The Dialog Advancer Network: A Conversation Manager for AutoTutor

Natalie K. Person[1], Laura Bautista[2], Roger J. Kreuz[2], Arthur C. Graesser[2], and the Tutoring Research Group

[1]Department of Psychology, Rhodes College, 2000 N. Parkway, Memphis, TN 38112
person@rhodes.edu
[2]Department of Psychology, University of Memphis, Memphis, TN 38152
a-graesser@memphis.edu, lbautist@memphis.edu, kreuzrj@mocha.memphis.edu
http://mnemosyne.csl.psyc.memphis.edu/home/graesser/
www.psyc.memphis.edu/faculty/kreuz/kreuz.htm

Abstract: The Dialog Advancer Network (DAN) is a mechanism that manages the conversation that occurs between a learner and a pedagogical agent. The DAN is currently being implemented in AutoTutor, a pedagogical agent that participates in conversations with students learning about introductory computer literacy topics. This paper includes an overview of AutoTutor and the DAN along with excerpts from pre- and post-DAN conversations. Data that describe AutoTutor's current conversational habits are provided along with proposed changes that will enable AutoTutor to more fully exploit the dialog move pathways within the DAN.

1 Background

Human one-to-one tutoring is second to no other instructional method in yielding positive student learning gains. This particular claim has been supported in numerous research studies and is not particularly controversial. However, when the tutors of typical tutoring situations are considered, this claim becomes somewhat perplexing. Most tutors in school settings are older students, parent volunteers, or teachers' aides that possess some knowledge about particular topic domains and virtually no knowledge about expert tutoring techniques. Given their limited knowledge, it is somewhat impressive that these untrained tutors are responsible for the considerable learning gains that have been reported in the tutoring literature. Effect sizes ranging from .5 to 2.3 standard deviations have been reported for untrained tutors versus other comparable learning conditions [1, 3].

In order to identify the mechanisms that produce such positive learning gains, several members of the Tutoring Research Group (TRG) extensively analyzed a large corpus of tutoring interactions that occurred between untrained human tutors and students [7, 8, 21, 24, 26]. One reoccurring finding in many of our analyses is that untrained, human tutors rarely adhere to sophisticated or ideal tutoring models (e.g., Socratic tutoring, reciprocal teaching, anchored learning) that have been advocated by education and ITS researchers. Instead, untrained human tutors tend to rely on pedagogically effective dialog moves that are embedded within the conversational turns of the tutorial dialog. More specifically, human tutors generate dialog moves that are sensitive to the quality and quantity of the preceding student turn. The tutor dialog move categories that we identified in human tutoring sessions are provided below.

(1) Positive immediate feedback. "That's right" "Yeah"
(2) Neutral immediate feedback. "Okay" "Uh-huh"
(3) Negative immediate feedback. "Not quite" "No"
(4) Pumping for more information. "Uh-huh" "What else"
(5) Prompting for specific information. "The primary memories of the CPU are ROM and _____"
(6) Hinting. "What about the hard disk?"
(7) Elaborating. "CD ROM is another storage medium."
(8) Splicing in the correct content after a student error.
(9) Summarizing. "So to recap," <succinct recap of answer to question>

After spending nearly a decade toiling over thousands of pages of tutoring transcripts and staring pie-eyed at hundreds of hours of videotaped tutoring sessions, we decided to it was time to put our knowledge to good use and build something. We built AutoTutor.

2 What is AutoTutor?

AutoTutor is an animated pedagogical agent that engages in a conversation with the learner while simulating the dialog moves of untrained human tutors. AutoTutor is currently designed to help college students learn about topics that are typically covered in an introductory computer literacy course (e.g., hardware, operating systems, the Internet). AutoTutor's architecture is comprised of five major modules: (1) an animated agent, (2) a curriculum script, (3) language analyzers, (4) latent semantic analysis (LSA), and (5) a dialog move generator. All of these modules have been discussed rather extensively in previous publications [see 5, 6, 9, 12, 14, 17, 20, 23, 25, 29]; and therefore, will only be mentioned briefly in this paper.

AutoTutor was created with Microsoft Agent. He is a two-dimensional embodied agent that remains on the screen during the entire tutoring session. The material that AutoTutor covers in the tutoring session is organized in a curriculum script. A curriculum script is a well-defined, loosely structured lesson plan that includes important concepts, questions, cases, and problems that teachers and tutors wish to cover in a particular lesson [7, 8, 16, 27]. AutoTutor's curriculum script includes 36 computer literacy questions/problems along with corresponding "ideal answers" (which are comprised of many potential good answers), anticipated bad answers, corrective splices (i.e., correct answers) for anticipated bad answers, and all of AutoTutor's dialog moves that contain information about the question/problem (i.e., elaborations, hints, prompts, prompt responses, and summaries).

AutoTutor begins the tutoring session with a brief introduction and then asks the student a question from the curriculum script. The student responds to the question by typing her/his answer on the keyboard and hitting the "Enter" key. A number of language analyzers operate on the words in the student's contribution. These analyzers include a word and punctuation segmenter, syntactical class sorter, and a speech act classifier. The speech act classifier assigns the student's input into one of five speech act categories: Assertion, WH-question, Yes/No question, Short Response, or Prompt Completion. These speech act categories enable AutoTutor to sustain mixed-initiative dialog as well as dictate the legal DAN pathways that AutoTutor may pursue. The DAN pathways will be discussed in the next section.

AutoTutor's knowledge about computer literacy is represented by Latent Semantic Analysis (LSA) [4, 5, 14, 15]. LSA is a statistical technique that measures the conceptual similarity of two text sources. LSA computes a geometric cosine (ranging from 0 to 1) that represents the conceptual similarity between the two text sources. In AutoTutor, LSA is used to assess the quality of student Assertions and to monitor other informative parameters such as Topic Coverage and Student Ability Level. Student Assertion quality is measured by comparing each Assertion against two other computer literacy text sources, one that contains potential good answers to the topic being discussed and one that contains the anticipated bad answers. The higher of the two geometric cosines is considered the best conceptual match, and therefore, determines how AutoTutor responds to the student Assertion. For the domain of computer literacy, we have found our application of LSA to be quite accurate in evaluating the quality of learner Assertions [9, 30].

AutoTutor's dialog move generator is controlled by 15 fuzzy production rules [13] that primarily exploit data provided by the LSA module. Each fuzzy production rule specifies the parameter values in which a particular dialog move should be generated. For example, consider the following dialog move rules:

(1) IF [Assertion match with good answer text = HIGH or VERY HIGH]
THEN [select POSITIVE FEEDBACK]

(2) IF [student ability = MEDIUM or HIGH & Assertion match with
good answer text = LOW] THEN [select HINT]

In Rule (1) AutoTutor will provide Positive Feedback (e.g., "Right") in response to a high quality student Assertion, whereas in Rule (2) AutoTutor will generate a Hint to bring the relatively high ability student back on track (e.g., "What about the size of the programs you need to run?"). The dialog move generator currently controls 12 dialog moves: Pump, Hint, Splice, Prompt, Prompt Response, Elaboration, Summary, and five forms of immediate short-feedback (positive, positive-neutral, neutral, negative-neutral, and negative).

For each of the 36 questions and problems, the student and AutoTutor collaboratively improve the quality of the student's contributions while participating in a conversation. Once AutoTutor is "convinced" that all or enough of the critical components for a particular question/ problem have been covered, the conversation proceeds to the next question or problem in the curriculum script.

3 Managing AutoTutor's Conversations with the DAN

Before the DAN. Our goal from the onset of the AutoTutor project has been to develop an agent that simulates the dialog moves of untrained tutors while participating in a conversation with the learner. Needless to say, we have encountered a number of problems along the way. After all, having knowledge about when AutoTutor should generate particular dialog moves does not guarantee that a smooth conversation will transpire. In fact, quite the opposite occurred in some of AutoTutor's initial tutoring sessions with students. Consider the following dialog excerpts. In Excerpts 1 and 2, the topic is parts and uses of a computer. In Excerpt 3, AutoTutor is discussing the features of RAM. AutoTutor's dialog moves are labeled in parentheses and the students' utterances that were directed to the human lab assistant are included in brackets.

Excerpt 1
1.1 AutoTutor (Prompt): See if you can get this. Computers can be used to analyze data by finding averages, standard deviations, or performing many other kinds of . . .?
1.2 Student to lab assistant: [Oh, just answer the question? Um. So, am I still supposed to answer that question (pointing to the main topic question that remains on the monitor)?]
1.3 Student types: i don't know
1.4 AutoTutor (Prompt Response): kinds of statistics.
1.5 Student to lab assistant: [Okay, this is really confusing. I don't understand what the question is supposed to be.]
1.6 Student types: i don't know

Excerpt 2
2.1 AutoTutor (Elaboration): Also, a computer has a monitor.
2.2 Student to lab assistant: [I just answer yes or no?]
2.3 Student types: yes

Excerpt 3
3.1 AutoTutor (Elaboration): And, the CPU uses RAM as a short-term memory.
3.2 Student to lab assistant: [Should I, like, I can just say how I think he's unintelligible?]
3.3 Student types: i am not sure
3.4 AutoTutor (Neutral Feedback): Uh-huh
3.5 AutoTutor (Elaboration): Also, RAM's contents disappear when the computer is turned off.
3.6 Student types: what?
3.7 AutoTutor (Neutral Feedback): Uh-huh
3.8 AutoTutor (Hint): So, how does the CPU get information from RAM?
3.9 Student types: I am not sure
3.10 AutoTutor (Neutral Feedback): Uh-huh
3.11 AutoTutor (Summary): So, RAM is primary memory storage and is changeable. It is used by the CPU as short-term memory. The CPU both writes to and reads from RAM. The contents of RAM disappear when the computer is turned off.

These three excerpts highlight several of AutoTutor's original conversational deficits. One glaring deficit involves the students' uncertainty about when it is their turn to respond to AutoTutor. Turn-taking is an integral feature of the conversational process. To facilitate the turn-taking process in human-to-human conversations, speakers signal to listeners that they are relinquishing the floor (i.e., it is the listener's turn to say something) [2, 10, 11, 18, 19, 28,]. However, human-to-computer conversations lack many of the subtle signals inherent to human conversations. When conversational agents like AutoTutor lack turn-taking signals, computer users (in our case, students) often do not know when or if they are supposed to respond. In conversations with AutoTutor, students were frequently confused after AutoTutor's Elaborations, Prompt Responses, and assertion-form Hints (some Hints were in question-form and were not problematic for students).

Another obvious deficit is that AutoTutor's dialog moves are not well adapted to the students' turns. For example, in Excerpt 3, AutoTutor's dialog moves are clearly not sensitive to the content of the student's turns. Participants engaged in human-to-human conversations, however, are able to adapt each conversational turn so that it relates in some way to the turn of the previous speaker. This micro-adaptation process is somewhat problematic for AutoTutor because the content of AutoTutor's dialog moves is determined *a priori*. That is, AutoTutor doesn't generate the content of his dialog moves on the fly but rather selects each dialog move from a scripted set of moves that is related to the tutoring topic being discussed. Hence, we recognized early on that AutoTutor needed a mechanism that would allow him to make quasi-customized dialog moves given his limited number of dialog move options.

After the DAN. In order to rectify many of AutoTutor's turn-taking and micro-adaptation problems, we created the Dialog Advancer Network (DAN) [22]. The DAN is a mechanism that manages the conversation that occurs between a student and AutoTutor. A simplified version of the DAN is provided in Figure 1. The DAN has improved AutoTutor's micro-adaptation capabilities by providing customized pathways that are tailored to particular student turn categories. For example, if a student wants AutoTutor to repeat the last dialog move, the DAN contains a Short Response pathway that allows AutoTutor to adapt to the student's request and respond appropriately. A DAN pathway may include one or a combination of the following components: (1) discourse markers (e.g., "Okay" or "Moving on"), (2) AutoTutor dialog moves (e.g., Positive Feedback, Pump, or Elaboration), (3) answers to WH- or Yes/No questions, or (4) canned expressions (e.g., "That's a good question, but I can't answer that right now").

The DAN also solved practically all of AutoTutor's previous turn-taking problems. Most of the turn-taking confusion was eliminated by the Advancer States that occur in many of the DAN pathways. Advancer States are designed to advance the conversational dialog and to disambiguate who has the floor in the conversation. For example, prior to implementing the DAN, the conversation often stopped after AutoTutor delivered an Elaboration, Hint, or Prompt Response dialog move because the student did not know what to do. In the current version of AutoTutor, students are no longer confused about whether they should respond because each of these dialog moves is followed by an Advancer State that requires AutoTutor to keep the floor. For example, AutoTutor keeps the floor after an Elaboration by articulating a predetermined discourse marker (e.g., "Moving on") and selecting another dialog move (see Figure 1).

Two conversations that are managed by the DAN are provided in Excerpts 4 and 5. These excerpts illustrate how the Advancer States advance the dialog and indicate to the student that AutoTutor is keeping the floor in the conversation. In Excerpt 4, AutoTutor and the student are discussing the advantages of RISC processors; in Excerpt 5 they are discussing how typed information gets passed from the keyboard to the hard disk.

Excerpt 4

4.1 AutoTutor (Prompt): Now, what kind of architecture allows a RISC processor to execute multiple instructions simultaneously?
4.2 Student types: simple
4.3 AutoTutor (Prompt Response): A superscalar architecture.
4.4 AutoTutor (Dialog Advancer): Okay,
4.5 AutoTutor (Elaborate): And, RISC processors have fewer bottlenecks that complicate processing.
4.6 AutoTutor (Dialog Advancer) Okay,
4.7 AutoTutor (Summary): Now to wrap up, RISC processors are faster and more efficient than CISC processors (AutoTutor continues to summarize this topic).

Excerpt 5

5.1 AutoTutor (Prompt): Well, where is most of the information you type in temporarily stored?
5.2 Student types: RAM
5.3 AutoTutor (Positive Feedback): Right!
5.4 AutoTutor (Prompt Response): In RAM,
5.5 AutoTutor (Dialog Advancer): Moving on,
5.6 AutoTutor (Elaborate): The CPU handles one instruction at a time.
5.7 AutoTutor (Dialog Advancer): Okay,
5.8 AutoTutor (Summary): Let’s review, after you enter information, it is sent to the CPU. The CPU carries out the instructions on the data. (AutoTutor continues to summarize this topic).

Since the implementation of the DAN, AutoTutor’s interactions with students have improved considerably. The numerous pathways within the DAN have refined AutoTutor’s micro-adaptation skills and the DAN Advancer States have eradicated much of the turn-taking confusion. Although the DAN is a relatively new feature of AutoTutor, it has already proven to be quite instrumental in helping us improve AutoTutor’s overall effectiveness as a tutor and as a conversational partner. In a recent analysis, we examined how AutoTutor utilized the DAN while interacting with students. More specifically, we wanted to document AutoTutor’s DAN pathway choice for each student turn. Sixty-four students enrolled in a computer literacy course agreed to interact with AutoTutor in exchange for course credit. AutoTutor covered 24 computer literacy topics during each of the tutoring sessions and written transcripts were generated for all of the sessions. Three of the 24 computer literacy topics were randomly selected from each of the 64 transcripts. Thus, 192 mini-conversations were included in the DAN analysis.

The frequency distribution for the most well-traveled pathways is provided in Table 1. All pathways with frequencies lower than 10 are not included in the table. We were somewhat encouraged in that AutoTutor utilized 30 of the 78 legal DAN pathways and that there were no instances of illegal pathways after student turns. It is clearly the case, however, that the current version of AutoTutor is not maximizing the DAN to its

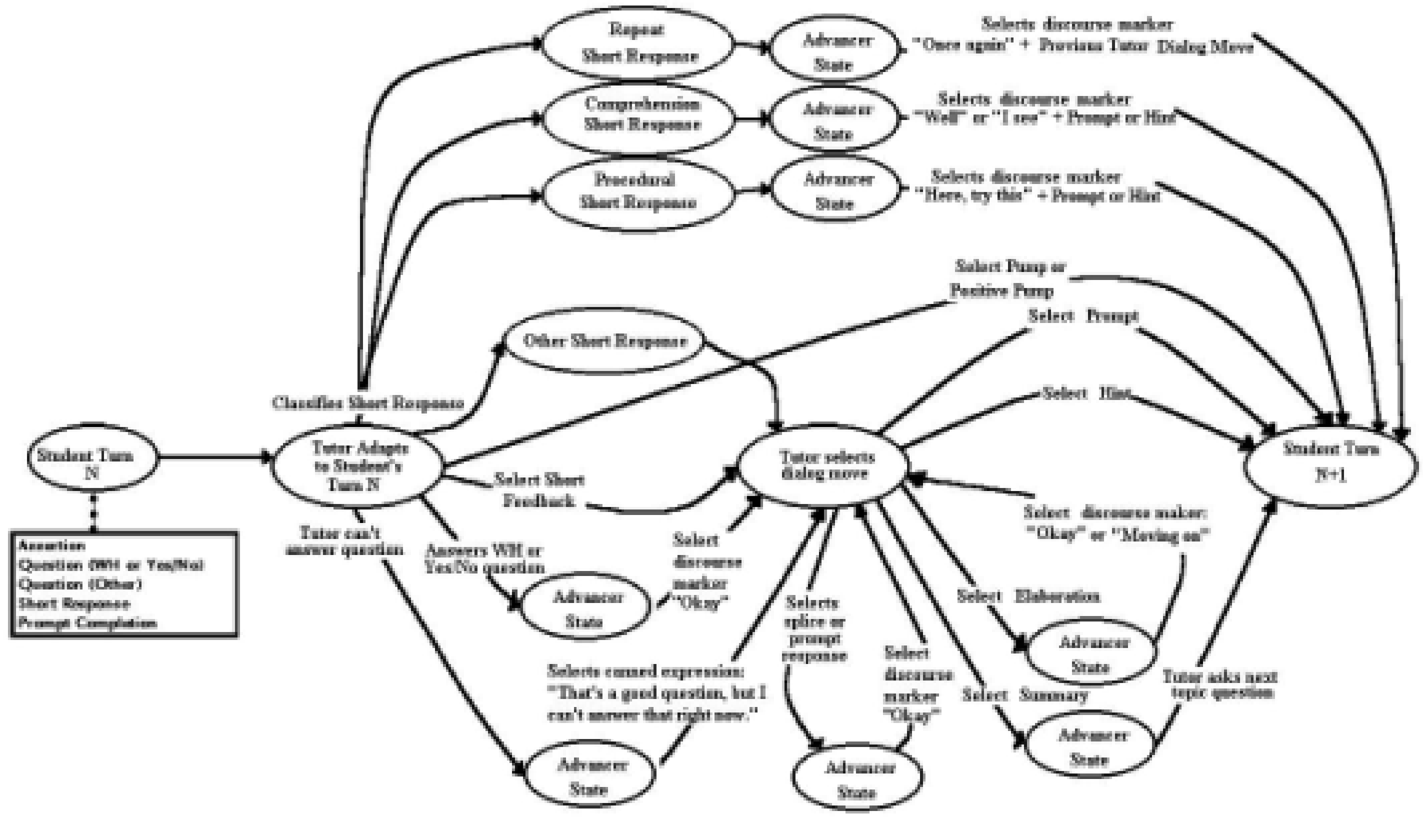

Figure 1. Dialogue Advancer Network (DAN)

full potential and that adjustments need to be made to break some of AutoTutor's poor conversational habits. To address every AutoTutor problem that the DAN analysis has elucidated is clearly beyond the scope of this paper. Thus, we have chosen to discuss only two, the Prompt problem and the Feedback Problem.

The two most frequently traveled DAN pathways were the *Prompt Response* → *Advancer* → *Prompt* pathway (215 occurrences) and the *Positive Feedback* → *Prompt Response* → *Advancer* → *Prompt* pathway (179 occurrences). These two pathways alone comprised roughly 35% of all of the chosen pathways. In addition, 245 of the remaining pathways also ended in a Prompt. Hence, approximately 56% of the pathways ended with AutoTutor Prompting the student. This is problematic for two reasons. First, prompting a student at these rates is pedagogically undesirable. Human tutors usually reserve Prompts for medium to low ability students who are reluctant to provide any information. By relying of Prompts so often, AutoTutor is not giving students the opportunity to elaborate their knowledge about the topics. Second, frequent prompting thwarts the conversational nature of AutoTutor that we are trying to promote. Prompts only require one or two word responses from students rather lengthier contributions that frequently occur in conversations. We are attempting to fix the Prompt problem by altering some of the pathways in the DAN and by modifying the fuzzy production rules that generate Prompts.

Table 1. Frequency Distribution of DAN Pathways Chosen by AutoTutor

DAN Pathway	*f*
Prompt Response → Advancer → Prompt	215
Positive Feedback → Prompt Response → Advancer → Prompt	179
Pump	169
Comprehension Short Response Advancer → Prompt	133
Repeat Short Response Advancer → Advancer	81
Neutral Feedback → Prompt	79
Prompt Response → Advancer → Summary	56
Positive Feedback → Prompt Response → Advancer → Summary	46
Prompt Response → Advancer → Elaboration → Advancer → Summary	37
Neutral Feedback → Hint	32
Positive Feedback → Prompt Response → Advancer → Elaboration → Advancer → Summary	26
Positive Feedback → Prompt Response → Advancer → Elaboration → Advancer → Prompt	10
Prompt Response → Advancer → Elaboration → Advancer → Prompt	10
*Total pathways	1134

*All pathways with frequencies below 10 are not included in the table.

AutoTutor also has a problem with some of the short-immediate feedback pathways. Specifically, AutoTutor never selects the DAN pathways that include three of the short feedback categories (i.e., positive-neutral, negative-neutral, and negative). It would be wonderful if AutoTutor chose to ignore these pathways because all of the student Assertions were high in quality. Unfortunately, this was not the case. Many of the student Assertions contained misconceptions and a number of others were only partially correct. We are addressing the Feedback problem by revising some of the dialog move productions rules and by adjusting some of the LSA values that are fed to the dialog move generator.

The DAN will undoubtedly continue to provide us with insights that will enable us to make improvements to AutoTutor. A number of other problems that were illuminated by the DAN analysis, but not mentioned is the paper, are currently being addressed by the Tutoring Research Group. We recognize that conversations with AutoTutor will probably never possess the dynamic and spontaneous features of human-to-human conversations. However, we do believe that AutoTutor is on his way to being an effective tutor and adequate conversational partner.

Acknowledgements

This research was funded by the National Science Foundation (SBR 9720314) in a grant awarded to the Tutoring Research Group (Art Graesser, Principal Investigator).

References

1. Bloom, B. S.: The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. Educational Researcher **13** (1984) 4-16
2. Clark, H. H., Schaefer, E. F.: Collaborating on contributions to conversations. Language and Cognitive Processes **2** (1987) 19-41
3. Cohen, P. A., Kulik, J. A., Kulik, C. C.: Educational outcomes of tutoring: A meta-analysis of findings. American Educational Research Journal **19** (1982) 237-248
4. Foltz, P.W.: Latent semantic analysis for text-based research. Behavior Research Methods, Instruments, and Computers **28** (1996) 197-202
5. Foltz, P. W., Britt, M. A., Perfetti, C. A.: Reasoning from multiple texts: An automatic analysis of readers' situation models. Proceedings of the 18th Annual Conference of the Cognitive Science Society. Lawrence Erlbaum and Associates, Mahwah, NJ (1996) 110-115
6. Graesser, A.C., Franklin, S., Wiemer-Hastings, P.: Simulating smooth tutorial dialogue with pedagogical value. Proceedings of the American Association for Artificial Intelligence. AAAI Press, Menlo Park, CA (1998) 163-167
7. Graesser, A. C., Person, N. K.: Question asking during tutoring. American Educational Research Journal **31** (1994) 104-137
8. Graesser, A. C., Person, N. K., Magliano, J. P.: Collaborative dialog patterns in naturalistic one-on-one tutoring. Applied Cognitive Psychology **9** (1995) 359-387
9. Graesser, A. C., Wiemer-Hastings, P., Wiemer-Hastings, K., Harter, D., Person, N. K., the TRG: Using latent semantic analysis to evaluate the contributions of students in AutoTutor. Interacting Learning Environments (in press)
10. Grice, H. P.: Logic and conversation. In Cole, P., Morgan, J. (eds.): Syntax and Semantics, Vol. 3: Speech acts. Academic Press, New York (1975)
11. Hobbs, J. R.: Coherence and coreference. Cognitive Science **3** (1979) 67-90
12. Hu, X., Graesser, A. C., the TRG: Using WordNet and latent semantic analysis to evaluate the conversational contributions of learners in the tutorial dialog. Proceedings of the International Conference on Computers in Education, Vol. 2. Springer, Beijing, China (1998) 337-341
13. Kosko, B.: Neural Networks and Fuzzy Systems. Prentiss Hall, New York (1992)
14. Landauer, T. K., Dumais, S. T.: A solution to Plato's problem: The latent semantic analysis of theory acquisition, induction, and representation of knowledge. Psychological Review **104** (1997) 211-240

15. Landauer, T. K., Foltz, P. W., Laham, D.: An introduction to latent semantic analysis. Discourse Processes **25** (1998) 259-284
16. McArthur, D., Stasz, C., Zmuidzinas, M.: Tutoring techniques in algebra. Cognition and Instruction **7** (1990) 197-244
17. McCauley, L., Gholson, B., Hu, X., Graesser, A. C., the TRG: Delivering smooth tutorial dialogue using a talking head. Proceedings of the Workshop on Embodied Conversation Characters. AAAI and ACM, Tahoe City, CA (1998) 31-38
18. McLaughlin, M. L.: Conversation: How talk is organized. Sage, Beverly Hills, CA (1984)
19. Nofsinger, R. E.: Everyday Conversation. Sage, Newbury Park, CA (1991)
20. Olde, B. A., Hoeffner, J., Chipman, P., Graesser, A. C., the Tutoring Research Group: A connectionist model for part of speech tagging. Proceedings of the American Association for Artificial Intelligence. AAAI Press, Menlo Park, CA (1999) 172-176
21. Person, N. K., Graesser, A. C.: Evolution of discourse in cross-age tutoring. In O'Donnell, A.M., King, A. (eds.): Cognitive Perspectives on Peer Learning. Erlbaum, Mahwah, NJ (1999) 69-86
22. Person, N. K., Graesser, A. C., the TRG: Designing AutoTutor to be an effective conversational partner. To appear in the Proceedings for the 4th International Conference of the Learning Sciences. Ann Arbor, MI (in press)
23. Person, N. K., Graesser, A. C., Kreuz, R. J., Pomeroy, V., the Tutoring Research Group: Simulating human tutor dialog moves in AutoTutor. To appear in the International Journal of Artificial Intelligence in Education (in press)
24. Person, N. K., Graesser, A. C., Magliano, J. P., Kreuz, R. J.: Inferring what the student knows in one-to-one tutoring: The role of student questions and answers. Learning and Individual Differences **6** (1994) 205-29
25. Person, N. K., Klettke, B., Link, K., Kreuz, R. J., the Tutoring Research Group: The integration of affective responses into AutoTutor. Proceedings of the International Workshop on Affect in Interactions. Siena, Italy (1999) 167-178
26. Person, N. K., Kreuz, R. J., Zwaan, R., Graesser, A. C.: Pragmatics and pedagogy: Conversational rules and politeness strategies may inhibit effective tutoring. Cognition and Instruction **13** (1995) 161-188
27. Putnam, R. T.: Structuring and adjusting content for students: A study of live and simulated tutoring of addition. American Educational Research Journal **24** (1987) 13-48
28. Sacks, H., Schegloff, E. A., Jefferson, G.: A simplest systematics for the organization of turn taking for conversation. In Schenkein, J. (ed.): Studies in the Organization of Conversational Interaction. Academic Press, New York (1978)
29. Wiemer-Hastings, P., Graesser, A. C., Harter, D., the Tutoring Research Group: The foundations and architecture of AutoTutor. Proceedings of the 4th International Conference on Intelligent Tutoring Systems. Springer-Verlag, Berlin Heidelberg New York (1998) 334-343
30. Wiemer-Hastings, P., Wiemer-Hastings, K., Graesser, A. C.: Improving an intelligent tutor's comprehension of students with Latent Semantic Analysis. Artificial Intelligence in Education (1999) 535-542