Designing AutoTutor to be an Effective Conversational Partner

Natalie K. Person[1], Arthur C. Graesser[2] and the Tutoring Research Group[2]
[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
http://mnemosyne.csl.psyc.memphis.edu/home/graesser/

Abstract: AutoTutor is an automated computer literacy tutor that participates in a conversation with the student. This paper describes the conversational features that have been incorporated into AutoTutor to improve his performance as a conversational partner. We depict AutoTutor's dialog move options in the Dialog Advancer Network (DAN). The DAN illustrates how AutoTutor adapts his dialog moves to the preceding student turn as well as how AutoTutor gives the floor back to the student. We also discuss how AutoTutor conforms to conversational norms, provides appropriate emotional responses, and appears interested in the student's conversational input.

Keywords: tutoring, discourse, artificial intelligence, pedagogy, modeling, cognitive science

Introduction

AutoTutor is an animated pedagogical agent that serves as a conversational partner with the student. AutoTutor is a working system that responds to students' natural language contributions by simulating the dialog moves of normal (not expert) human tutors. The creation of AutoTutor was inspired by numerous studies that have systematically analyzed the collaborative discourse that occurs between human tutors and students (Fox, 1993; Graesser & Person, 1994; Graesser, Person, & Magliano, 1995; Hume, Michael, Rovick, & Evens, 1996; McArthur, Stasz, & Zmuidzinas, 1990; Merrill, Reiser, Ranney, & Trafton, 1992; Moore, 1995; Person & Graesser, 1999; Person, Graesser, Magliano, & Kreuz, 1994; Person, Kreuz, Zwaan, & Graesser, 1995; Putnam, 1987). One reoccurring finding in several of these studies is that human tutors rarely adhere to "ideal" tutoring models that are often integrated into intelligent tutoring systems. Instead, human tutors tend to rely on pedagogically effective strategies that are embedded within the conversational turns of the tutorial dialog.

Our primary goal from the onset of the AutoTutor project has been to build an agent that delivers pedagogically effective dialog moves while participating in a conversation with the learner. Not surprisingly, building a knowledgeable conversational partner has presented us with a number of formidable challenges. For example, participants engaged in human-to-human conversations are able to customize each conversational turn so that it relates in some way to the turn of the previous speaker. The content of AutoTutor's dialog moves, however, 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. Therefore, we have had to equip AutoTutor with mechanisms that allow him to make quasi-customized dialog moves given his limited number of dialog move options. We recognize that AutoTutor's conversational dialogs will probably never possess the dynamic and spontaneous features of human-to-human conversations. However, we do believe that AutoTutor's conversational skills are superior to those of other pedagogical agents. That is, AutoTutor can respond to *any* natural language student contribution in a conversationally (and pedagogically) appropriate way (data to support this claim are reported in Person, Graesser, Kreuz, Pomeroy, & the TRG, 1999). The aim of this paper is to highlight AutoTutor's conversational features that facilitate collaborative tutorial dialogs.

A Brief Description of AutoTutor

AutoTutor is a fully automated computer tutor that helps college students learn about introductory computer literacy topics. As mentioned earlier, AutoTutor responds to student input by simulating the dialog moves that are frequently employed by normal, unskilled human tutors. We chose to simulate normal human tutors (rather than expert tutors) for two reasons. First, normal, untrained tutors are the norm in most school systems. Most tutors are older students, parent volunteers, or teacher's aides that possess some knowledge about particular topic domains and virtually no knowledge about expert tutoring techniques. Second, and more importantly, these untrained tutors are effective in terms of student learning gains. Effect sizes ranging from .5 to 2.3 standard deviations have been reported for normal, unskilled tutors (Bloom, 1984; Cohen, Kulik, & Kulik, 1982).

In a typical tutoring session with AutoTutor, students learn the fundamentals of computer hardware, the operating system, and the Internet. AutoTutor's interface is comprised of four features: an animated embodied agent, a text box for typed student input, a text box that displays the problem/question being discussed by the tutor and the student, and a graphics box that displays pictures and animations that are related to the topic at hand. AutoTutor begins the session by introducing himself and then presents the student with an open-ended question or problem that is selected from a curriculum script. The question/problem remains in a text box at the top of the screen until AutoTutor moves on to the next topic. For some questions and problems, there are graphical displays and animations that appear in a specially designated box on the screen. Once AutoTutor has presented the student with a problem or question, a multi-turn tutorial dialog occurs between AutoTutor and the student. All student contributions are typed into the keyboard and appear in a text box at the bottom of the screen. AutoTutor responds to each student contribution with one or a combination of pedagogically appropriate dialog moves (Person, Graesser, Kreuz, Pomeroy & the TRG, 1999). These dialog moves are conveyed via synthesized speech, intonation variations, facial expressions, and rudimentary pointing gestures and do not appear in text form on the screen.

How AutoTutor Works

AutoTutor's architecture is comprised of five major modules: an animated agent, a curriculum script, language analyzers, latent semantic analysis (LSA), and a dialog move generator. All of these modules have been discussed rather extensively in previous publications (see Foltz, 1996; Graesser, Franklin, Wiemer-Hastings, & the TRG, 1998; Graesser, Wiemer-Hastings, Wiemer-Hastings, Harter, Person, & the TRG, in press; Hu, Graesser, & the TRG, 1998; Landauer & Dumais, 1997; McCauley, Gholson, Hu, Graesser, & the TRG, 1998; Olde, Hoeffner, Chipman, Graesser, & the TRG, 1999; Person et al., 1999; Person, Klettke, Link, Kreuz, & the TRG, 1999; Wiemer-Hastings, Graesser, Harter, & the TRG, 1998); and therefore, will not be elaborated in great detail 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 (Graesser & Person, 1994; Graesser et al. 1995; McArthur et al., 1990; Putnam, 1987). AutoTutor's curriculum script includes 36 computer literacy questions/problems that vary in terms of questioning format and difficulty level (see Person et al., 1999). For each of the 36 questions and problems, the curriculum script also includes the following: corresponding "ideal answers" (which are comprised of many potential good answers), anticipated bad answers, corrective splices (i.e., correct answers) for each anticipated bad answer, and numerous dialog moves (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: (1) a word and punctuation segmenter, (2) a parts of speech identifier, and (3) 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, Directive, and Short Response. The Assertions are most relevant to our present implementation of LSA. Namely, LSA is used to assess the *quality* of a learner contribution once it has been classified as an Assertion. AutoTutor uses different strategies for dealing with the other speech act categories: WH-question, YES/NO question, Directive, and Short Response. These strategies, which are needed for a smooth mixed-initiative dialog, will be discussed in the next section.

AutoTutor's knowledge about computer literacy is represented by Latent Semantic Analysis (LSA) (Foltz, 1996; Foltz, Britt, & Perfetti, 1996; Landauer & Dumais, 1997; Landauer, Foltz, & Laham, 1998). LSA is a statistical technique that compresses a large corpus of texts into K dimensions (usually between 100 and 500); we adopted 200 for our current version of AutoTutor. We performed LSA on 2.3 MB of texts that included AutoTutor's curriculum script, two computer literacy textbooks, and 30 articles that discuss hardware, operating systems, and the Internet. The dimensions in LSA serve as orthogonal factors that are used to compute a conceptual relatedness value (a geometric cosine between 0 and 1) between any two "bags" of words. For example, to assess the quality of a student Assertion, LSA matches the student Assertion against two separate conceptual bags, a bag that contains potential good answers for a topic versus a bag that contains the anticipated bad answers. The higher of the two LSA values is considered the best conceptual match, and therefore, determines how AutoTutor construes the student's Assertion. AutoTutor also relies on LSA to monitor two other informative parameters, Topic Coverage and Student Ability Level. For the domain of computer literacy, we have found our application of LSA to be quite

accurate in evaluating the quality of learner Assertions (Graesser, et al., in press; Wiemer-Hastings, Wiemer-Hastings, & Graesser, 1999).

The Assertion quality values computed by LSA play a critical role in determining the type of feedback and dialog move AutoTutor will generate next. AutoTutor's dialog move generator is governed by 15 fuzzy production rules (see Kosko, 1992) 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 bag = HIGH or VERY HIGH] THEN [select POSITIVE FEEDBACK]

(2) IF [student ability = MEDIUM or HIGH & Assertion match with good answer bag = 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?"). AutoTutor currently has a repertoire of 11 dialog moves that are controlled by the dialog move generator. They are Pump, Hint, Splice, Prompt, Elaborate, Summarize 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 and/or problem have been covered, he moves on to the next question or problem in the curriculum script.

AutoTutor's Conversational Features

AutoTutor promotes active student learning by participating in a conversation with the student. Each curriculum script question is designed to engage the student in a multi-turn dialog. Thus, AutoTutor's conversational skills are crucial to his overall effectiveness as a tutor. We are the first to admit that AutoTutor is not ready for the Turing test. However, we do believe that AutoTutor can be an effective tutor and conversational partner provided that he is able to do the following:

(1) Indicate when the student has the floor to contribute. 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). 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. For example, in a previous version of AutoTutor, students were often confused after a number of AutoTutor's dialog moves. That is, the tutorial conversation would abruptly stop, and students would often ask lab assistants what to do. This is illustrated in the following dialog excerpts in which the features of RAM (Excerpts 1 and 2) and the parts and uses of a computer (Except 3 and 4) are being discussed. It is important to remember that AutoTutor's utterances are spoken and do not appear in text form on the monitor. The student responds to AutoTutor by typing a response on the keyboard. In the following excerpts, we've included the students' utterances that were directed to the human lab assistant in brackets.

Excerpt 1
1.1 AutoTutor (Prompt): I bet you know this. When you turn off your computer, whatever is in RAM. . ?

1.2 Student to lab assistant: [See, it's these kinds of questions, I don't know, like, he needs to say, 'here's the next question' or something. He just leaves it at a pause. And I guess I'm supposed to fill in the blank. Is lost maybe?]

1.3 Student types: is lost

1.4 AutorTutor (Prompt Completion): Disappears.

1.5 Student types: i don't know

Excerpt 2
2.1 AutoTutor (Elaboration): Also, The CPU uses RAM as a short-term memory.

2.2 Student to lab assisant: [It's like he's making fun of me or something for not knowing what this is.]

2.3 Student types: i don't know

Excerpt 3
3.1 AutoTutor (Elaboration): Also, a computer has a monitor.

3.2 Student to lab assistant: [I just answer yes or no?]

3.3 Student types: yes

Excerpt 4
4.1 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 . . .?

4.2 Student to lab assistant: [Oh, just answer the question? Um. So, am I still supposed to answer that question (pointing to the topic question that remains on the screen)?]

4.3 Student types: i don't know

4.4 AutoTutor (Prompt Completion): kinds of statistics.

4.5 Student to lab assistant: [Okay, this is really confusing. I don't understand what the question is supposed to be.]

4.6 Student types: i don't know

Each of these dialog excerpts illustrates the students' uncertainty about how to respond (i.e., turns 1.2, 2.2, 3.2, 4.2, and 4.4). We solved many turn-taking problems such as these by creating the Dialog Advancer Network (DAN) (see Figure 1). The DAN allowed us to identify the dialog moves (e.g., Prompts, Hints, and Elaborations) that were responsible for the dead spots in the conversation. Once the dead spots were identified, we were able to adjust the content and intonation of AutoTutor's dialog moves so that students are certain when it is their turn to contribute. The DAN in Figure 1 illustrates the entire dialog move option space for the student and the current version of AutoTutor. Some of the DAN's most salient features are the Advancer States. The Advancer States advance the conversation by clarifying who has the floor in the conversation. For example, prior to the DAN, the conversation often ceased after AutoTutor delivered an Elaboration dialog move because the student did not know what to do. Since the implementation of the DAN, 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).   Since the implementation of the DAN tool, AutoTutor's conversations with students appear to be much smoother and there are considerably fewer conversational deadspots. Two conversations that are managed by the DAN are provided in Excerpts 5 and 6. In Excerpt 5, AutoTutor and the student are discussing the advantages of RISC processors; in Excerpt 6, they are discussing how typed information gets passed from the keyboard to the hard disk.

Excerpt 5
5.1 AutoTutor (Prompt): Now, what kind of architecture allows a RISC processor to execute multiple instructions simultaneously?
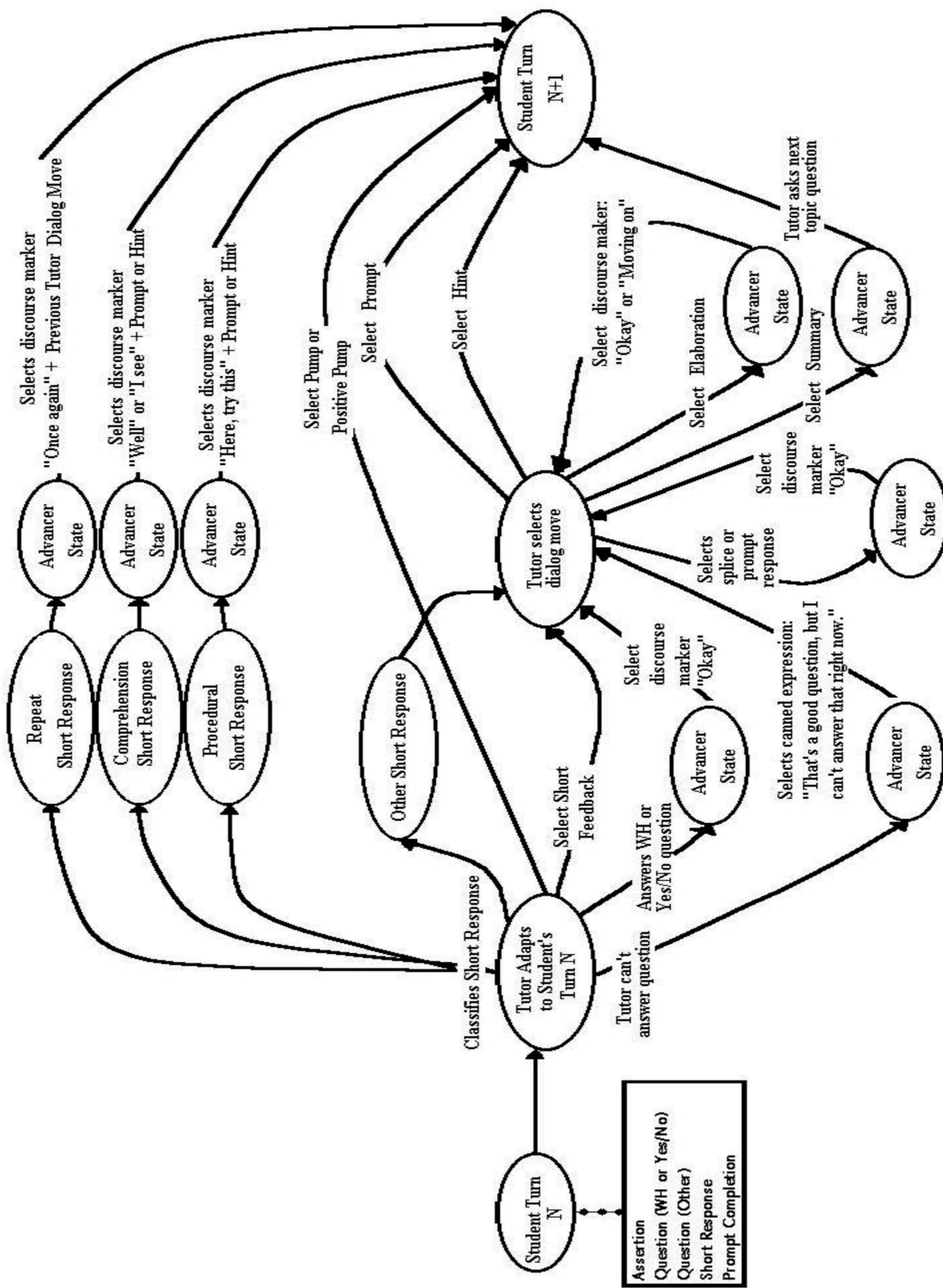
5.2 Student types: simple

Figure 1. Dialogue Advancer Network (DAN)

5.3 AutoTutor (Prompt Completion): A superscalar architecture.

5.4 AutoTutor (Dialog Advancer): Okay,

5.5 AutoTutor (Elaborate): and RISC processors have fewer bottlenecks that complicate processing.

5.6 AutoTutor (Dialog Advancer) Okay,

5.7 AutoTutor (Summary): Now to wrap up, RISC processors are faster and more efficient than CISC processsors (AutoTutor continues to summarize this topic).

Excerpt 6
6.1 AutoTutor (prompt): Well, where is most of the information you type in temporarily stored?

6.2 Student types: RAM

6.3 AutoTutor (Positive Feedback): Right!

6.4 AutoTutor (Prompt Completion): In RAM,

6.5 AutoTutor (Dialog Advancer): Moving on,

6.6 AutoTutor (Elaborate): The CPU handles one instruction at a time.

6.7 AutoTutor (Dialog Advancer): Okay,

6.8 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).

Excerpts 5 and 6 include the Dialog Advancers that are depicted in the DAN. These Dialog Advancers indicate to the student that AutoTutor is keeping the floor in the conversation. In the near future, we hope to equip AutoTutor with hand gestures and other paralinguistic signals (e.g., eye gaze) that will enhance the turn-taking process.

(2) Adapt each dialog move to the previous turn of the student. Coherence emerges in human conversations because participants generally try to adapt the content of their turns so that they are relevant (or linked) to the preceding conversational turn (Clark. & Schaefer, 1987; Grice, 1975; Hobbs, 1979; McLaughlin, 1984; Nofsinger, 1991; Sacks, Schegloff, & Jefferson, 1978). To some extent, AutoTutor is equipped with mechanisms that facilitate this "turn-adaptation" process. As the DAN illustrates, AutoTutor assigns student contributions to different speech act categories (e.g., Assertion, Question, and Short Response) and then generates discourse markers and dialog moves accordingly. More specifically, AutoTutor is able to sustain mixed-initiative dialog by acknowledging student Questions (e.g., "What is RAM?") and by responding to Short Responses (e.g., "I'm lost." or "I don't know.")

AutoTutor's dialog move generator also assists his turn-adaptation capabilities. Each dialog move production rule specifies the conditions (e.g., Assertion quality, topic coverage, and student ability) for which a certain dialog move should be generated. Hence, AutoTutor can adapt each dialog move to the preceding student turn by taking into account the quality of the student's Assertion, the topic material that has been covered, and the student's ability level.

(3) Deliver dialog moves that are consistent with conversational norms. We have had to revise the content of the curriculum script several times during AutoTutor's development. Previous versions of the curriculum script included AutoTutor dialog that violated conversational norms, particularly, the Gricean maxims of quantity, quality, and manner. Initially, he was very redundant and the length of his utterances was too long (violating the quality and quantity maxims). In addition, his speech was very pedantic and the grammatical structure of his sentences was too complicated (violating the manner maxim). We eliminated many of these problems by rewriting the curriculum script so that AutoTutor's dialog would sound more conversational. In short, we shortened the length of AutoTutor's sentences, included more colloquial language, simplified his syntactical structures, and added discourse markers that occur in normal conversations (e.g., "Okay", "Now", "Also").

(4) Appear interested in what the learner is saying. Good conversationalists are able to *appear* interested in what the speaker is saying and provide emotionally appropriate responses. It is the case that AutoTutor can vary his facial expressions and intonation to correspond to various forms of short-feedback (i.e., positive, neutral, or negative). However, AutoTutor's other attentional and emotional capacities are a bit underdeveloped. In order to improve AutoTutor's attention skills, we are currently creating a backchannel feedback mechanism. Backchannel feedback indicates to the speaker that the listener is paying attention, and furthermore, it is a salient feature of human conversations (Clark & Schaefer, 1989; Duncan & Fiske, 1977; Yngve, 1970). AutoTutor will provide backchannel feedback to students with subtle head nods and barely audible "Uh huhs." As for AutoTutor's emotional capacities, our next version of AutoTutor will include a three-dimensional agent that is capable displaying complex emotions and rendering more sophisticated gestures and head movements. Much like the conversational agents developed by Cassell and colleagues, we plan to have AutoTutor deliver dialog moves that are comprised of synchronized speech, intonation, facial expression, head nods, and hand gestures (Cassell, Pelachaud, Badler, Steedman, Achorn, Beckett, Douville, Prevost, & Stone, 1994; Cassell & Thórisson, 1999).

References

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,* 4-16.

Cassell, J., Pelachaud, C., Badler, N.I, Steedman, M., Achorn, B. Beckett, T., Douville, B., Prevost, S., & Stone, M. (1994). Animated conversation: Rule-based generation of facial expression, gesture, & spoken intonation for multiple conversational agents. *Computational Graphics, 28,* 413-420.

Cassell, J., & Thórisson, K.R. (1999). The power of a nod and a glance: Envelope vs. emotional feedback in animated conversational agents. *Applied Artificial Intelligence*, *13*, 519-538.

Clark, H. H. & Schaefer, E. F. (1987). Collaborating on contributions to conversations. *Language and Cognitive Processes, 2,* 19-41.

Clark, H. H. & Schaefer, E. F. (1989). Contributing to discourse. *Cognitive Science, 13,* 259-294.

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

Duncan, S., & Fiske, D. W. (1977). *Face–to-face interaction: Research, methods, and theory.* New York: John Wiley.

Foltz, P.W. (1996). Latent semantic analysis for text-based research. *Behavior Research Methods, Instruments, and Computers, 28,* 197-202.

Foltz, P. W., Britt, M. A., & Perfetti, C. A. (1996). Reasoning from multiple texts: An automatic analysis of readers' situation models. *Proceedings of the 18th Annual Conference of the Cognitive Science Society* (pp. 110-115). Mahwah, NJ: Erlbaum.

Fox, B. (1993). *The human tutorial dialog project.* Hillsdale, NJ: Erlbaum.

Graesser, A. C., & Person, N. K. (1994). Question asking during tutoring. *American Educational Research Journal, 31,* 104-137.

Graesser, A.C., Franklin, S., & Wiemer-Hastings, P. & the Tutoring Research Group (1998). Simulating smooth tutorial dialog with pedagogical value. *Proceedings of the American Association for Artificial Intelligence* (pp. 163-167). Menlo Park, CA: AAAI Press.

Graesser, A. C., Person, N. K., & Magliano, J. P. (1995). Collaborative dialog patterns in naturalistic one-on-one tutoring. *Applied Cognitive Psychology, 9,* 359-387.

Graesser, A.C., Wiemer-Hastings, P., Wiemer-Hastings, K., Harter, D., Person, N., & the Tutoring Research Group (in press). Using latent semantic analysis to evaluate the contributions of students in AutoTutor. *Interactive Learning Environments.*

Grice, H. P. (1975). Logic and conversation. In P. Cole & J. Morgan, (Eds.), *Syntax and semantics, vol. 3: Speech acts* (pp. 41-58). New York: Academic Press.

Hobbs, J. R. (1979). Coherence and coreference. *Cognitive Science*, *3*, 67-90.

Hu, X., Graesser, A. C., & the Tutoring Research Group (1998). 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, (pp. 337-341). Beijing, China: Springer.

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

Kosko, B. (1992). *Neural networks and fuzzy systems.* New York: Prentice Hall.

Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review.*

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes, 25,* 259-284.

McArthur, D., Stasz, C., & Zmuidzinas, M. (1990). Tutoring techniques in algebra. *Cognition and Instruction, 7,* 197-244.

McCauley, L., Gholson, B., Hu, X., Graesser, A. C., & the Tutoring Research Group (1998). Delivering smooth tutorial dialog using a talking head. *Proceedings of the Workshop on Embodied Conversation Characters* (pp. 31-38). Tahoe City, CA: AAAI and ACM.

McLaughlin, M. L. (1984). *Conversation: How talk is organized.* Beverly Hills, CA: Sage.

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

Moore, J.D. (1995). Participating in explanatory dialogues. Cambridge, MA: MIT Press.

Nofsinger, R. E. (1991). *Everyday Conversation*. Newbury Park, CA: Sage.

Olde, B. A., Hoeffner, J., Chipman, P., Graesser, A. C., & the Tutoring Research Group (1999). A connectionist model for part of speech tagging. *Proceedings of the American Association for Artificial Intelligence* (pp. 172-176). Menlo Park, CA: AAAI Press.

Person, N. K., & Graesser, A. C. (1999). Evolution of discourse in cross-age tutoring. In A.M. O'Donnell and A. King (Eds.), *Cognitive perspectives on peer learning* (pp. 69-86). Mahwah, NJ: Erlbaum.

Person, N. K., Graesser, A. C., Kreuz, R. J., Pomeroy, V. & the Tutoring Research Group. (1999*). Simulating human tutor dialog moves in AutoTutor.* Submitted to *International Journal of Artificial Intelligence in Education*.

Person, N. K., Graesser, A. C., Magliano, J. P., & Kreuz, R. J. (1994). Inferring what the student knows in one-to-one tutoring: The role of student questions and answers. *Learning and Individual Differences, 6*, 205-29.

Person, N. K., Klettke, B., Link, K., Kreuz, R. J., & the Tutoring Research Group (1999). The integration of affective responses into AutoTutor. *Proceeding of the International Workshop on Affect in Interactions* (pp. 167-178). Siena, Italy.

Person, N. K., Kreuz, R. J., Zwaan, R., & Graesser, A. C. (1995). Pragmatics and pedagogy: Conversational rules and politeness strategies may inhibit effective tutoring. *Cognition and Instruction, 13,* 161-188.

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

Sacks, H., Schegloff, E. A., & Jefferson, G. (1978). A simplest systematics for the organization of turn taking for conversation. In J Schenkein (Ed.), *Studies in the organization of conversational interaction.* New York: Academic Press.

Wiemer-Hastings, P., Graesser, A. C., Harter, D., & the Tutoring Research Group (1998). The foundations and architecture of AutoTutor. *Proceedings of the 4th International Conference on Intelligent Tutoring Systems* (pp. 334-343). Berlin, Germany: Springer-Verlag.

Wiemer-Hastings, P., Wiemer-Hastings, K., & Graesser, A. C. (1999). Improving an intelligent tutor's comprehension of students with Latent Semantic Analysis. *Artificial Intelligence in Education* (pp. 535-542). Amsterdam: IOS Press.

Yngve, V. H. (1970). On getting a word in edgewise. In M. A. Cambell et al. (Eds.), *Papers from the sixth regional meeting*, *Chicago Linguistics Society.* Chicago: University of Chicago Linguistics Department.