student's gain *irrespective of his/her incoming competence* and we have: $NLG = \frac{posttest - pretest}{1 - pretest}$. Here *posttest* and *pretest* refer to the students' test scores before and after the training respectively; and 1 is the maximum score. More specifically, the NormGain tutorial tactics induced by using the student's $NLG \times 100$ as the final reward while the InvNormGain ones was induced by using the student's $(1 - NLG) \times 100$ as the final reward. Apart from the reward functions, they were induced using the same general procedure.

2.1 Knowledge Component (KC) Based Pedagogical Strategies

In the learning literature, it is commonly assumed that relevant knowledge in domains such as math and science is structured as a set of independent but co-occurring Knowledge Components (KCs) and that these KCs are learned independently. A KC is "a generalization of everyday terms like concept, principle, fact, or skill, and cognitive science terms like schema, production rule, misconception, or facet"[23]. For the purpose of ITSs, these are the atomic units of knowledge. It is assumed that a tutorial dialogue about one KC will have no impact on the student's understanding of any other KCs. This is an idealization, but it has served ITS developers well for many decades, and is a fundamental assumption of many cognitive models[2, 16]. When dealing with a specific KC, the expectation is that the tutor's best policy for teaching that KC (e.g., to Elicit vs. to Tell) would be based upon the student's mastery of the KC in question, its intrinsic difficulty, and other relevant, but not necessarily known, factors specific to that KC. In other words, an optimal policy for one KC might not be optimal for another. Therefore, one assumption made in this paper is that inducing pedagogical policies specific to each KC would be more effective than inducing an overall KC-general policy.

The domain selected for this project is a subset of the physics work-energy domain, which is characterized by eight primary KCs. Given these independence assumptions, the problem of inducing a policy for ET decisions and a policy for JS decisions may be decomposed into 8 sub-problems of each type, one per KC. More specifically, in order to learn a policy for each KC, we annotated our tutoring dialogues and action decisions with the KCs covered by each action. For each KC, the final kappa was ≥ 0.77, fairly high given the complexity of the task. A domain expert also mapped the pre- and post-test problems to relevant KCs. This resulted in a KC-specific NLG score for each student. KC_1 does not arise in any JS decisions and thus only an ET policy was induced for it. For the remaining seven KCs, a pair of policies, one ET policy and one JS policy, were induced. So we induced 15 KC-based NormGain and 15 KC-based InvNormGain policies. There were some decision steps that did not involve any of the eight primary KCs. For them, two KC-general policies, an ET policy and a JS policy, were induced. To sum, a total of 17 NormGain and 17 InvNormGain policies were induced.

2.2 Inducing NormGain and InvNormGain Policies

In order to apply RL to induce pedagogical tutorial tactics, a training corpus is needed. In this study, we have three training corpora available from the previous

study[6]: the Exploratory corpus collected in 2007, the DichGain corpus collected in 2008, and a Combined training corpus. In order to examine a range of possible tactics, we included 50 features based upon six categories of features considered by previous research[15, 3, 11] to be relevant. We also used a different method of searching the power set of the 50 features and directly used the $NLG \times 100$ for inducing NormGain policies and$(1-NLG)\times 100$ for inducing InvNormGain ones instead of dichotomizing the NLGs when inducing DichGain policies previously.

Fig. 2 shows an example of a learned NormGain policy on KC_{20}, "Definition of Kinetic Enegy", for ET decisions. The policy involves three features:

[**StepDifficulty:**] encodes a step's difficulty level. Its value is estimated from the Combined Corpus based on the percentage of correct answers given on the step.

[**TutorConceptsToWords:**] which represents the ratio of the physics concepts to words in the tutor's dialogue. This feature also reflects how often the tutor has mentioned physics concepts overall.

[**TutorAverageWordsSession:**] The average number of words in the tutor's turn in this session. This feature reflects how verbose the tutor is in the current session.

[**Feature:**]
StepDifficulty: $[0, 0.38) \rightarrow 0; [0.38, 1] \rightarrow 1$
TutorConceptsToWords: $[0, 0.074) \rightarrow 0; [0.074, 1] \rightarrow 1$
TutorAverageWordsSession: $[0, 22.58) \rightarrow 0; [22.58, \infty) \rightarrow 1$
[**Policy:**]
Elicit: 0:0:0 0:0:1 1:0:1 1:1:0 1:1:1
Tell: 0:1:0
Else:0:1:1 1:0:0

Fig. 2: A NormGain Policy on KC_{20} For ET Decisions

MDP generally requires discrete features and thus all the continuous features need to be discretized. Fig. 2 listed how each of the three features was discretized. For example, For StepDifficulty, if its value is above 0.38, it is 1 otherwise, it is 0. There were 8 rules learned: in 5 situations the tutor should elicit, in one situation it should tell; in the remaining 2 cases either will do. For example, when all three features are zero (which means when the current step's difficulty level is low, the tutor ratio of physics concepts to words is low, and the tutor is not very wordy in the current session), then the tutor should elicit as 0:0:0 is listed next to the [elicit]. As you can see, three features already provide relatively complex tutorial tactics and the induced policies were not like most of the tutorial tactics derived from analyzing human tutorial dialogues.

The resulting NormGain and InvNormGain policies were then implemented back into Cordillera yielding two new versions of the system, named NormGain-Cordillera and InvNormGain-Cordillera respectively. The induced tutorial tactics were evaluated on real human subjects to see whether the NormGain students would out-perform the InvNormGain peers.

3 Methods

3.1 Participants

Data was collected over a period of two months during the summer of 2009. Participants were 64 college students who received payment for their participation. They were required to have a basic understanding of high-school algebra. However, they could not have taken any college-level physics courses. Students were randomly assigned to the two conditions. Each took from one to two weeks to complete the study over multiple sessions. In total, 57 students completed the study (29 in the NormGain group and 28 in the InvNormGain group).

3.2 Domain & Procedure

Our work used the Physics work-energy domain as covered in the first-year college physics course. The eight primary KCs were: the weight law (KC1), definition of work (KC14), Definition of Kinetic Energy (KC20), Gravitational Potential Energy (KC21), Spring Potential Energy (KC22), Total Mechanical Energy (KC24), Conservation of Total Mechanical Energy (KC27), and Change of Total Mechanical Energy (KC28).

All participants experienced the identical procedure and materials. More specifically, participants all completed 1) a background survey; 2) read a textbook covering the target domain knowledge; 3) took a pretest; 4) solved the same seven training problems in the same order on Cordillera; and 5) finally took a posttest. The pretest and posttest were identical. Except following the policies (NormGain vs. InvNormGain), the remaining components of Cordillera, including the GUI interface, the same training problems, and the tutorial scripts, were identical for all students.

3.3 Grading

All tests were graded in a double-blind manner by a single experienced grader. In a double-blind manner, neither the students nor the grader know who belongs to which group. For all identified relevant KCs in a test question, a KC-based score for each KC application was given. We evaluated the student's competence in the following sections based on the sum of these KC-based scores. This is because the KC-based pre- and post-test scores were used to define the reward functions when applying RL to induce policies. Later analysis showed that the same findings stand for other scoring rubrics. The tests contained 33 test items which covered 168 KC occurrences. For comparison purposes all test scores were normalized to fall in the range of [0,1].

4 Results

Random assignment appears to have balanced the incoming student competence across conditions. There were no statistically significant differences between the two conditions in the pre-test scores $t(55) = 0.71, p = .484$. Additionally, no

Table 1: NormGain vs. InvNormGain on Various Test Scores

	NormGain	InvNormGain	Stat	cohen d
Pretest	0.42 (0.16)	0.39 (0.23)	$t(55) = 0.71, p = .484$	0.15
Posttest	0.65 (0.15)	0.54 (0.20)	$t(55) = 2.32, p = 0.024$	$0.65 * *$
Adjusted Posttest	0.63 (.095)	0.55 (.095)	$F(1, 54) = 10.689, p = .002$	$0.86 * *$
NLG	0.41 (0.19)	0.25 (0.21)	$t(55) = 3.058, p = 0.003$	$0.81 * *$

significant differences were found between the two conditions on the mathSAT scores and the total training time spent on Cordillera: $t(39) = 0.536, p = 0.595$ and $t(55) = -.272, p = .787$ respectively.

A one-way ANOVA was used to test for learning performance differences between the pre- and posttests. Both conditions made significant gains from pre-test to post-test: $F(1, 56) = 31.34$, $p = .000$ for the NormGain condition and $F(1, 54) = 6.62$, $p = .013$ for the InvNormGain condition. Table 1 compares the pre-test, post-test, adjusted-post-test, and NLG scores between the two conditions. In Table 1, the Adjusted Post-test scores were compared between the two conditions by running an ANCOVA using the corresponding pre-test score as the covariate. The second and third columns in Table 1 list the means and SDs σ of the NormGain and InvNormGain groups' corresponding scores. The fourth column lists the corresponding statistical comparison and the fifth column lists the effect size of the comparison and we used Cohen's d. This is defined as the mean learning gain of the experimental group minus the mean learning gain of the control group, divided by the groups' pooled standard deviation. Table 1 shows that there was no significant difference between the two groups on pre-test scores. However, there were significant differences between the two groups on the post-test, adjusted-post-test, and NLG scores. Across all measurements, the NormGain group performed significantly better than the InvNormGain peers. The effect size, Cohen's d, was large.

To summarize, our results showed that both groups had significant learning gains after training on Cordillera. More importantly, although no significant difference was found in time on task and in the pre-test scores, the NormGain group out-performed the InvNormGain group on the post-test, adjusted post-test, and NLG scores regardless of the grading criteria. Therefore, the overall results show that the micro-level pedagogical tutorial decisions made a significant difference in the students' learning.

Later a post-hoc comparison was done across the NormGain, Exploratory and DichGain groups because the NormGain policies were induced from the Exploratory and DichGain corpora. Despite the lack of random assignments, no significant difference was found among the three groups in the pretest. However, the NormGain group significantly outperformed both groups in the posttest, adjusted posttest scores, and NLGs[5]. Similarly, a post-hoc comparison was done across the InvNormGain, Exploratory and DichGain groups but no difference was found among the three groups on pretest, posttest, adjusted posttest scores or NLGs. The lack of a significant difference among the InvNormGain, DichGain,

and Exploratory groups seemingly contradicts the initial predictions since the InvNormGain strategies were specifically induced to enhance those decisions that contribute less or even none to the students' learning. Therefore, a lower performance on the students' part there than in at least the DichGain group, which sought to enhance the tutorial decisions that contribute to the students' learning, was expected. One possible explanation for the lack of difference among the three groups is that the tutorial tactics employed by the DichGain- and Random-Cordillera systems were ineffective and thus presented a minimum bar. By 'ineffective' it does not mean that they prevented the students from learning but rather that they were not able to make a positive impact on their learning above and beyond the baseline provided by Cordillera itself. Here the basic practices and problems, domain exposure, and interactivity of Cordillera set a minimum bar of students' learning that the tactics, however poor, cannot prevent. This is only a post-hoc explanation not a tested hypothesis, however it merits further investigation.

5 Conclusion

In this study, students were randomly assigned to balanced conditions and received identical training materials and procedures apart from the tutoring tactics employed. After spending the same amount of time on training, the NormGain group outperformed the InvNormGain group in terms of posttest scores, the adjusted post-test scores and the normalized learning gains. This results support the hypothesis that micro-step interactive tutorial decisions such as the Elicit vs. Tell and Justify vs. Skip-justify decisions do affect students' learning. Therefore, future work is needed to investigate the induced NormGain and InvNormGain tutorial tactics and find out what actually caused these learning differences.

Moreover, this study also suggests that RL is a feasible approach for inducing pedagogical policies by using a relatively small human interaction corpus. However, it is not trivial. The DichGain tutorial tactics, for example, did not seem to be more effective than the random decisions in Random-Cordillera. As future work, we would like to explore the use of richer POMDP models, and do additional empirical evaluation of the RL approach.

Finally, this study suggests that the fine-grained interaction (micro-steps) of human tutoring are a potential source of pedagogical power, but human tutors may not be particularly skilled at choosing the right micro-steps. Given how much computation we had to perform in order to learn which micro-steps were best, it is hardly surprising that human tutors have not (yet) acquired similar skill. This raises an interesting question: Can a NL tutoring system that is extensively trained be significantly more effective than expert human tutors? This would be an excellent question for future research.

Acknowledgments NSF (#0325054) supported this work and NSF (#SBE-0836012) supported its publication. We thank Learning Research Development Center for providing all the facilities for this work. We also thank Collin Lynch and the reviewers for helpful comments.

References

[1] V. Aleven, A. Ogan, et al. Evaluating the effectiveness of a tutorial dialogue system for self-explanation. In *Intelligent Tutoring Systems*, vol. 3220 of *LNCS*, pp. 443–454. Springer, 2004.

[2] J. R. Anderson. *The architecture of cognition.* Cambridge, Mass. : Harvard University Press, 1983.

[3] J. Beck, B. P. Woolf, et al. Advisor: A machine learning architecture for intelligent tutor construction. In *AAAI*, pp. 552–557. AAAI Press, 2000.

[4] B. S. Bloom. The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher*, 13:4–16, 1984.

[5] M. Chi. *Do Micro-Level Tutorial Decisions Matter: Applying Reinforcement Learning To Induce Pedagogical Tutorial Tactics.* Ph.D. thesis, School of Art & Science University of Pittsburgh, Dec. 2009.

[6] M. Chi, P. W. Jordan, et al. To elicit or to tell: Does it matter? In V. Dimitrova, R. Mizoguchi, et al., eds., *AIED*, pp. 197–204. IOS Press, 2009. ISBN 978-1-60750-028-5.

[7] M. T. H. Chi, N. de Leeuw, et al. Eliciting self-explanations improves understanding. *Cognitive Science*, 18(3):439–477, 1994.

[8] M. T. H. Chi, S. Siler, et al. Can tutors monitor students' understanding accurately? *Cognition and Instruction*, 22(3):363–387, 2004.

[9] A. Collins, J. S. Brown, et al. Cognitive apprenticeship: Teaching the craft of reading, writing and mathematics. In L. B. Resnick, ed., *Knowing, learning and instruction: Essays in honor of Robert Glaser*, chap. 14, pp. 453–494. 1989.

[10] M. Evens and J. Michael. *One-on-one Tutoring By Humans and Machines.* Mahwah, NJ: Erlbaum, 2006.

[11] K. Forbes-Riley, D. J. Litman, et al. Comparing linguistic features for modeling learning in computer tutoring. In *AIED*, vol. 158, pp. 270–277. IOS Press, 2007.

[12] A. C. Graesser, N. Person, et al. Collaborative dialog patterns in naturalistic one-on-one tutoring. *Applied Cognitive Psychology*, 9:359–387, 1995.

[13] A. C. Graesser, K. VanLehn, et al. Intelligent tutoring systems with conversational dialogue. *AI Magazine*, 22(4):39–52, 2001.

[14] S. Katz, G. O'Donnell, et al. An approach to analyzing the role and structure of reflective dialogue. *International Journal of AI and Education*, 11:320–343, 2000.

[15] J. D. Moore, K. Porayska-Pomsta, et al. Generating tutorial feedback with affect. In V. Barr and Z. Markov, eds., *FLAIRS Conference*. AAAI Press, 2004.

[16] A. Newell, ed. *Unified Theories of Cognition.* Harvard University Press; Reprint edition, 1994.

[17] R. T. Putnam. Structuring and adjusting content for students: A study of live and simulated tutoring of addition. *American Educational Research Journal*, 24(1):13–48, 1987.

[18] R. S. Sutton and A. G. Barto. *Reinforcement Learning.* MIT Press Bradford Books, 1998.

[19] J. R. Tetreault and D. J. Litman. A reinforcement learning approach to evaluating state representations in spoken dialogue systems. *Speech Communication*, 50(8-9):683–696, 2008.

[20] K. VanLehn. The behavior of tutoring systems. *International Journal AI in Education*, 16(3):227–265, 2006.

[21] —. The interaction plateau: Answer-based tutoring < step-based tutoring = natural tutoring. In *ITS*, vol. 5091 of *LNCS*, p. 7. Springer, 2008.

[22] K. VanLehn, A. C. Graesser, et al. When are tutorial dialogues more effective than reading? *Cognitive Science*, 31(1):3–62, 2007.

[23] K. VanLehn, P. Jordan, et al. Developing pedagogically effective tutorial dialogue tactics: Experiments and a testbed. In *Proc. of SLaTE Workshop on Speech and Language Technology in Education ISCA Tutorial and Research Workshop.* 2007.