Responding to Student Uncertainty in Spoken Tutorial Dialogue Systems

Heather Pon-Barry
ponbarry@csli.stanford.edu

Karl Schultz
schultzk@csli.stanford.edu

Elizabeth Owen Bratt
ebratt@csli.stanford.edu

Brady Clark
bzack@northwestern.edu

Stanley Peters
peters@csli.stanford.edu

Mailing address:

Stanford University
CSLI
Cordura Hall
210 Panama Street
Stanford, CA 94305-4115

Responding to Student Uncertainty in Spoken Tutorial Dialogue Systems

Heather Pon-Barry, Karl Schultz, Elizabeth Owen Bratt, Brady Clark, and Stanley Peters

Center for the Study of Language and Information, Stanford University
210 Panama Street, Stanford, CA 94305-4115, USA
{ponbarry, schultzk, ebratt, bzack, peters}@csli.stanford.edu

Abstract. In designing and building tutorial dialogue systems it is important not only to understand the tactics employed by human tutors but also to understand *how* tutors decide when to use various tactics. We argue that these decisions are based not only on student problem-solving steps and the content of student utterances, but also on the meta-communicative information conveyed through spoken utterances (e.g., pauses, disfluencies, intonation). Since this information is often infrequent or unavailable in typed input, tutorial dialogue systems with speech interfaces have the potential to be more effective than those without. This paper gives an overview of the Spoken Conversational Tutor (SCoT) that we have built and describes how we are beginning to make use of spoken language information in SCoT. Specifically, we describe a study aimed at using meta-communicative information to gauge student uncertainty and respond accordingly. In this study, we identify linguistic devices used by human tutors when responding to utterances containing signals of uncertainty, integrate these response strategies into two versions of SCoT, and evaluate their relative effectiveness. Our main hypothesis—that tutors are more effective if they use these linguistic devices in response to student uncertainty—was not confirmed, but our secondary hypothesis—that tutors using these linguistic devices are more effective than tutors that do not use them—was supported by the results.

INTRODUCTION

Studies of human-to-human tutorial interaction have identified many dialogue tactics that human tutors use to facilitate student learning (Graesser, Person, & Magliano, 1995; Heffernan, 2001; Evens & Michael, In press). These include tactics such as pumping the student for more information, giving a concrete example, or making reference to the dialogue history. Furthermore, transcripts of human to human tutorial interactions have been analyzed in order to understand patterns between the category of a student utterance (e.g. partial answer, request for clarification) and the category of a tutor response (e.g. positive feedback, leading question) (Person & Graesser, 2003). However, since the majority of dialogue-based Intelligent Tutoring Systems (ITS) rely on typed student input, information from the student utterance is limited to the content of what the student typed. Human tutors have access not only to the words uttered by the student, but also to meta-communicative information such as timing and the way a response is delivered; they use this information to diagnose the student and to choose appropriate tactics (Fox, 1993). This suggests that in order for a dialogue-based ITS to tailor its choice of tactics in the way that humans do, the student utterances must be spoken rather than typed.

Intelligent tutoring systems that have little to no natural language interaction have been deployed in public schools and have been shown to be more effective than classroom instruction alone (Koedinger et al., 1997). However, the effectiveness of both expert and novice human tutors (Bloom, 1984; Cohen, Kulik, & Kulik, 1982) suggests that there is more room for improvement. In recent years, many intelligent tutoring systems have begun to incorporate natural language dialogue. Some tutorial dialogue systems employ short, directed dialogues when the student goes down an incorrect path or completes a step requiring explanation (VanLehn et al., 2002; Zinn, Moore, & Core, 2002; Aleven, Koedinger, & Popescu, 2003). Other tutorial dialogue systems lead longer natural language dialogues throughout the problem-solving process (Person et al., 2001; Michael et al., 2003; Heffernan & Koedinger, 2002). Person et al. (2001) found average learning gains of 0.7 standard deviations greater than the control, and VanLehn et al. (2002) found that adding natural language capabilities to an existing model-tracing tutor increased learning gains by 0.9 standard deviations. These results suggest that dialogue-based tutoring systems may be more effective than tutoring systems with no dialogue. However, all of these systems use either keyboard-to-keyboard interaction or keyboard-to-speech interaction (where the student's input is typed and the tutor's output is spoken). This progression towards human-like use of natural language suggests that tutoring systems with speech-to-speech interaction might be even more effective.

The current state of speech technology has allowed researchers to build successful spoken dialogue systems in domains ranging from travel planning to in-car route navigation (Walker et al., 2002; Belvin, Burns, & Hein, 2001). There is reason to believe that spoken tutorial dialogue systems can be just as successful.

In the remainder of this paper, we describe (a) the potential advantages of spoken interaction, (b) the SCoT framework and how it provides an infrastructure for comparative study of tutorial strategies, (c) an evaluation we conducted to test our hypotheses about how tutors respond to signals of uncertainty, and (d) the challenges we have faced along the way.

POTENTIAL ADVANTAGES OF SPOKEN DIALOGUE

In this section we discuss research that suggests that tutorial dialogue systems with speech interfaces have certain advantages over tutorial dialogue systems that depend solely on typed input. First, spoken language contains meta-communicative information that, alongside student actions and the literal meaning of student utterances, helps human tutors decide upon effective tutoring tactics and infer an accurate student model. Second, spoken interaction allows students to gesture with their hands while speaking. Third, spoken tutorial interactions contain a higher number of student turns and self-explanations. These potential advantages are described in turn below.

Spoken dialogue contains many meta-communicative features that human tutors can use to gauge student understanding and student affect. These features include:

- hedges (e.g. "I guess I just thought that was right")
- disfluencies (e.g. "um", "uh", "What-what is in this space?")
- prosodic features (e.g. intonation, pitch, energy)
- temporal features (e.g. pauses, speech rate)

Human tutors may use the dialogue features listed above to infer a more accurate assessment of student confidence or uncertainty, and consequently adapt the discussion to the student's strengths and weaknesses. Litman and Forbes-Riley (2004) have demonstrated that the prosodic and acoustic information conveyed by speech can improve the detection of confusion and may be useful for adapting tutoring to the student. In building an ITS, many of these features of spoken language can be detected, and used both in selecting the most appropriate tutoring tactic and in developing a more accurate student model. For example, long-term factors such as the student's knowledge of the domain and motivation, as well as short-term factors like the student's understanding of the tutor's utterance may be more accurately determined (Litman et al. 2004).

In analyses of non-tutorial dialogue, we see evidence that this meta-communicative information is an important part of conversation. Studies in psycholinguistics have shown that when answering questions, speakers produce hedges, disfluencies, and rising intonation when they have a lower "feeling-of-knowing"

(Smith & Clark, 1993) and that listeners are sensitive to these phenomena (Brennan & Williams, 1995). In a Wizard-of-Oz style comparison of typed vs. spoken communication to access an electronic mail system, the number of disfluencies was found to be significantly higher in speech than in typing (Hauptmann & Rudnicky, 1988). There are no formal analyses comparing the relative frequencies of hedges, however, a comparison of transcripts of typed dialogues and transcripts of spoken dialogues suggests that some hedges (e.g. "I guess") are significantly more frequent in speech, while other hedges (e.g. "I think") are equally frequent in both speech and typing (data from Bhatt, 2004 and CIRCSIM corpus).

A second benefit of spoken interaction is the ability to coordinate speech with gesture. Compared to keyboard input, spoken input has the advantage of allowing the student to use their hands to gesture (e.g., to point to objects in the workspace) while speaking. Studies have shown that speech and direct manipulation (i.e., mouse-driven input) have reciprocal strengths and weaknesses which can be leveraged in multimodal interfaces (Grasso & Finin, 1997). For certain types of tutoring (i.e., tutoring where the student is doing a lot of pointing and placing), spoken input and direct manipulation together may be better than just speech or just direct manipulation. Furthermore, allowing the student to explain their reasoning while pointing to objects in the GUI creates a *common workspace* between the participants (Clark, 1996) which helps contextualize the dialogue and facilitate a mutual understanding between the student and tutor, making it easier for the tutor to know if the student is understanding the problem correctly.

Finally, recent evidence indicates that in human-to-human tutorial interaction, spoken dialogues are more effective than typed dialogues. A study of self-explanation (the process of explaining solution steps in the student's own words) suggests that spontaneous self-explanation is more frequent in spoken rather than typed tutorial interactions (Hausmann & Chi, 2002). In addition, a comparison of spoken vs. typed human tutorial dialogues showed that the spoken dialogues were more effective (i.e., produced larger learning gains) than the typed dialogues (Litman et al., 2004).

OVERVIEW OF SCoT

The design of SCoT is based on the assumption that the activity of tutoring is a joint activity[1] where the content of the dialogue (language and other communicative signals) follows basic properties of conversation but is also driven by the activity at hand (Clark, 1996). Following this hypothesis, SCoT's architecture separates conversational intelligence (e.g. turn management, construction of a structured dialogue history, use of discourse markers) from the activity that the dialogue accomplishes (in this case, reflective tutoring). SCoT is developed within the Architecture for Conversational Intelligence (Lemon, Gruenstein & Peters, 2002), a general purpose architecture supporting multimodal, mixed-initiative dialogue.

SCoT-DC, the current instantiation of our tutoring system, is applied to the domain of shipboard damage control. Shipboard damage control refers to the task of containing and eliminating the effects of fires, floods, and other critical events that can occur aboard Navy vessels. In a typical interaction, students carry out a reflective discussion with SCoT-DC after completing a problem-solving session with DC-Train (Bulitko & Wilkins, 1999), a fast-paced, real-time, multimedia training environment for damage control. An excerpt from a dialogue between a student and SCoT-DC is shown below in Figure 1.

Tutor: There is one more thing you forgot to do. What is it?

Student: um … <pause> … dewater the compartment

Tutor: No, that's not right.

Tutor: When we discussed the fire in the Dry Provisions Storeroom earlier, what did you say needed to be done in order to contain the fire?

[1] A joint activity is an activity where participants coordinate with one another to achieve both public and private goals (Clark, 1996). Moving a desk, playing a duet, and shaking hands are all examples of joint activities.

Student: set fire boundaries

Tutor: Right. Floods need to be contained just like fires. Now, what should you do to prevent a flood from spreading?

Student: I should set flood boundaries

Tutor: Yes, that's right.

Fig. 1. Excerpt of dialogue with SCoT

Before the start of each dialogue, SCoT receives a log of the student's DC-Train session. Next, SCoT's knowledge reasoner generates an expert solution against which to compare the student's actions. The dialogue in Figure 1 illustrates one possible top-level tutorial strategy: stepping through the student's actions and eliciting missing actions. Other top-level strategies include discussing an analogous hypothetical situation, or exhaustively recreating the details of a problem-solving session.

Over the next few sections we will explain how the various components of SCoT interact to produce the dialogue excerpt in Figure 1. The two most relevant components for this discussion are the *dialogue manager* and the *tutor*. The separation of functionality into these two components allows for a wide variety of dialogue to be carried out. By keeping each component appropriately abstracted from the other, the dialogue manager is able to specialize in controlling and managing the speech interaction while the tutor can concentrate on formulating the appropriate discourse plan. These components are presented briefly in the next two sections. A more detailed system description is available in (Clark et al., 2005).

Dialogue Manager

The dialogue manager handles aspects of conversational intelligence (e.g. turn management, construction of a structured dialogue history, use of discourse markers) in order to separate purely conversational aspects of the interaction from tutorial aspects. It contains multiple dynamically updated components—the two main ones are the *dialogue move tree*, a structured history of dialogue moves, and the *activity tree*, a hierarchical representation of the past, current, and planned activities initiated by either the tutor or the student. Figure 2 below shows the *dialogue move tree* corresponding to the sample dialogue in Figure 1.

```
Root
    … [earlier parts of dialogue] …
    -> Report  : "There is one more thing you forgot to do"
    -> WH-Query : "What is it?"
        -> WH-Answer : "um"
        -> WH-Answer : "dewater the compartment"
    -> Report : "No, that's not right"
    -> WH-Query : "When we discussed the fire in the Dry Provisions Storeroom earlier, what did you
                  say needed to be done in order to contain the fire?"
        -> WH-Answer : "set fire boundaries"
    -> Report : "Right."
    -> Report : "Floods need to be contained just like fires."
    -> WH-Query : "Now, what should you do to prevent a flood from spreading?"
        -> WH-Answer : "I should set flood boundaries"
    -> Report : "Yes, that's right."
```

Fig. 2. Sample Dialogue Move Tree

The dialogue move tree is a hierarchical set of dialogue moves representing the various threads of conversation. It is used by the dialogue manager in supporting multi-threaded conversation and in managing turn-taking. Each dialogue move has a type (Report, WH-Query, YN-Query, WH-Answer, YN-Answer or Root) that imposes restrictions on which other types can attach to it. For example, a WH-Query dialogue move indicates that something is being requested, and tells the dialogue manager that only utterances which (a) address the query and (b) come from the other speaker should be attached to it.

In this way, the dialogue manager protects the tutor from uninterpretable or out-of-context utterances and allows the tutor to plan responses or re-plan high-level goals only when relevant. The tutor thinks on the level of activities, which are represented in the *activity tree*. It is important to mention more explicitly how dialogue moves and activities are related. During the normal course of execution, the dialogue manager monitors the states of activities on the *activity tree* (e.g., planned, current, cancelled, done) and will generate an utterance based on certain state changes. For example, when the tutor's activity to elicit a student action (*elicit_action*) becomes "current", this is when a WH-Query is created, actually asking the question (thus aiming to elicit a response from the student). Figure 3 below is an example of what the *activity tree* would look like at the end of the sample dialogue (from Figure 1). For SCoT, each activity initiated by the tutor corresponds to a tutorial goal; the decompositions of these goals are specified by activity recipes contained in the recipe library (activity recipes will be described further in the next section).

```
-> Root
    -> Discuss_Errors_In_Step
        -> Introduce_Step
             -> State_Num_Unperformed_Necessary_Actions
        -> Elicit_Unperformed_Necessary_Actions
             -> Elicit_Action
                  -> Acknowledge_Incorrect_Answer
                  -> Give_Referring_Back_Hint_question
                       -> Acknowledge_Correct_Answer
                       -> State_Relation_Between_Referring_Back_Problem
                       -> Ask_Followup_Question
                            -> Acknowledge_Correct_Answer
```

Fig. 3. Sample Activity Tree

In addition to maintaining the *dialogue move tree* and the *activity tree*, the dialogue manager also controls all of the other natural language components. In the current version of SCoT we use Nuance[2] as our automatic speech recognizer, Festival and FestVox[3] for limited-domain text-to-speech synthesis, and Gemini (Dowding et al., 1993) as a natural language parser. Figure 4 shows a sample logical form (LF) for one of the student's responses in the sample dialogue:

```
answer(vp(action(set),
      semantic([object(np(n(containment(fire_boundary)),
                  semantic([]),
                  grammatical([number(pl)]))),
            location(null),
            adv_list([])]),
      grammatical([tense_mood_aspect([tense(inf)])])))
```

Fig. 4. Logical Form for *"set fire boundaries"*

[2] http://www.nuance.com

[3] http://festvox.org

This LF, coming from the Gemini parser, gives us three important pieces of information. The first is that the most likely interpretation of this utterance is as an *answer* to something, which the dialogue manager can use for attaching it logically in the *dialogue move tree*. The other two pieces of information are the embedded verb phrase and noun phrase. These two entities comprise the content of the student's answer and are what will be compared to the correct answer(s) held by the tutor.

Tutor

The tutor component contains the tutorial knowledge necessary to plan and carry out a flexible and coherent tutorial dialogue. The tutorial knowledge is divided between a *planning and execution system* and a *recipe library* (see Figure 5).

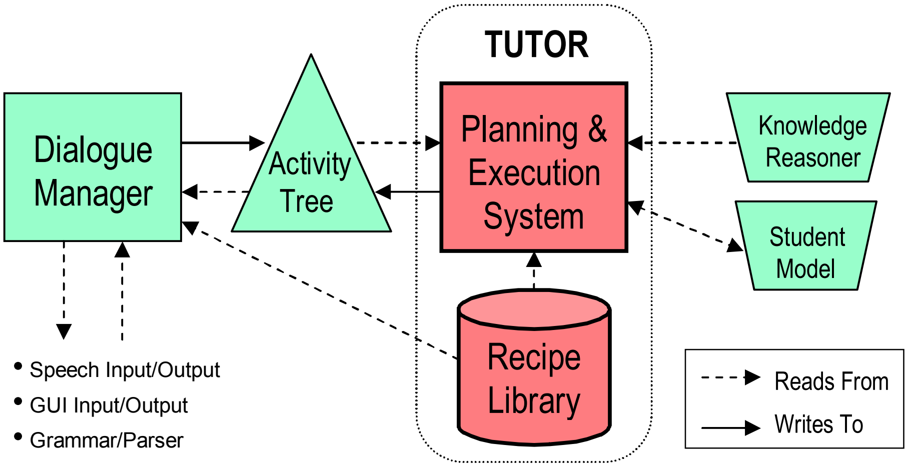


Fig. 5. Subset of SCoT architecture

The **planning and execution system** is responsible for (a) selecting initial dialogue plans, (b) revising plans during the dialogue, (c) classifying student utterances, and (d) deciding how to respond to the student. All of these tasks rely on external knowledge sources such as the knowledge reasoner, the student model, and the dialogue move tree (collectively referred to as the *Information State*). The planning and execution system "executes" tutorial activities by placing them on the *activity tree*, where they are interpreted and executed by the dialogue manager. By separating tutorial knowledge from external knowledge sources, this architecture allows SCoT to lead a flexible dialogue and to continually re-assess information from the Information State in order to select the most appropriate tutorial tactic.

The **recipe library** contains specifications for how to decompose a tutorial activity into other activities and low-level actions. These 'activity recipes' are the constructs by which the tutor reasons about and plans tutorial interaction—a recipe can be thought of as a tutorial goal and a plan for how the tutor will achieve the goal. The library contains activity recipes for both low-level tactics (e.g. responding to an incorrect answer) and high-level strategies (e.g. discussing the student's omitted actions). The recipes are written in a scripted language (Gruenstein, 2002) allowing for automatic translation into system activities. An example activity recipe will be shown below in Figure 6.

The division of knowledge in the tutor component (between the recipe library and the planning and execution system) allows us to independently evaluate hypotheses such as the ones described later in this paper (i.e. test whether their presence or absence changes the effectiveness of SCoT). Each hypothesis is realized by a combination of activity recipes, and the planning and execution system ensures that a coherent dialogue will be produced regardless of which activities are put on the activity tree.

An activity recipe for processing a response to an *elicit_action* activity is shown below. A recipe contains three primary sections: *DefinableSlots*, *MonitorSlots*, and *Body*. The *DefinableSlots* specify what information is passed in to the recipe, the *MonitorSlots* specify which parts of the Information State are used in determining how to execute the recipe, and *Body* specifies how to decompose the activity into other activities or low-level actions. The recipe below decomposes the activity of processing a student's response into anywhere from one to three other activities, depending on the *MonitorSlots* (the classification of the student's response, whether uncertainty in that utterance was detected, and whether the specific topic has already been discussed). The planning and execution system places these activities on the activity tree, and the dialogue manager begins to execute their respective recipes.

```
Activity <process_response_for_elicit_action> {

    DefinableSlots {
        currentProblem;
        currentUtterance;
    }

    MonitorSlots {
        currentUtterance.answerClassification;
        currentUtterance.uncertaintyDetected;
    }

    Body {
        if(answerClassification == CORRECT) {
            Acknowledge_Correct_Answer;
            if (uncertaintyDetected) Paraphrase_Correct_Answer;
        }
        else {
            if (answerClassification == INCORRECT)
                Acknowledge_Incorrect_Answer);
            else if (answerClassification == DONT_KNOW)
                Acknowledge_Neutral;

            if (topicDiscussed && uncertaintyDetected)
                Give_Referring_Back_Hint_Question;
            else {
                Give_Convey_Information_Hint;
                Reask_Question;
            }
        }
    }
```

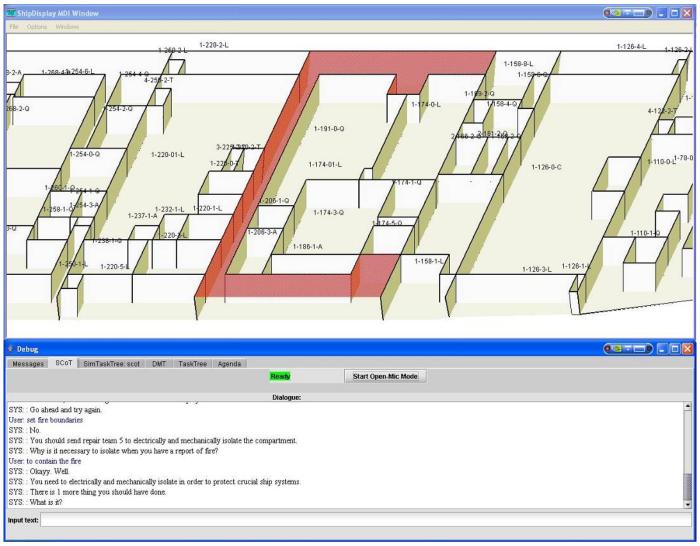
Fig. 6. Activity Recipe for *process_response_for_elicit_action*

This recipe would be responsible for adding the *Acknowledge_Incorrect_Answer* and *Give_Referring_Back_Hint_Question* activities to the *activity tree* shown earlier in Figure 3. All activity recipes have this same structure. The modular nature of the recipes helps us test our hypotheses by making it easy to alter the behavior of the tutor. Furthermore, the tutorial recipes are not particular to the domain of damage control; through our testing of various activity recipes we hope to get a better understanding of domain-independent tutoring principles.

Other components in the Information State that the tutor makes use of are the **knowledge reasoner** and the **student model**. The knowledge reasoner provides a domain-general interface to domain-specific information; it provides the tutor with procedural, causal, and motivational explanations of domain-specific actions. The student model characterizes the causal connections between pieces of target domain knowledge and observable student actions. It can be dynamically updated both during the problem solving session and during the dialogue.

Multimodality

Another way that SCoT takes advantage of the spoken interface is through multimodal interaction. Both the tutor and the student can interactively perform actions in an area of the graphical user interface called the *common workspace*. In the current version of SCoT-DC, the common workspace consists of a 3D representation of the ship which allows either party to zoom in or out and to select (i.e. point to) compartments, regions, and bulkheads (lateral walls of a ship). This is illustrated below in Figure 7, where the common workspace is the large window in the upper portion of the screen.



SYS: : Go ahead and try again.
User: set fire boundaries
SYS: : No.
SYS: : You should send repair team 5 to electrically and mechanically isolate the compartment.
SYS: : Why is it necessary to isolate when you have a report of fire?
User: to contain the fire
SYS: : Okayy. Well.
SYS: : You need to electrically and mechanically isolate in order to protect crucial ship systems.
SYS: : There is 1 more thing you should have done.
SYS: : What is it?

Fig. 7. Screen shot of SCoT-DC

The tutor contextualizes problems being discussed by highlighting compartments in specific colors (e.g. red for fire, gray for smoke) to indicate the type and location of the crises. With the addition of graphical context, the tutor can use speech that is succinct and immediately conveys the nature of the problem. The graphical representations also aid the tutor by allowing it to use simpler spoken referents in situations where various different entities are present in the context, such as "This compartment" rather than "Compartment 2-174-6-Q."

Because the dialogue in SCoT is spoken rather than typed, students are free to use their hands to make gestures in the common workspace while they are speaking. This allows them to "point to" compartments, regions, and bulkheads in the ship display while they are explaining an action they took in the session, or asking a hypothetical question.

Understanding terminology is an important issue in the domain of damage control and students are at times asked to label or select corresponding entities in the common workspace in order to demonstrate their knowledge. Allowing the students to respond with graphic manipulations as well as with spoken answers can lighten the student's cognitive load by letting them use fewer words and/or their modality of choice (Oviatt, 1999). This functionality in SCoT was not present in the version used by the study described in this article. However, a study comparing the effectiveness of four combinations of multimodal input and output in SCoT was conducted at the US Naval Academy in February 2005. The results of this large-scale study (200+ participants) are currently being processed.

ANALYSIS OF HUMAN TUTORING TRANSCRIPTS

In order to develop hypotheses about how tutors respond to signals of uncertainty, we conducted a small-scale analysis of human tutorial transcripts. It has been observed that "tutors use the timing of a student's response, and the way the response is delivered, in addition to what might be called the 'literal content' of the response, as a source of diagnostic information" (Fox, 1993). Fox argues that human tutorial dialogue contains many "transition relevance places," where one party gives the other the chance to take the floor, and that tutors infer a student's level of understanding by observing when and how the student takes the floor (e.g., by completing a sentence started by the tutor, by repeating a phrase spoken by the tutor, or by answering a question right away versus after a delay).

In our investigation, we were interested in both the timing and the delivery of responses—in particular, in cues that can signal student uncertainty (e.g., hedges, response latencies, mid-sentence pauses, filled pauses, trailing off, and fragmented or incomplete sentences). Specifically, we were interested in how tutors vary the way they respond to student answers depending on (a) the student's language, and (b) the manner in which the answer was spoken. To do this, we examined transcripts of one-on-one human tutoring in multiple domains. This section summarizes the analysis performed and the subsequent observations.

Data

Transcripts of human tutorial dialogues from various corpora in the domains of physiology, algebra, and shipboard damage control were analyzed in order to understand how human tutors respond to signals of uncertainty, focusing specifically on the linguistic constructions used in responding.

The dialogues in the domain of physiology came from the CIRCSIM-Tutor corpus of human tutorial dialogues collected by M. Evens, J. Michael, and A. Rovick at Rush Medical College. The corpus includes transcripts of 6 face-to-face tutoring sessions (approximately 2000 dialogue turns) and 75 keyboard-to-keyboard tutoring sessions, of which 5000 lines have been annotated for tutorial goal structure, student answer classification, and other relevant information. These dialogues were collected in order to guide the development of the CIRCSIM-Tutor system (Michael et al., 2003).

The dialogues in the domain of algebra came from the Ms. Lindquist Algebra Tutor corpus of human tutoring dialogues collected by Neil Heffernan (Heffernan, 2001). The corpus includes a transcript of a one-on-one hour-long tutoring session between an experienced mathematics tutor and an eighth grade student. The transcript contains approximately 400 dialogue turns.

The dialogues in the domain of shipboard damage control were collected at the US Navy's Surface Warfare Officer's School (SWOS) in Newport, RI. The 15 debriefs contain approximately 240 dialogue turns in total.

We divided the data into two groups—one for typed tutorial dialogue and one for spoken tutorial dialogue. The typed dialogue group contains the 5000 lines of annotated transcripts from the CIRCSIM corpus. The spoken dialogue group contains the CIRCSIM face-to-face transcripts, the Ms. Lindquist transcript, and the damage control transcripts. In this paper, these two groups are referred to as the 'typed dialogue transcripts' and the 'spoken dialogue transcripts'.

Method

As previously mentioned, spoken dialogue contains many meta-communicative features that human tutors can use to gauge student understanding and student affect. For this analysis, we were interested in the timing and the delivery of responses—in particular, in cues that can signal student uncertainty. The features that we considered to be candidate signals of uncertainty are summarized below in Table 1 (recall discussion from 'Advantages of Spoken Dialogue' section).

Type of Cue	Example
Lexical	hedges (e.g., “I think…”, “Maybe…”)
Temporal	response latencies mid-sentence pauses filled-pauses (e.g., “uh”, “um”)
Other	trailing off at the end of a sentence fragmented or incomplete sentences

Table 1. Signals of Uncertainty

Because the typed dialogue transcripts were annotated for tutorial goals, we used them as a starting point from which to get a preliminary understanding of the tactics tutors use in responding to uncertain student answers. The first signal of uncertainty examined was hedging.[4] Bhatt (2004) outlines a list of hedge categories, which we adopted for this investigation. They are shown below in Table 2.

Hedge Keywords	
“I think”	“it sounds as though”
“I thought”	“X should…”
“probably”	“it shouldn’t X, should it?”
“I guess”	“I assumed that…”
“I’m not sure”	“I can try…”
“kind of”	“what I understand…”
“I believe”	answers phrased as questions
“maybe”	

Table 2. Hedge keywords from Bhatt (2004)

In order to understand the distribution of tutor responses to hedged student answers, we analyzed answer-response pairs from the typed dialogue transcripts along the dimensions of incorrect vs. correct and hedged vs. non-hedged. Results are described below in Table 3.

Observations and Hypotheses

In the typed dialogue transcripts (approximately 270 dialogue turns), tutor responses to hedged and non-hedged answers occurred with the following distribution:

[4] We do not assume that hedging always indicates uncertainty, but rather that hedging *can* indicate uncertainty. Furthermore, we do not intend to suggest that hedged or uncertain answers are more likely to be incorrect. In fact, Bhatt (2004) found that students’ hedges are not a reliable cue to errors or misconceptions.

	Incorrect answers (n = 17)	**Correct answers (n = 39)**
Hedge	Refer back to past dialogue (3) Point out misconception (3) Follow incorrect line of reasoning (2) State answer (2)	Paraphrase student answer (4) Other (1)
No Hedge	Inform of mechanism (2) Try different line of reasoning (3) Give hint (2)	Acknowledge & move on (34)

Table 3. Categories of Tutor Responses to Student Answers

Although it may appear that the various tactics for responding to student answers in Table 3 have no pattern to their distribution, a closer examination reveals that the tactics used in responding to hedged answers all involve elaboration on the current topic while the tactics used in responding to non-hedged answers do not. It makes sense that a tutor might elaborate on the current topic—either to fill in possible gaps in knowledge or to give positive reinforcement for known material.

Two of the response tactics identified in the typed dialogue transcripts, reminding the student of past dialogue and paraphrasing the student's answer, involved linguistic manipulation of the sort we were interested in. In the next step, we examined the spoken dialogue transcripts to understand in what situations human tutors used these tactics in one-on-one tutoring.

The first tutorial tactic examined was *referring back to past dialogue*, i.e., constructions where the tutor reminds the student of something previously discussed. Of the 1600 turns in the spoken dialogue transcripts there were 180 incorrect student answers, 72 of which contained signals of uncertainty. Thirty-one instances of a tutor reminding a student of something previously discussed were identified. Of the responses to *incorrect uncertain* student answers, 29.2% referred back to previous dialogue; of the responses to *incorrect certain* student answers, only 4.6% referred back to previous dialogue. An example from the Ms. Lindquist corpus is reprinted below in Figure 8. The example shows a student answer containing many mid-sentence pauses.

Student: 600-30+20 divided by :::::::::::::: two :::::::: no this parts wrong ::: [writes 600-(30+20)/2 and then scratches out the 600-]

Tutor: Right.

Tutor: That [points at (30+20)/2] looks great but it doesn't work. OK You would think it would, you are just averaging, but it doesn't work. What did we define average speed as earlier?

Fig. 8. Example of Reference to Previous Dialogue (':' = 0.5 sec pause)

The examples found support the generalization that tutors frequently refer back to past dialogue in response to incorrect student answers that contain signals of uncertainty such as hedges, mid-sentence pauses, or trailing off. It is plausible for a tutor to purposefully remind a student of previous dialogue when the student shows signals of uncertainty because it encourages reflection. Chi (2000) argues that self-reflection often leads to self-repair, and that compared to hearing corrective feedback, students learn more when encouraged to reflect. Also, analyses of human tutorial dialogue have shown that tutors generally exploit prior explanations rather than repeating the same information twice (Moore, Lemaire & Rosenblum, 1993) and that reflective discussions can increase learning (Katz et al., 2003).

The next tutorial tactic examined was *paraphrasing*. Of the 1600 turns in the spoken dialogue transcripts there were 337 correct student answers, 103 of which contained signals of uncertainty. Fifteen instances of a tutor paraphrasing a student's answer were identified.[5] Of the responses to *correct uncertain* student answers, 10.1% paraphrased the student's answer; of the responses to *correct certain* student answers, only 1.3% paraphrased the student's answer. An example from the CIRCSIM corpus is reprinted below in Figure 9. It shows a series of student utterances containing three sentences that trail off at the end as well as the hedge "I guess."

The examples found support the generalization that tutors paraphrase correct student answers containing signals of uncertainty (i.e., hedges, mid-sentence pauses, trailing off) more frequently than correct student answers without signals of uncertainty. This generalization seems plausible because paraphrasing reinforces knowledge that the student may be uncertain of and helps them to think about the answer more concisely. Furthermore, paraphrasing can be seen as an attempt to *ground* the conversation, to establish joint actions as part of a common ground (Clark, 1996) and let the student know that s/he has succeeded in communicating the information s/he was attempting to convey.

Tutor: And [initial fiber resting length] relates to which of these parameters?

Student: Let's see, initial fiber resting length would be...

Student: I'd say it's the preload which is...

Student: Well, it relates to the stroke volume, but that's ...

Tutor: Now the question is what determines stroke volume, and you told me contractility, and what else?

Student: Well, I guess if the right atrial pressure were a lot higher, then there would be more of an impetus for the blood to go into the right ventricle, and that would increase that somewhat.

Tutor: So right atrial pressure represents one of the determinants.

Student: Yes.

Tutor: OK.

Fig. 9. Example of paraphrasing a student's answer

In summary, the following two observations were made:

1. Tutors paraphrase *correct* student answers more frequently for answers containing signals of uncertainty than for answers without uncertainty
2. Tutors refer back to previous dialogue after *incorrect* student answers more frequently for answers containing signals of uncertainty than for answers without uncertainty

In the next section, we will explain how these patterns of responding were incorporated into tutoring tactics in SCoT.

[5] Although only 15 instances of tutors paraphrasing a student's answer were identified, many more instances of paraphrasing non-answers were found.

EVALUATION

Our observations and analysis of human tutoring led to the following hypothesis:

> *Tutors that respond to student uncertainty with particular linguistic devices (paraphrasing and referring to past dialogue) will be more effective than those that do not.*

The linguistic devices of paraphrasing correct answers and referring back to earlier dialogue in responding to incorrect answers were combined into one tutorial "treatment" in SCoT and compared to a control treatment where the tutor responded to correct answers with simple acknowledgements and to incorrect answers with generic hints. The signals of uncertainty that SCoT detected and made use of are listed below.

- Lexical hedges ("I think", "I thought", "I guess", "maybe", etc.)
- Filled-pauses ("uh", "um")
- Response latencies (time between tutor's question and student's response)

Other signals of potential interest (e.g., mid-sentence pauses, rising intonation) were not included because system development had not progressed far enough to support them in time for this study. Figures 10 and 11 below show examples (taken from the evaluation) of paraphrasing and referring to previous dialogue.

Tutor: Assuming you have a report of fire, there are three other things you should have done before ordering firefighting. What is one of them?

Student: yeah, um isolate the compartment

Tutor: Yes, that's right.

Tutor: <Paraphrase> You should send repair five to electrically and mechanically isolate the compartment.

Fig. 10. Paraphrasing

Tutor: Ok, which repair team has jurisdiction over this compartment? [highlights compartment]

Student: <pause>…repair three

Tutor: No, that's not right.

Tutor: <ReferBack> When we went over the regions of the ship earlier, which region did we say this was? [highlights region]

Student: fore

Tutor: Yes, and which repair team has jurisdiction over the fore region?

Student: repair two

Tutor: Yes, that's right.

Fig. 11. Referring back to previous dialogue

Methodology

Participants

Forty native English speakers were recruited to participate in this experiment (17 female, 23 male). All subjects were novices in the domain of damage control, thirty-six had no prior experience using speech recognition systems.

Experiment Design

Subjects were randomly assigned to one of four groups (10 per group) and each group received a different style of tutoring, summarized below in Table 1. "T1" refers to the tutorial treatment of paraphrasing correct answers and referring back to past dialogue after incorrect answers (regardless of uncertainty). "T2" (the control treatment) consists of responding to correct answers with simple acknowledgements and to incorrect answers with generic hints.

Group	Treatment for Knowledge Area A	Treatment for Knowledge Area B
I	T1	T2 (control)
II	T2 (control)	T1
III	T1 if uncertain; otherwise control	T2 (control)
IV	T2 (control)	T1 if uncertain; otherwise control

Table 4. Four Experimental Conditions

In order to counter-balance for subject differences, the damage control knowledge that SCoT tutors on was divided into two independent knowledge areas and all subjects received the T1-style tutoring in one knowledge area and the control tutoring in the other. In this way, efficiency was maximized because Group II served as a control for Group I *and* Group I served as a control for Group II (likewise for Groups III and IV). This between-subjects design allows us to determine how the use of T1 devices affects learning gains in each of the four groups.

Knowledge Area A (sequencing) refers to the task of issuing orders at the correct times. Knowledge Area B consists of two sub-areas: boundaries and jurisdiction. Setting boundaries refers to the task of correctly specifying six parameters that describe the perimeter of the area that needs to be cooled or sealed to prevent a crisis from spreading. Jurisdiction refers to the task of giving orders to the appropriate personnel on the ship. Because setting boundaries and assigning jurisdiction both depend primarily on the location of the crisis and not on its other characteristics, they are grouped together.

The experiment was conducted in two rounds. Round 1 consisted of subject groups I and II, and Round 2 consisted of subject groups III and IV. The contingency "T1 if uncertain, otherwise control" present in Round 2 corresponds directly to the hypothesis above. Round 1 was run beforehand in order to determine whether the T1 responses (paraphrasing and referring back), when employed regardless of the student's indications of uncertainty, had any effect on learning. Also, the median latencies for each question-type from Round 1 were used as the thresholds for classifying latencies in Round 2. In this paper, I will refer to the treatments in Round 1 as "non-contingent T1" and the treatments in Round 2 as "contingent T1".

Procedure

Each subject ran through the three DC-Train simulator sessions, in which they were required to solve practical problems of damage control, interspersed with two SCoT dialogues in which they discussed their performance on the preceding DC-Train session. In each SCoT dialogue both knowledge areas were covered, i.e., every dialogue contained both T1-style responses and T2 control responses, but never for the same knowledge area. See Pon-Barry (2004) for transcripts.

Measuring Learning Gains

Learning was measured in two ways. Theoretical knowledge of principles was tested in a 22 question multiple-choice pre-test and a post-test of the same format (11 questions in each knowledge area). Practical mastery of both knowledge areas was assessed through quantitative performance measures (described below in more detail) drawn from each of the three DC-Train scenarios. Problem solving in the damage control domain is different from traditional tutoring domains (e.g., algebra) because the problem state is dynamically changing and there is not one unique solution path per scenario. The DC-Train sessions consist primarily of the user issuing commands and receiving reports. While we do control for time on task (scenarios end after 15 minutes), there is no way to control how many commands a user issues or how many "expert" actions will be suggested. For this reason, a variety of performance measures (including raw scores and percentages) were calculated.

Results

Written Test Results

Learning gains between the pre-tests and post-tests are summarized in Table 5. Raw gains are simply the post-test score minus the pre-test score (as percentages), and normalized gains are: [(post-test – pre-test) / (1.0 – pre-test)]. The mean normalized gains are shown graphically in Figure 12.

Group	Raw Gain Knowledge Area A (Stdev)	Raw Gain Knowledge Area B (Stdev)	Normalized Gain Knowledge Area A (Stdev)	Normalized Gain Knowledge Area B (Stdev)
I	19.1 (10.0)	24.6 (20.1)	49.3 (23.3)	79.6 (35.1)
II	08.2 (10.9)	25.5 (14.7)	22.7 (30.9)	87.2 (18.2)
III	08.2 (12.5)	20.9 (18.7)	15.3 (41.2)	81.9 (34.9)
IV	11.8 (14.9)	25.5 (15.9)	33.0 (36.5)	82.7 (18.9)

Table 5. Learning gains between pre- and post-tests

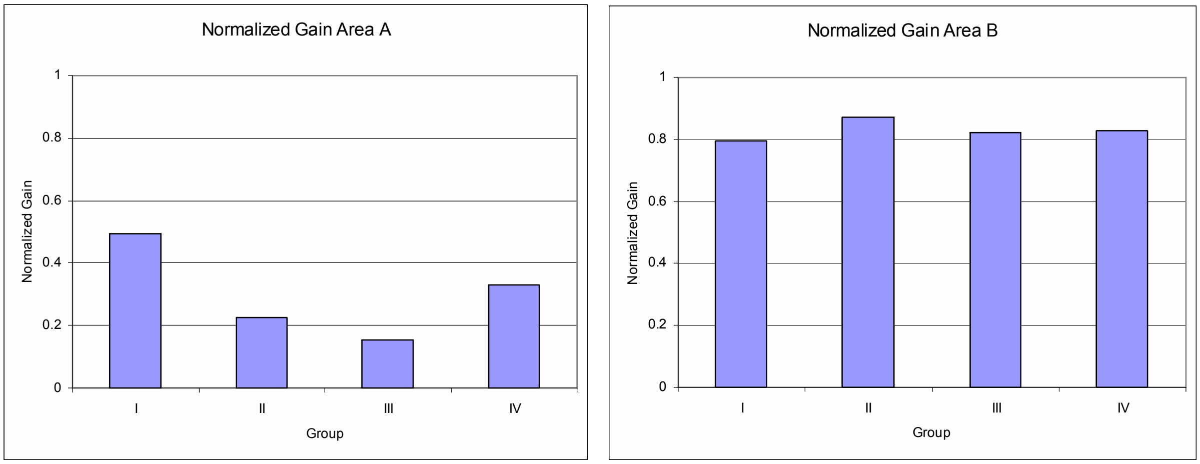


Fig 12. Mean normalized learning gains for Area A (left) and Area B (right)

In order to see whether the differences in means were statistically significant, a one-way ANOVA was run on the normalized test score gains of the four groups. The differences were not significant for either knowledge area, although the probability that the differences were not due to chance was much lower for Area A (p = .143) than for Area B (p = .945). Comparing only groups I and II, the differences in mean gain matched our predictions; group I had greater gains than group II for Area A, and group II had greater gains than group I for Area B. Independent samples T-tests showed that while these differences for Area A were statistically significant (p = 0.042), the differences for Area B were not (p = 0.559). A T-test comparing differences in mean gain between groups III and IV showed no significant differences in either knowledge area.

Performance Results (DC-Train)

Performance measures from the simulator were examined in both knowledge areas. The results for Knowledge Area A showed little difference between all four groups, while the results for Knowledge Area B showed significant differences between groups I and II, and non-significant differences between groups III and IV.

Figure 13(a) below shows the Knowledge Area A ('sequencing') gains for each group. Every action that a student performs (i.e., every command that s/he issues) is graded as either on-time, early, late, or extra. Figure 13(a) represents the gain in percent of student actions that were on-time (i.e., correct) between the first and the third DC-Train scenario. A one-way ANOVA showed none of the differences to be significant. Raw scores were also calculated and did not have any significant differences between groups.

The performance results for Knowledge Area B ('boundaries and jurisdiction') showed more variation. The Area B raw scores represent the number of commands issued to the correct party plus the number of boundary commands issued with the correct bulkheads (regardless of whether the command was issued on time). The overall gains in percent correct are shown in Figure 13(b).

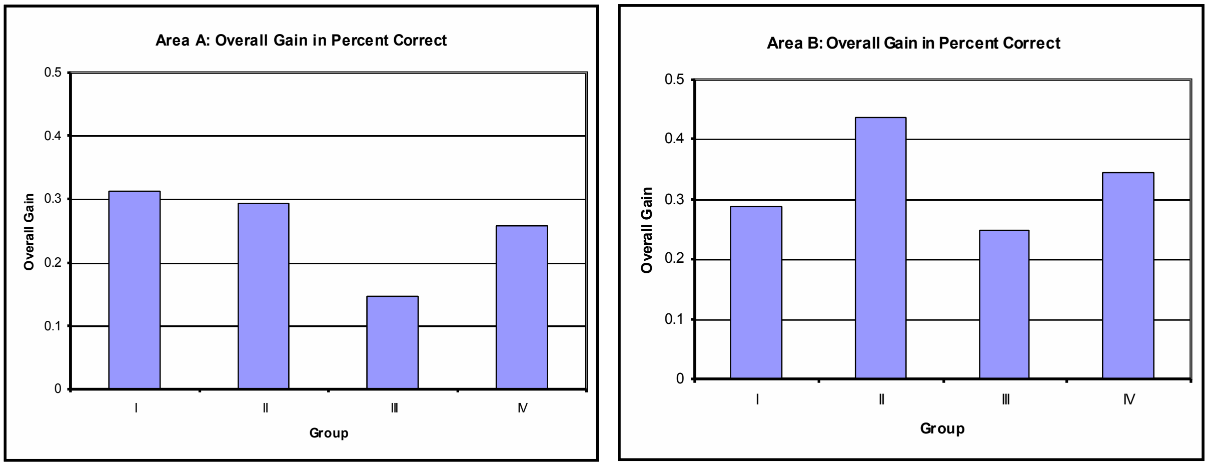


Fig 13. (a) Area A: Total gains in percent correct, (b) Area B: Total gains in percent correct

For Knowledge Area B, Group II received *non-contingent* T1 and Group IV received *contingent* T1, whereas Groups I and III received the control. If T1 really is more effective than the control, we would expect group II to have larger gains than group I, and group IV to have larger gains than group III. If our hypothesis about *when* T1 is most effective is correct, we would expect Group IV to have gains at least as large as Group II. Figure 13(b) shows that while Group IV did not have larger gains than Group II, Group

II did have larger gains than Group I, and Group IV did have larger gains than Group III. Independent samples T-tests showed the difference between Groups I and II to be significant (p = 0.022), but the difference between the Groups III and IV to be non-significant (p = 0.405). Pon-Barry (2004) discusses these results in greater detail.

Detection of Uncertainty

In general, speech recognition accuracy was good (word error rate = 14.5%), although many utterances were rejected (rejection rate = 20.8%). However, there were no significant correlations between recognition accuracy or rejection rate and learning gains—suggesting that poor speech recognition did not diminish learning.

Table 6 below reports the ratio of T1 occurrences to T1 opportunities. There were 80 total T1 opportunities in Round 1 and 79 in Round 2. For Groups I and II, T1 was used at every opportunity, so the ratio is 1:1. For Groups III and IV, the ratios represent how often signals of uncertainty were detected.

Group	**SCoT Dialogue 1:** **T1 used/T1 opportunities**	**SCoT Dialogue 2:** **T1 used/T1 opportunities**
I	100 %	100 %
II	100 %	100 %
III	62.3 %	47.5 %
IV	60.5 %	48.6 %

Table 6. Percent of times T1 "kicked-in" for groups III and IV

Both groups showed less uncertainty in their second tutoring session than in their first, which suggests that they were gaining confidence over time. This was justified, as we have seen they were in fact learning. Because Groups III and IV showed roughly equal amounts of uncertainty at each session, we can infer that the knowledge area of the question asked did not affect how frequently T1 "kicked-in" (for group III 'T1 opportunities' are limited to Knowledge Area A questions; for group IV they are limited to Knowledge Area B questions).

The number of times that each "uncertainty cue" was detected (out of 1670 total student responses from Round 2) is summarized below in Table 7. Unfortunately, hedges and filled-pauses were scarce to non-existent in the data.

Uncertainty Cue	**Number of times cue detected in Round 2**	**Actual number of occurrences in Round 2**
Hedge	0	3
Filled-pause	20	32
Latency > threshold	893	N/A

Table 7. Number of occurrences of uncertainty cues

The data in Table 7 shows that the vast majority of contingent uses of T1 in Round 2 occurred because the student's latency in responding was greater than the threshold. We used response latency as a measure of uncertainty, intending to measure the delay in a subject's response to the system. All responses greater than the given threshold were considered "uncertain." To set the thresholds, we took the median values of the latencies measured in Round 1 of the experiment, with each type of question (sequencing, boundaries, and jurisdiction) considered separately. Although response latency was not a primary focus in the empirical work described earlier, this experiment used latencies in place of mid-sentence pauses because latencies were easier to detect. Automatically detecting pauses and aligning them with the recognized strings is a capability we plan to include in future experiments.

DISCUSSION

The statistically significant difference between T1 (paraphrasing correct answers and referring back to previous discourse in response to incorrect answers) and the control condition (simply acknowledging correct answers and giving generic hints in response to incorrect answers) found in Groups I and II shows that even subtle language variations can affect learning gains. Use of the same linguistic devices contingently in Groups III and IV produced results in the same direction (greater learning with T1 than the control), although this difference was not statistically significant. All the evidence from this experiment is consistent with the claim that paraphrasing and referring back are helpful linguistic devices for tutors to use.

The fact that the test scores showed different patterns than the performance scores is not surprising. The written test evaluates a student's understanding of propositional knowledge without any time pressure, and because the questions were multiple choice a student at chance is expected to get 25% correct. The simulator tests how well students can turn their knowledge into actions in a rapidly changing time-pressured environment. Because the space of possible actions is so large and the grading of the actions depends on a dynamically changing state, a student at chance would get far less than 25% of actions correct. Furthermore, in the area of sequencing, a student must keep track not only of the commands he or she issues, but also of incoming reports (about multiple crises) in order to issue a command on time. So, it seems that while the T1-style tutoring in sequencing gave Group I a better understanding of the propositional knowledge, it may not have been sufficient to affect their performance in the simulator. This finding is relevant for developers of ITSs in general—where learning gains are often measured with written tests alone. Such measurement may be fine for domains where the goal of the tutoring is to improve test scores (e.g., in the classroom), but if the goal is to give students a deeper understanding and the ability to apply their knowledge in practice, then it is important to look at other measures of learning as well.

Regarding the low frequency of hedges and filled-pauses, most subjects spoke verbosely to SCoT at the beginning of their sessions, but switched to giving terse and less natural answers after realizing that many long or complicated phrases could not be understood (see Pon-Barry, 2004). One interesting point, though, is that prior to this experiment, SCoT used a push-to-talk style of interaction. We switched to an open-mic style of interaction (i.e., the system listens continuously) for this evaluation in hopes that it would lead to more hedges, filled-pauses, and other features that are common in human-human conversation. Subjects were more talkative with this version of SCoT (in number of turns taken) than they were with the push-to-talk version, but as just mentioned, this talkativeness diminished as the dialogue progressed. This suggests that with better coverage of natural language phrasings and the ability to detect features such as mid-sentence pauses, a future study like this one might show different or more significant results.

LESSONS LEARNED

Although using spoken language in an intelligent tutoring system has the potential to bring about many of the benefits described in this article, it has also raises many challenges. A few important lessons we have learned are described in this section.

Student Affect

Maintaining student motivation is a challenge for human tutors and intelligent tutoring systems alike. We have observed issues relating to student affect, possibly stemming from the spoken nature of the dialogue.

For example, in a previous version of SCoT, listeners remarked that repeated usage of phrases such as *"You made this mistake more than once"* and "*We discussed this same mistake earlier"* made the tutor seem overly critical. Other (non-spoken) tutorial systems give similar types of feedback (e.g. Evens & Michael, In press), yet none have reported this sort feedback to cause such negative affect. This suggests that users might have different reactions when listening to, rather than reading, the tutor's output, and that further work is necessary to better understand this difference.

A related factor is student fatigue. Experiments such as the one described in this paper demand a high level of concentration from the student, and by the end of 2.5 to 3 hours, many students become tired or mentally worn out (see questionnaire results about "effort required" in Pon-Barry, 2004). It is likely that this mental fatigue adversely affects performance in the final simulator session and/or on the post-test. On average, subjects in this evaluation completed the post-test in half the time it took them to complete the pre-test. Obviously, they were much more familiar with the material during the post-test, but it is also possible that they were not putting as much effort into the questions as they had in the pre-test. In the field of intelligent tutoring systems, where learning gains are often a criterion of success, minimizing student fatigue is an issue that should not be overlooked.

Differences between Human-Human and Human-Computer Conversation

One important lesson learned from this study is that the signals of uncertainty present in human-to-human spoken dialogue may not occur with the same frequency in human-to-computer spoken interaction, even in the best possible dialogue systems. Because most people talk to computers differently than they talk to other humans, we believe a good approach to choosing appropriate signals of uncertainty would be based on an analysis of comparable human-to-computer dialogues. At the same time, the current state of SCoT and of other automated tutoring systems may not reflect their long-term capabilities well enough to determine whether users will ultimately use meta-communicative signals such as hedges to them. As a point of comparison, humans typing to each other with a chat program use fewer hedges than humans speaking face to face, but those who are faster, more prolific typists use hedges at rates approaching those in speech (Brennan & Ohaeri 1999). Furthermore, our experience using open-mic interaction in SCoT (compared to a push-to-talk interface in the previous evaluation) suggests that the interface itself can encourage (or discourage) natural, conversational speech.

Personification of SCoT

Although the meta-communicative features that we observed in human-to-human tutorial interaction (e.g., hedges, filled-pauses) occurred only rarely in this experiment, we should be careful not to assume too quickly that people will never use them when talking to computers. During this evaluation, we observed many users personifying SCoT (cf. Reeves & Nass 1996) both during the tutoring sessions and in the post-experiment questionnaires. Students have apologized to SCoT ("oh sorry"), thanked SCoT ("thank you [laugh]"), and written the following in their questionnaires (responding to *What did you like the most/least about interacting with this system?*):

- "I liked that he sounded as if he were responding directly to me, and how he had a good sense of my performance on the preceding simulations."
- "sometimes, he was patronizing. i didnt like that. i think he's just kinda angry."

Given these examples of social interaction with an automated tutor, it seems reasonable to begin the process of modeling uncertainty in human-to-computer speech based on *all* relevant characteristics of uncertainty in human-to-human speech.

CONCLUSION

In this paper, we argued that spoken language interaction is an integral part of human tutorial dialogue and that information from spoken utterances is very useful in building dialogue-based intelligent tutors that can understand and respond to students as thoroughly and as effectively as human tutors. We described the Spoken Conversational Tutor we have built, and presented the results of an evaluation which used SCoT to test our hypotheses on how human tutors vary their responses depending on the signals of uncertainty in student utterances. The results showed statistically significant differences in learning gain between the *non-contingent* tutoring and the control, and non-significant differences in learning gain between the *contingent* tutoring and the control. Our primary hypothesis that tutors are more effective if they paraphrase and refer back in response to signals of uncertainty was not confirmed, but the results did affirm our secondary hypothesis that paraphrasing and referring back are helpful linguistic devices and that tutors using them are more effective than those who do not. Furthermore, the fact that paraphrasing and referring back are generic linguistic devices and not specific to tutorial dialogue suggests that the effectiveness of human tutoring may be due to general characteristics of conversation in addition to the specific tutoring techniques.

We are still far from understanding exactly how human tutors make use of spoken language features such as disfluencies and pauses, but we are building a tutorial framework that allows us to test various hypotheses, and in time reach a better understanding of how to take advantage of spoken language in intelligent tutoring systems.

ACKNOWLEDGEMENTS

This work was supported by the Office of Naval Research under research grant N000140010660, a multidisciplinary university research initiative on natural language interaction with intelligent tutoring systems. Further information is available at http://www-csli.stanford.edu/semlab/muri. This paper is an extended version of Pon-Barry, Clark, Schultz, Bratt, & Peters (2004a).

REFERENCES

Aleven V., Koedinger, K. R., & Popescu, O. (2003). A Tutorial Dialog System to Support Self-Explanation: Evaluation and Open Questions. In U. Hoppe, F. Verdejo, & J. Kay (Eds.), *Proceedings of the 11th International Conference on Artificial Intelligence in Education, AI-ED 2003,* (pp. 39-46).

Belvin, R., Burns, R., & Hein, C. (2001). Development of the HRL Route Navigation Dialogue System. In *Proceedings of the First International Conference on Human Language Technology Research,* Paper H01-1016.

Bhatt, K. (2004). Classifying student hedges and affect in human tutoring sessions for the CIRCSIM-Tutor intelligent tutoring system. Unpublished M.S. Thesis, Illinois Institute of Technology.

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

Brennan, S. E., & Ohaeri, J. (1999). “Why do electronic conversations seem less polite? The costs and benefits of hedging.” *International Joint Conference on Work Activities, Coordination, and Collaboration* (pp. 227-235).

Brennan, S. E., & Williams, M. (1995). The feeling of another's knowing: Prosody and filled pauses as cues to listeners about the metacognitive states of speakers. *Journal of Memory and Language*, 34, 383-398.

Bulitko, V., & Wilkins., D. C. (1999). Automated instructor assistant for ship damage control. In *Proceedings of the Eleventh Conference on Innovative Applications of Artificial Intelligence, IAAI-99,* (pp. 778-785).

Chi, M. T. H. (2000). Self-explaining expository texts: The dual processes of generating inferences and repairing mental models. In R. Glaser (Ed.), *Advances in Instructional Psychology.* Hillsdale, NJ: Lawrence Erlbaum Associates. 161-238.

Clark, B., Lemon, O., Gruenstein, A., Bratt, E., Fry, J., Peters, S., Pon-Barry, H., Schultz, K., Thomsen-Gray, Z., & Treeratpituk, P. (2005). A general purpose architecture for intelligent tutoring systems. In *Natural, Intelligent and Effective Interaction in Multimodal Dialogue Systems*. Edited by N. Ole Bernsen, L. Dybkjaer, & J. van Kuppevelt. Dordrecht: Kluwer.

Clark, H. H. (1996). *Using Language.* Cambridge: University Press.

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.

Dowding, J., Gawron, M., Appelt, D., Cherny, L., Moore, R., & Moran, D. (1993). Gemini: A natural language system for spoken language understanding. In *Proceedings of ACL 31,* Columbus, OH, 54-61.

Evens, M., & Michael, J. (In press). *One-on-One Tutoring by Humans and Machines.* Mahwah, NJ: Lawrence Earlbaum Associates.

Fox, B. (1993). *Human Tutorial Dialogue*. New Jersey: Lawrence Erlbaum.

Graesser, A. C., Person, N. K., & Magliano J. P. (1995). Collaborative dialogue patterns in naturalistic one-to-one tutoring sessions. *Applied Cognitive Psychology*, 9, 1-28.

Grasso, M. A., & Finin, T. W. (1997). Task integration in multimodal speech recognition environments. *Crossroads*, 3(3), 19-22.

Gruenstein, A. (2002). Conversational Interfaces: A Domain-Independent Architecture for Task-Oriented Dialogues. Unpublished M.S. Thesis, Stanford University.

Hausmann, R. & Chi, M. T. H. (2002). Can a computer interface support self-explaining? *Cognitive Technology*, 7(1), 4-15.

Hauptmann, A. G. & Rudnicky, A. I. (1988). Talking to computers: An empirical investigation. *International Journal of Man-Machine Studies*, 28(6), 583-604

Heffernan, N. T. (2001). Intelligent tutoring systems have forgotten the tutor: Adding a cognitive model of human tutors. Dissertation. Computer Science Department, School of Computer Science, Carnegie Mellon University. Technical Report CMU-CS-01-127.

Heffernan, N. T., & Koedinger, K. R. (2002). An Intelligent Tutoring System Incorporating a Model of an Experienced Human Tutor. *Proceedings of the 6th International Conference on Intelligent Tutoring Systems*, (pp. 596-608).

Katz, S., Allbritton, D., & Connelly, J. (2003). Going beyond the problem given: How human tutors use post-solution discussions to support transfer. *International Journal of Artificial Intelligence and Education*, 13, 79-116.

Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. (1997). Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education*, 8, 30-43.

Lemon, O., Gruenstein, A., & Peters, S. (2002). Collaborative activities and multitasking in dialogue systems. In C. Gardent (Ed.), *Traitement Automatique des Langues (TAL, special issue on dialogue)*, 43(2), 131-154.

Litman, D., & Forbes-Riley, K., (2004). Predicting student emotions in computer-human tutoring dialogues. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics* (pp. 352-359).

Litman, D., Rose, C. P., Forbes-Riley, K., VanLehn, K., Bhembe, D., & Silliman, S. (2004). Spoken Versus Typed Human and Computer Dialogue Tutoring. In J. Lester, R. Vicari, & F. Paraguacu (Eds.) *Proceedings of the 7th International Conference on Intelligent Tutoring Systems* (pp. 368-379).

Michael, J., Rovick, A., Zhou, Y., Glass, M., & Evens, M. (2003). Learning from a computer tutor with natural language capabilities. *Interactive Learning Environments*, 11(3), 233-262.

Moore J. D., Lemaire B., Rosenblum J. A. (1996). Discourse generation for instructional applications: Identifying and exploiting relevant prior explanations. *The Journal of the Learning Sciences*, 5(1), 49-94.

Oviatt, S. (1999). Ten Myths of Multimodal Interaction. *Communications of the ACM*, 42(11), 74-81.

Person, N.K., Graesser, A.C., Bautista, L., Mathews, E., & the Tutoring Reasearch Group. (2001). Evaluating student learning gains in two versions of AutoTutor. In J. D. Moore, C. L. Redfield, & W. L. Johnson (Eds.) *Proceedings of Artificial intelligence in education: AI-ED in the wired and wireless future* (pp. 286-293). Amsterdam: IOS Press.

Person, N.K., & Graesser, A.C. (2003). Fourteen facts about human tutoring: Food for thought for ITS developers. *AIED 2003 Workshop Proceedings, Workshop on Tutorial Dialogue Systems: With a View Towards the Classroom* (pp. 335-344).

Pon-Barry, H. (2004). In search of Bloom's missing sigma: Adding the conversational intelligence of human tutors to an intelligent tutoring system. Unpublished M.S. Thesis, Symbolic Systems Program, Stanford University.

Pon-Barry, H., Clark, B., Schultz, K., Bratt, E., & Peters, S. (2004a). Advantages of spoken language interaction in ttorial dialogue systems. In James Lester, Rosa Maria Vicari, & Fabio Paraguacu (eds.) *Proceedings of the 7th International Conference on Intelligent Tutoring Systems*, Maceió, Brazil, (pp. 390-400).

Pon-Barry, H., Clark, B., Bratt, E., Schultz, K., & Peters, S. (2004b). Evaluating the effectiveness of SCoT: a spoken conversational tutor. In J. Mostow & P. Tedesco (Eds.) *ITS 2004 Workshop on Dialog-based Intelligent Tutoring Systems* (pp. 23-32).

Reeves, B. & Nass, C. (1996). *The media equation: How people treat computers, television, and new media like real people and places*. Stanford, CA: CSLI.

Smith, V. L., & Clark, H. H. (1993). On the course of answering questions. *Journal of Memory and Language,* 32, 25-38.

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

Walker, M., Rudnicky, A., Prasad, R., Aberdeen, V., Bratt, E., Garofolo, J., Hastie, H., Le, A., Pellom, B., Potamianos, A., Passonneau, R., Roukos, S., Sanders, G., Seneff, S., & Stallard, D. (2002). DARPA Communicator: Cross-system results for the 2001 evaluation. In *Proceedings of the 7th International Conference on Spoken Language Processing* (pp. 269-272).

Zinn, C., Moore, J., & Core, M. (2002). A 3-tier planning architecture for managing tutorial dialogue. In *Proceedings of the 6th International Conference, ITS 2002,* (pp. 574-584).