DISCOURSE PROCESSES, *42*(1), 37–74

Annotation of Tutorial Dialogue Goals for Natural Language Generation

Jung Hee Kim
Department of Computer Science
North Carolina A & T State University, Greensboro

Reva Freedman
Department of Computer Science
Northern Illinois University, DeKalb

Michael Glass
Department of Mathematics and Computer Science
Valparaiso University, Valparaiso

Martha W. Evens
Department of Computer Science
Illinois Institute of Technology, Chicago

We annotated transcripts of human tutoring dialogue for the purpose of constructing a dialogue-based intelligent tutoring system, CIRCSIM-Tutor. The tutors were professors of physiology who were also expert tutors. The students were 1st year medical students who communicated with the tutors using typed communication from separate rooms. The tutors made use of a rich variety of strategies, some specific to particular content areas and others more general, such as showing that the student holds contradictory beliefs about the domain. In this article, we describe our model of hierarchical goal structure for tutorial dialogues. We catalog each major pedagogical method we found in the dialogues, showing its structure and illustrating the features

Correspondence should be addressed to Jung Hee Kim, Computer Science Department, North Carolina A & T State University Greensboro, NC 27411. E-mail: jungkim@ncat.edu

needed to represent each subgoal in its correct narrative and interpersonal context. We compare our goal structure with other analyses of tutorial dialogues.

This article describes our experience marking up tutorial dialogue for the purposes of constructing a dialogue-based intelligent tutoring system, CIRCSIM-Tutor. Skilled human tutors exhibit a rich variety of strategies, tactics, and language. Machine tutors typically do not. We took as our goal discovering phenomena in transcripts of highly skilled human tutoring and analyzing them in enough detail to be able to partially imitate them with a computer. There is a wealth of information that can be recovered from a transcript of human tutoring. The phenomena that we selected to annotate were driven by our vision of the machine tutor: how it should operate at a computational level and what kinds of behavior we wanted from it.

The domain of CIRCSIM-Tutor is the baroreceptor reflex, a negative feedback mechanism that maintains a steady blood pressure in the human body. The intended audience is composed of first year medical students studying physiology. This research is a consequence of experience with a long line of instructional software for this material dating back to the early 1970s (Dickinson, Goldsmith, & Sackett, 1973; Rovick & Brenner, 1983; Rovick & Michael, 1986, 1992). In that time the model of instruction evolved from pure quantitative simulation of the physiological variables, with no machine-directed instruction, to language-based interactive tutorial dialogue regarding qualitative causal reasoning about the physiology.

With the goal of producing a more intelligent dialogue-based intelligent tutoring system, we have observed many hours of human tutoring of baroreceptor reflex problems. The transcripts of these sessions have informed every aspect of the CIRCSIM-Tutor project, serving as the basis of studies on tutoring topics such as hinting behavior (Hume, Michael, Rovick, & Evens, 1996; Zhou et al., 1999a), acknowledgment of student answers (Brandle, 1998; Spitkovsky & Evens, 1993), student initiatives (Shah, Evens, Michael, & Rovick, 2002), surface generation of language (Kim, Freedman, & Evens, 1998a), adjustment of tutorial goals (Kim et al., 1998b), the use of analogies as a tutorial tactic (Lulis, Evens, & Michael, 2004a, 2004b), the interpretation of student input (Glass, 1999, 2001), and the comparison of novice to expert tutors (Glass, Kim, Evens, Michael, & Rovick, 1999).

In this article we discuss the annotation of tutorial goal structure in the transcripts, along with related annotations needed for the construction of a computer tutor. We describe the goal structures that we identified and the hierarchy in which they are organized. This study is largely descriptive, not quantitative. Because we are mining the transcripts for their methods, if there is only one instance of a tutoring tactic for a particular problem, we would like to have it. Finally, we compare our work to other analyses of tutorial goal structure.

The next section describes the basis of our annotation system. The following one presents the specific goals, goal attributes, and implementation plans that we found in human tutoring sessions. The last two sections of the article relate our work to analyses from other sources and present our conclusions and plans for future work.

OVERVIEW OF THE MODEL

The Model Underlying Our Annotation

In this section, we describe our abstract model of dialogue; in the next section we show how this model applies to the CIRCSIM-Tutor dialogues.

Scope of analysis. The analysis described in this article is restricted to dialogue where the tutor has the initiative. The tutor having the initiative does not preclude student engagement, for example via questions and hints. We consider here only the tutor's part of the exchanges, although Zhou and colleagues (Zhou, 2000; Zhou et al., 1999b) analyzed student answers in the same dialogues. Elsewhere (Shah et al., 2002), we analyze episodes where the student has the initiative and the tutor is mainly engaged in responding to the student.

In these dialogues the tutor presents a task to a student, then leads him or her in a conversation that helps correct any misconceptions. Grosz (1977) showed that in a task-directed dialogue, the structure of the dialogue tends to match the structure of the task. As a result, the dialogues are in fact largely tutor-driven.

Tree structure. We view the tutor side of each tutor–student dialogue as a tree-structured goal hierarchy. Modeling utterances by the intention of the speaker originates with speech act theory (Austin, 1962; Searle, 1969). These ideas were incorporated by Grosz and Sidner (1986) into a theory covering intentions in continuous discourse. According to Grosz and Sidner, utterances in a discourse are naturally aggregated into discourse segments, just as words are grouped into constituent phrases. The discourse has an overall purpose, and each segment also has a purpose. Grosz and Sidner identified two relations between discourse segments, the parent–child relation of hierarchical dominance and the sibling relation of linear precedence ("satisfaction precedence"), which create a partial ordering of discourse segment purposes.

We extend the work of Grosz and Sidner (1986) to a fully ordered tree by specifying that each sibling must precede the next. For many of the discourse segments we have analyzed, a strict ordering of content is required due to the causal nature of the domain. For others, this extension simply assumes that the order used by our human tutors is a good one to model, whether or not it is the only acceptable one.

Use for text generation. Our primary goal in this analysis is to build a database of tutorial plans for natural language generation. These plans will be input to the newest version of the CIRCSIM-Tutor intelligent tutoring system. CIRCSIM-Tutor is a dialogue-based interactive system where the system plays the role of the tutor. For this reason, we are mainly interested in the tutor's side of the conversation. In addition, we restrict ourselves to dialogue where the tutor has the initiative. Freedman (1997) showed that this restriction does not restrict the mechanized tutor's ability to teach. Because a secondary goal is to learn about the tutoring strategies of human tutors, we have also annotated strategies that a computerized tutor may not be able to handle in the same way that human tutors do.

There are several motivations for using a tree of tutorial goals as the underlying representation for dialogue generation. The research on discourse intentions described earlier predicts a hierarchical structure for our dialogues, and our study bears this out. Reiter and Dale (1997, 2000) showed that use of a hierarchical structure is the best way to generate sizeable single-speaker texts that have coherence and thematic structure. Experience with the earliest of intelligent tutoring systems (Carbonell, 1970) showed that it is not enough for the tutor to produce true sentences with little overall plan. Furthermore, although linguists such as Halliday and Hasan (1976) and Schiffrin (1988) described in detail many of the properties of coherent text, these descriptions are not sufficient for generating such text. They provide methods for marking coherence in the text but not for creating the overall plan that is the basis for coherent discourse.

The tree structure model doesn't completely describe the dialogue, but it does provide a good description of the tutorial material that most interests us. Many of the unplanned utterances come from responses to the student. Based on multiple studies of one-on-one tutoring using tutors in schools, Person and Graesser (2003) showed that most tutor dialogue acts employ a small variety of moves to address the student's immediately preceding turn. Perhaps because tutors in our dialogues had considerable prior tutoring experience and a research background in the domain and its pedagogy, they apparently employ and stick to more elaborate tutorial planning than ordinary tutors do.

Hierarchical model. A key level in our analysis is the transaction, defined in Conversation Analysis (Sinclair & Coulthard, 1975) as the lowest level that corresponds to an elementary task. In our transcripts, a transaction is the correction of one incorrect prediction by the student. Levels above the transaction model the structure of the problem-solving activity the student is engaged in, which is described in the next section.

Our model contains three levels of goals rooted in the remediation of one incorrect prediction (i.e., one transaction): the method level, the topic level, and the primitive level. A goal at the method level defines the strategy that the tutor employs to teach the student the correct value of one variable. Multiple methods can appear in one transaction if the first one does not succeed. The method level can be used to implement various types of deductive reasoning, interactive questioning, and exploration of anomalies.

The topic level is used to represent each piece of domain content that must be taught in service of the selected strategy. Although a method could consist of just one topic, for example a method that gives the student the answer after all other methods have failed, most methods are made up of a series of topics. We often use the term *schema* to refer to methods, especially multistep ones.

The primitive level shows how the information in a topic is communicated to the student. The two main primitive goals are *T-elicits* and *T-informs*. The primitive *T-elicits* is used when the tutor wants the student to participate actively by answering a question or responding to a command. The *T-informs* primitive is used to indicate that the tutor gives some information to the student. The primitives are leaf nodes of the tree structure; they are not further decomposed.

Although most topics consist of just one primitive, multistep topics are also possible. In addition, sometimes the tutor realizes that, due to the nature of the student's misconception, further explanation is needed to teach a topic. In that case, the tutor can use a nested method to teach a topic instead of a primitive or series of primitives. Thus our model is recursive. Although an arbitrary number of levels is permitted, in practice tutors tend not to nest deeply.

Any goal can have attributes that provide additional information. The attributes provide arguments for the predicate expressed by the goal or modifications to the quality of the predicate. For example, *T-elicits* and *T-informs* always occur with arguments specifying the content, and often with arguments providing further detail about how that content should be expressed. Goals inherit attributes annotated on the higher level goals they serve. A primitive goal thus inherits all the attributes on the goals above it.

Partial schemata as seen in the transcripts. The patterns described earlier are frequently interrupted, especially when the student gives an unexpected answer to a question. The student's reply can include many categories, including the following: the expected answer, another answer that is basically correct but does not use the correct terminology, a "near miss" (a step toward the correct answer), a partially correct answer, a wrong answer, a "don't know" response, a student initiative, or a combination of the aforementioned. For an error in language or a misconception that can be corrected with one utterance, the tutor may continue the tutoring method. Before continuing, the tutor may acknowledge the student's statement or address the content provided by the student. For a more serious misconception, the tutor may use a nested method to teach some background content. After the nested method is complete, the tutor returns to the remediation of the variable in question. As a third option, the tutor can drop the remaining steps of an incomplete method and replace it by another, perhaps easier, one. A cue phrase like "let's try again" is sometimes used to delimit a second attempt to satisfy a goal.

This means that instead of seeing complete schemata in the transcripts, we often see the beginning of a schema, then the beginning of another schema after the tutor changes plans. It is these changes in plan that yield a richness of tutoring tactics and methods. For purposes of building a tutor, when we see goals abandoned in the face of unexpected student responses, we want to capture and document the alternative teaching methods that the tutors bring to bear so that we can implement them.

The tutor chooses a method to implement a given goal based on a number of factors. These factors might include domain knowledge (e.g., whether a particular method is applicable in a given set of circumstances), dialogue history (e.g., the student's previous utterance or previous methods the tutor has tried), and information about the student (e.g., how well the student is doing). Although we want to model the goal-updating strategies themselves so that we can implement rules for choosing among different realizations for the same goal, this question is outside the scope of our study.

The tutor must maintain the back-and-forth structure of a two-party conversation as well as carrying out the hierarchical goals in the tutorial plan. Thus we annotate the student's responses and the tutor's acknowledgment of those responses. We code the student's reply and the tutor's acknowledgment as *S-ans* and *T-ack,* respectively. From the point of view of analysis, these items could be considered as the final steps of each tutoring method. However, from the point of view of generation, we prefer not to actually code them as part of the tutoring method.

Inference on the part of the coder. Our approach to annotation requires several types of inference on the part of the coder. As in any form of goal-based annotation, the goals and the schemata realizing them are seldom explicitly stated in the dialogue transcripts. They must be imputed by the coder. For example, when the tutor argues by exhibiting a contradiction between the student's prediction and other relevant facts, the tutor does not explicitly state "I am presenting a contradiction I want you to explain." The meaning is inferred by the hearer at the time, and later by the person annotating the dialogue, who chooses the tag *T-presents-contradiction.* To build plans for text generation, the coder must also identify the tutor's rationale for choosing one method over another and for choosing from the variety of possible responses to a student error. In addition, because a schema is often not expressed in complete form in the text, the coder must sometimes reconstruct a schema from partial examples. It is known (Reiter & Sripada, 2002; Reiter, Sripada, & Robertson, 2003) that human transcripts, although an indispensable source of information, do not provide sufficient data for building a computerized dialogue generation system. In this regard we are fortunate that in engineering earlier versions of the CIRCSIM-Tutor software (Michael, Rovick, Glass, Zhou, & Evens, 2003), our expert tutors were available on a consulting basis to corroborate the validity of the intentions that we ascribed to them and help decide which tutoring methods to employ under which conditions. They also extensively vetted the final working software, correcting mistakes in its dialogue plans.

Intended use in reactive planner. The version of CIRCSIM-Tutor currently under development uses the APE reactive planning package (Freedman, 2000b; Freedman, Haggin, Nacheva, Leahy, & Stilson, 2004; Freedman, Rosé, Ringenberg, & VanLehn, 2000) for dialogue management. Each method and topic can be implemented as a plan. Each plan includes a goal and its arguments, the preconditions under which the plan applies, and a single- or multistep recipe that instantiates the goal. Goals have arguments that can be used to mirror the attributes of the goals in the markup. Arguments are inherited from every higher level enclosing goal. Thus when decomposing a goal, the system has access to the complete context in which the goal appears. Preconditions are used so that different methods are available depending on context. The list of topics in a multistep method can be used as a basis for the recipe.

APE decomposes each plan recursively. When a primitive is reached, APE communicates with the student and waits for the student's reply. A problem with pure hierarchical decomposition is that it requires completing every plan that is begun, even when the topic is no longer relevant to the conversation. This produces very artificial text, because human beings change their plans when the situation changes (Bratman, 1987). Therefore, rather than planning the complete text as in a monologue, APE uses a reactive planning approach. APE interleaves planning and execution, planning only as much as necessary to generate the next turn. This allows the system to change the plan after every student response.

Goal Structure of the Tutorial Dialogues

The tutorial dialogues are organized around helping the student learn to solve problems involving the baroreceptor reflex, the negative feedback system that regulates blood pressure in the human body. The student is presented with a description of a procedure or perturbation, an event that changes blood pressure. The student is then asked to predict the qualitative changes (whether they go up, go down, or stay the same) in seven core physiological variables in a prediction table, as illustrated in Table 1.

Any incorrect student predictions are then tutored. Some of this tutoring occurs immediately after an incorrect prediction is made, but usually the tutor waits until all seven predictions for a stage have been collected. For pedagogical purposes, the problem is divided into three chronological stages, denoted DR (the Direct Response period before the reflex kicks in), RR (the Reflex Response period right af-

TABLE 1
Prediction Table

	Stage		
Core Variable	*DR*	*RR*	*SS*
Inotropic state	0	–	–
Central venous pressure	–	–	–
Stroke volume	–	–	–
Heart rate	+	0	+
Cardiac output	+	–	+
Total peripheral resistance	0	–	–
Mean arterial pressure	+	–	+

Note. DR = Direct Response period; RR = Reflex Response period; SS = Steady State achieved over time.

ter the reflex begins to function), and SS (the Steady State achieved over time). The cycle of collecting predictions and tutoring is repeated for each stage, with the tutor analyzing the predictions, finding the errors, and embarking on remedial dialogue.

Within one cycle of collecting predictions and tutoring, the structure of the dialogue is less fixed, but it usually follows a recognizable pattern (Freedman, 1996; Freedman & Evens, 1996). It is usually controlled by the tutor's agenda. Most of the tutorial part of the dialogue can be segmented into a sequence of tutoring episodes that tutor incorrectly predicted variables, with an occasional additional topic, such as a summary, that is not immediately triggered by a student error. If the student makes no errors, the tutors generally ask a question or two to test the student's understanding. Within each stage, the variables are discussed in the sequence they are encountered in the solution of the problem.

The idealized structure of a tutorial goal tree is illustrated in Figure 1. The goals at each level are achieved by carrying out a schema composed of a series of goals at the next lower level. In Figure 1, two segments of dialogue are generated for each variable that the student did not predict correctly. *T-introduces-variable* introduces the variable as a referent in the conversation and *T-tutors-variable* does the actual tutoring. In this article we focus especially on the methods, the goals that are used to tutor individual predictions. The method goals are subordinate to the goal *T-tutors-variable.*

Table 2 shows a short episode of tutoring one incorrect prediction, which we annotate with the goal structure illustrated in Figure 2. In this case, *T-tutors-variable* is satisfied by a single method subgoal, *T-shows-contradiction.* The *T-shows-contradiction* goal is carried out by two topics: (a) *T-presents-contradiction,* where the contradiction is laid out for the student, and (b) *T-tutors-contradiction,* where the right answer is obtained by studying the contradiction.

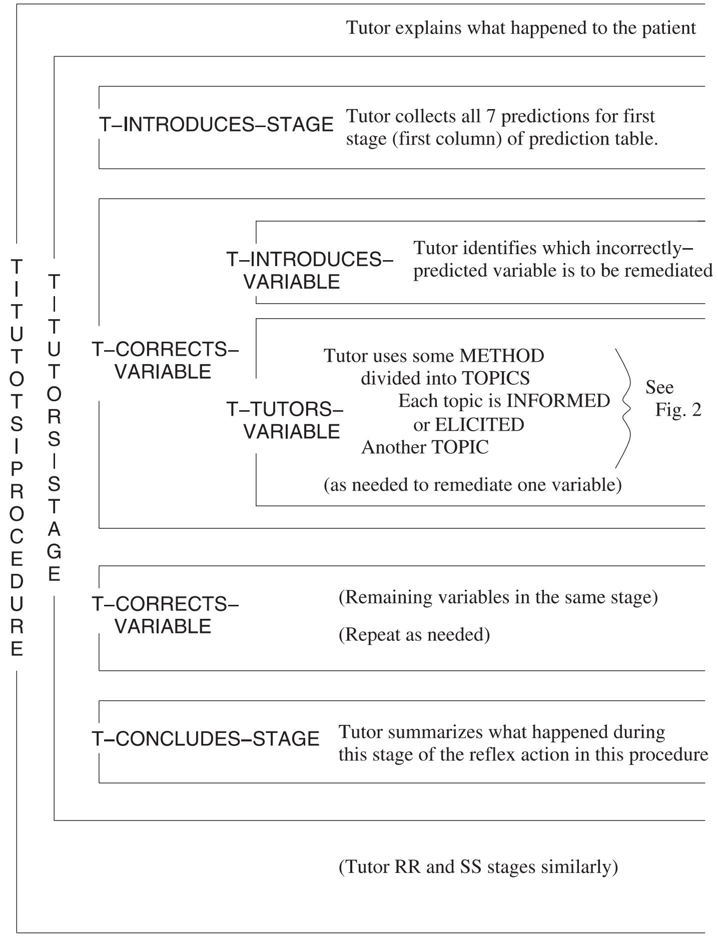
T-TUTOR-PROCEDURE
Tutor explains what happened to the patient
T-TUTORS-STAGE
T-INTRODUCES-STAGE
Tutor collects all 7 predictions for first stage (first column) of prediction table.
T-CORRECTS-VARIABLE
T-INTRODUCES-VARIABLE
Tutor identifies which incorrectly-predicted variable is to be remediated
T-TUTORS-VARIABLE
Tutor uses some METHOD
divided into TOPICS
Each topic is INFORMED
or ELICITED
Another TOPIC
See Fig. 2
(as needed to remediate one variable)
T-CORRECTS-VARIABLE
(Remaining variables in the same stage)
(Repeat as needed)
T-CONCLUDES-STAGE
Tutor summarizes what happened during this stage of the reflex action in this procedure
(Tutor RR and SS stages similarly)

FIGURE 1 Goal structure of tutoring dialogue above the method level.

TABLE 2
Short Episode of Tutoring

Tu:	So, RAP and CC determine SV. You predicted that CC would be unchanged and that RAP increased. How can SV be unchanged?
St:	I thought that the immediate response would be for SV to stay the same and … . [Student gives some incorrect answer.]
Tu:	Not quite. The first beat after the change in pacemaker function … .

Note. K27:68–70. Tu = tutor; St = student; RAP = right atrial pressure; CC = cardiac contractility; SV = stroke volume.

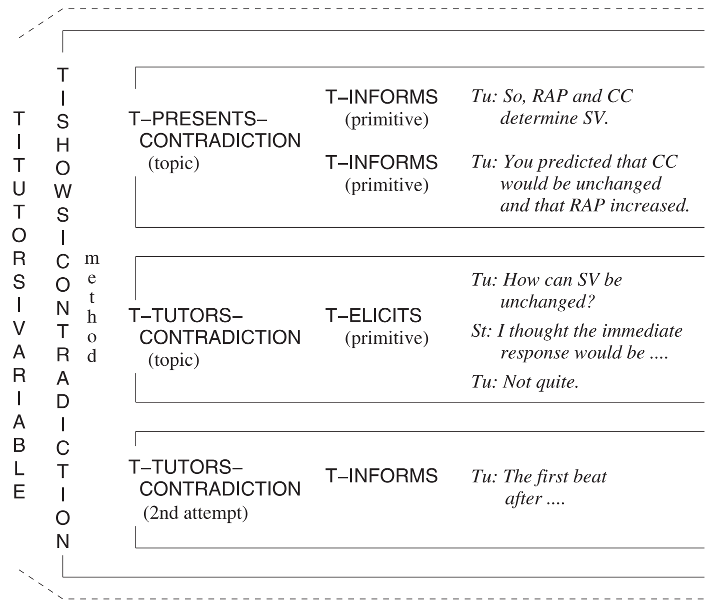
T–TUTORS–VARIABLE
T–SHOWS–CONTRADICTION
method
T–PRESENTS–CONTRADICTION (topic)
T–INFORMS (primitive)
Tu: So, RAP and CC determine SV.
T–INFORMS (primitive)
Tu: You predicted that CC would be unchanged and that RAP increased.
T–TUTORS–CONTRADICTION (topic)
T–ELICITS (primitive)
Tu: How can SV be unchanged?
St: I thought the immediate response would be ....
Tu: Not quite.
T–TUTORS–CONTRADICTION (2nd attempt)
T–INFORMS
Tu: The first beat after ....

FIGURE 2 Method, topic, and primitive goal structure for a short episode of tutoring.

Figure 2 also illustrates another phenomenon. The *T-tutors-contradiction* goal, realized as a question, failed. The tutor repeated this goal; we annotated that as a second attempt. In the second attempt the tutor informed the student of the answer instead of eliciting it. From the beginning of our work on CIRCSIM-Tutor, our expert colleagues, Joel Michael and Allen Rovick, emphasized the importance of discovering and implementing alternative tutoring strategies. Some students respond to one approach and some to another; it is also imperative for the system to have an alternative approach when one approach fails. Woolf (1984) also stressed the importance of having a variety of tutoring strategies and tactics at different levels.

Textual and Interpersonal Attributes of Goals

Halliday (1985) analyzed language as expressing three kinds of meaning simultaneously: experiential meaning, the propositional content of an utterance; textual meaning, the contribution of the utterance to the narrative coherence of the conversation; and interpersonal meaning, the attitude of the speaker. Textual meaning is often expressed by the ordering and juxtaposition of statements, by conjunctions, and by boundary markers such as cue words. In written text, interpersonal meaning is often expressed by lexical items such as “certainly” and “well … .”

The intent of our hierarchical goal structure is to express the propositional content of the dialogue, which is largely captured by the tutorial goal in our application. But it is sometimes necessary to label textual and interpersonal features so that the automated tutor can generate an appropriate utterance in context, just as a human tutor would. Therefore, where required, we enrich individual goals with attributes describing the interpersonal and narrative aspects of the tutor's utterance that the system must take into account. For example, the tutor's second sentence of Figure 2, coded with the primitive goal *T-informs,* also contains attributes *"var = cc, rap," "info = det-value,"* and *"narrative-mode = reference"* (not shown in Figure 2 for the sake of brevity).

The *narrative-mode* attribute, used to clarify the role of an utterance in the structure of the argument, is an example of Halliday's textual meaning. Here are two examples of the *narrative-mode* attribute applied to the same *T-informs* primitive:

- You predicted the CO increased. *narrative-mode = reference*
- So, CO increases. *narrative-mode = summary*

The propositional content of these two statements is largely the same: CO increases. However they also have different rhetorical functions and cannot be substituted for each other in all contexts. The first does not imply that CO actually increased, as we could try to prove otherwise. The second is uttered only after we know for certain that CO did increase, and we could use that fact in an argument about some other variable.

The *attitude* attribute is used to express the tutor's personal stance with respect to the material being uttered, Halliday's interpersonal meaning. We have chosen to annotate interpersonal meaning when it serves to complement the tutor's narrative. For example we code *attitude = remind* when the tutor gives an explicit reminder as in "but remember that this is the DR period." In this case, reminding is not part of the narrative structure of the argument, because the argument would still be coherent if the tutor said "this is the DR period." Here are some examples of the use of *attitude:*

- CO increases. *attitude* unmarked
- But remember that CO increases. *attitude = remind*
- CO certainly does increase. *attitude = bolster-answer*

The *bolster-answer* option is used to respond to a student utterance, expressing the tutor's explicit acknowledgment that the student's statement is true. Sometimes it serves to acknowledge the correct portion of a partially correct answer before addressing the wrong part.

Markup Procedure

The annotated text described in this article was extracted from our first approximately 50 hr of human tutoring. The human tutors were our colleagues, Joel Michael and Allen Rovick, professors of physiology at Rush Medical College. The students were first year medical students enrolled in a physiology class. Tutoring was scheduled at the point when the students had heard lectures and completed readings in cardiovascular physiology and were starting to integrate this knowledge and use it to solve problems. These examples of human tutoring were collected keyboard-to-keyboard style to simulate instruction via the computer: the tutor and the student sit in different rooms and type messages to each other. Communication was mediated by the messaging software Computer Dialogue System (CDS), which allows only one participant to type at a time until that participant relinquishes control to the other (Li, Seu, Evens, Michael, & Rovick, 1992). We have pretest and posttest evidence to show that these keyboard-to-keyboard human tutoring sessions were effective in producing learning gains for students.

The major part of the material described in this section comes from an intensely marked up extract of about 250 turns of dialogue from 13 different sessions, about 5% of the total corpus. We started with all examples of the DR stage of a problem that posits that the patient's pacemaker malfunctions. We eliminated transcripts where the students made no errors. To see more complete examples of tutoring methods, we eliminated text where collecting and correcting predictions were interleaved. Markup was performed by Jung Hee Kim and Reva Freedman and their consensus was vetted by Michael Glass and Martha W. Evens.

This extract was quite rich in tutoring episodes, remediating 26 instances of incorrectly predicted variables. Many of them required several tutorial attempts. The extract contains very little dialogue not directly engaged in tutoring. There were about 500 individually marked segments of text, each on the average about half a turn. We identified nine method goals, representing nine different tutoring strategies, along with associated topics and attributes. Because we discovered many different goals within a small sample size, we did not compute statistics about their relative frequency.

We have also sparsely annotated larger sections of the corpus, which has now grown to more than 70 sessions. The subsequent markup concentrated primarily on tutoring methods we had not studied before such as *T-tutors-via-analogy.*

To assess annotation reliability, two authors independently marked up method and topic goals in the DR stages from three transcripts not otherwise incorporated in this study, a total of 70 dialogue turns. Each annotator segmented the text into as many as 18 method goals. The two coders' identifications of the locations of the method text segments agree, with Cohen's $\kappa = .72$. For the coding of the particular method goal assigned to each segment, $\kappa = .44$. A threshold for moderately good agreement is .67 (Di Eugenio & Glass, 2004). The lower agreement on the method names, as compared to the method segmentation structure, derives from several phenomena. First, for a given level of agreement, Cohen's formula is affected by having a large number of coding categories with some more prevalent than others (Di Eugenio & Glass, 2004). Ten different methods were identified in the sample. Second, new tutoring methods were discovered in what was previously unannotated text. One of them was noticed and annotated by both independent annotators (increasing agreement) and another by only one annotator (decreasing agreement). Finally, because tutors deviate from idealized behavior, reliable coding of the resulting incomplete realizations of ideal methods is difficult.

With regard to reliability of topic goal annotations, the two raters agreed on the topic goals with $\kappa = .56$. In the consensus markup we agreed on text that would be skipped or only lightly annotated; recall that our purpose was not to explain every phenomenon we encountered but to identify tutoring patterns for mechanization. Because the independent raters had no way to agree on what text to omit during the reliability test, we computed agreement statistics on the 18 topic segments they both chose to annotate.

We chose to annotate the transcripts in machine-readable SGML format. The transcript snippet from Table 2 is marked up as shown in Table 3. We chose SGML for three reasons. First, it can be communicated between researchers easily and rapidly. Second, this format makes it easy to generate counts and other summary statistics. Most important, this form of annotation can be used as input for machine-learning programs (Freedman, Zhou, Glass, Kim, & Evens, 1998; Kim, Glass, Freedman, & Evens, 2000).

In the transcript excerpts in this article, the attribution K22:31 indicates turn 31 of keyboard dialogue 22. Where appropriate, we have deleted extraneous material or made other small changes to better illustrate a key point.

TUTORING GOALS

Now that we have described our approach to dialogue analysis and to markup, we present our catalog of the tutorial goals that we identified in our transcripts. Because the topic goals are usually associated with particular method goals, we describe a method and its associated topics together. Table 4 summarizes the method and topic goals in this chapter.

Common Method, Topic, and Primitive Goals

Most method-level goals are realizations of the goal *T-tutors-variable* and give a plan for remediating one incorrectly predicted variable. Methods realize various types of deductive reasoning, interactive questioning, exploration of anomalies, and corrections of misconceptions.

TABLE 3
SGML Annotation of Text in Table 2

```
<T-shows-contradiction>
  <T-presents-contradiction>
    <T-informs info=determinants narrative-mode=summary>
      So, RAP and CC determine SV.
    </T-informs>
    <T-informs info=det-value narrative-mode=reference>
      You predicted that CC would be unchanged and
      that RAP increased.
    </T-informs>
  </T-presents-contradiction>
  <T-tutors-contradiction>
    <T-elicits info=reason>
      How can SV be unchanged?
    </T-elicits>
    <S-ans catg=incorrect>
      [Here the student gave some incorrect answer]
    </S-ans>
    <T-ack catg=incorrect>
      Not quite.
    </T-ack>
  </T-tutors-contradiction>
  <T-tutors-contradiction attempt=2>
    <T-informs info= … .>
      [Here the tutor gives an answer]
    </T-informs>
  </T-tutors-contradiction>
</T-shows-contradiction>
```

Note. SGML = Standard Generalized Markup Language; RAP = right atrial pressure; CC = cardiac contractility; SV = stroke volume.

TABLE 4
Summary of Method Goals and Associated Topic Goals

Method	*Topic*
T-does-neural-DLR	T-tutors-mechanism, T-tutors-DR-info, T-tutors-value, T-tutors-definition
T-tutors-via-determinants	T-tutors-determinants, T-tutors-relationship, T-tutors-value
T-moves-forward	T-tutors-consequence-value
T-explores-anomaly	T-presents-anomaly, T-tutors-anomaly
T-shows-contradiction	T-presents-contradiction, T-tutors-contradiction
T-diagnoses-error	T-identifies-problem
T-tutors-via-analogy	
T-tutors-via-algebraic-approach	
T-tutors-via-negative-reflex-concept	T-invokes-teleology, T-invokes-reflex-consequence
T-moves-toward-PT	
T-tutors-via-deeper-concepts	
T-tutors-equation	
T-prompts-start	
T-summarizes	

Elements of the context that determine the selection of a particular method include the variable to be tutored and the structure of the current subtree of the dialogue. Many methods are applicable only to some variables or types of variables, such as only variables directly controlled by the nervous system. Some methods are applicable only to the first or to a subsequent attempt to teach a variable. For example, the *T-moves-forward* method is possible only if the value of the causally prior variable was previously introduced in the conversation.

Core methods. Here we describe method-level goals that are fundamental to teaching about the baroreceptor reflex.

- *T-does-neural-DLR:* Because the domain of these dialogues is the control of blood pressure mediated by the nervous system, variables directly controlled by the nervous system are treated specially. The *T-does-neural-DLR* method is a question-and-answer approach to teaching this content. DLR stands for *directed line of reasoning* (Sanders, 1995). The following example typifies this method. We have annotated the individual topic goals that together serve to implement the method:

[K10: 29–38]

Tu: Can you tell me how TPR is controlled? [T-tutors-mechanism]
St: Autonomic nervous system.
Tu: And the predictions that you are making are for the period before any neural changes take place. [T-tutors-DR-info]
So what about TPR? [T-tutors-value]
St: No change.

The aforementioned *T-tutors-neural-DLR* method dialogue is divided into the following topics:

- *T-tutors-mechanism:* Conveys the knowledge that the mechanism of control for the variable being remediated is neural. In this example, the tutor elicited that information from the student.
- *T-tutors-DR-info:* Conveys the knowledge that in the DR stage of the reflex response, the reflex has yet to react. In this example, the tutor informed the student of this information.
- *T-tutors-value:* Conveys the corrected prediction for the neurally controlled variable under discussion, in this case elicited from the student.

Another possible topic for neural DLRs is

- *T-tutors-definition:* Ensures that the student knows the definition of the variable or physiological concept in question.

When applied to a neural variable, *T-tutors-definition* has the effect of ensuring that the student knows the mechanism of control. In the following example, the tutor elicits the definition of variable IS, then proceeds to elicit its value:

[K47: 56–60]

Tu: Can you tell me what you think that IS means? [T-tutors-definition]

St: The contractility of the heart caused by preload and sympathetic stimulation.

Tu: Partly correct. … Only the sympathetic effect is called a change in IS.
Then if IS is only affected by changes in sympathetic stimulation of the heart muscle what must be the value of IS in the DR period? [T-tutors-value]

Following the correction of a variable using *T-does-neural-DLR,* it is possible to tutor another neural variable by noticing that it is analogous to the one just discussed. We coded these as *T-does-neural-DLR,* not as *T-tutors-via-analogy* (see later) because the underlying reasoning is the same as just used, not an analogy with a nondomain model. However, we have not yet completely coded these instances into topic goals.

[K1: 30–34]

Tu: What other variables are neurally controlled? [Identify analogous variables]

St: CC, HR.

Tu: Again correct. Now we know that in this guy HR is under the control of the artificial pacemaker.But what would happen to CC? [T-tutors-value]

St: CC 0 [“0” means is “unchanged”].

Tu: Right on!

- *T-tutors-via-determinants:* For nonneural variables the most common schema is *T-tutors-via-determinants.* With this method the tutor corrects the value of a variable by invoking a relation with the core variables that determine its value.

[K22: 40–50]

Tu: What parameter determines the value of RAP? [T-tutors-determinants]

St: CO.

Tu: What relationship do they have? [T-tutors-relationship]

St: Inverse.

Tu: Right, then what is the value of RAP? [T-tutors-value]

The three topics *T-tutors-determinants, T-tutors-relationship,* and *T-tutors-value* constitute a pattern for physiological causal reasoning, a pattern independent of the baroreceptor reflex domain. If, in the execution of this method, students exhibit incorrect or incomplete knowledge of the causal relations, these relations can be tutored.

The *T-tutors-via-determinants* method and its topics *T-tutors-determinants* and *T-tutors-relationship* are linguistically based. Tutors and students usually do not distinguish between different modes of causality in conversation. Sometimes, as in the relation of RAP to CO in the earlier example, there is a long chain of physical cause and effect. Sometimes A determines B by definition. Sometimes there are other competing determinants that go unmentioned. The tutors usually ignore these distinctions. Although conclusions reached in this way may be defeasible, the tutors prefer to use *T-tutors-via-determinants* to establish correct answers first. Oddities in the logic are sometimes explored further, after the correct predictions have been arrived at, by using the *T-explores-anomaly* method shown later.

- *T-moves-forward:* The *T-moves-forward* method is an abbreviation of *T-tutors-via-determinants* that applies when the determinant has already been mentioned in the conversation and is still salient. In general, the tutor uses *T-tutors-via-determinants* for the first causal link in a chain, then *T-moves-forward,* because a second *T-tutors-via-determinants* would result in infelicitous redundancy.

[K36: 170–173]

[The prior discussion is a *T-tutors-via-determinants* method, ending as follows:]
Tu: So what happens to RAP? [T-tutors-value]
St: RAP decreases.
[Now begins the *T-moves-forward* method]
Tu: And if RAP decreases what will happen to SV? [T-tutors-consequence-value]
St: SV will D [=decrease].

Inside the *T-moves-forward* method, the *T-tutors-consequence-value* topic is used to show the value of a variable as a consequence of the value of a determinant that is already within the discourse space.

Open-ended methods. Here we list several tutoring goals that are distinguished by their use of open-ended questions. Due to the difficulty of machine processing of the answers to these questions, implementing them in the current CIRCSIM-Tutor system would be a challenge. Identifying the open-ended methods has been valuable for our understanding of the tutoring process, but we have not annotated the topic goals with as much detail as for the other methods.

- *T-explores-anomaly:* This method is used in cases where the physiological facts appear to be inconsistent, but are nevertheless correct. Its purpose is to ensure that the student truly understands the deeper qualitative relations among the variables. Thus *T-explores-anomaly* does not serve by itself to remediate incorrect predictions, so it does not necessarily occur in the service of *T-tutors-variable.* It is invoked shortly after an apparently anomalous situation is predicted as a result of tutoring a variable, but it can also be invoked when the student made no errors.

[K26: 76–77]

Tu: So, CO decreases even though SV increases. [T-presents-anomaly]
How can you explain this? [T-tutors-anomaly]
St: The decrease due to the lowered heart rate is greater than the increase due to increased stroke volume.

Inside the *T-explores-anomaly* method the topics are *T-presents-anomaly,* which is self-explanatory, and *T-tutors-anomaly,* encompassing an explanation of the apparent anomaly. In the aforementioned example the explanation was elicited from the student. This topic is not annotated in further detail.

- *T-shows-contradiction:* A physiological inconsistency in the student's answers serves as the condition for utilizing the *T-shows-contradiction* method. Its rhetorical structure is the same as tutoring an apparent anomaly, comprised of two topics *T-presents-contradiction* and *T-tutors-contradiction*:

[K25: 150]

Tu: In SS you have predicted that CO and TPR will decrease but also that MAP will be unchanged … [T-presents-contradiction]
How is this possible? [T-tutors-contradiction]

Note here that the question is rhetorical, as the desired response is for the student to change his or her predictions.

- *T-diagnoses-error:* This method is used when the tutor wants to identify the student's problem.

[K27: 50]

Tu: Why do you think that TPR will decrease? [T-identifies-problem]

Further conversation ensues, but aside from annotating the question with the *T-identifies-problem* topic we do not annotate any further topics.

Additional noncausative reasoning methods. The methods described so far have all employed reasoning within the domain of physiological causation of the baroreceptor reflex. However the tutor may employ other persuasive arguments to lead the student to the correct answers. Here we illustrate methods that embody these other types of argument.

- *T-tutors-via-analogy:* Gentner (1998) defined analogies to be "a kind of similarity in which the same system of relations holds across different objects." Lulis et al. (2004a, 2004b) described many instances of our tutors using analogies from domains outside the baroreceptor reflex, or even outside physiology. Here is a fragment from a longer *T-tutors-via-analogy* method likening blood flow to car traffic:

[K64: 57]

St: … (a backflow of blood prior to the heart, and therefore an increase in CVP) and fewer cars coming through … .
Tu: … The increase in CVP caused by a fall in CO is not the result of blood BACKING UP. Everything goes in one direction.
St: Well, slowing down would be a better way to put it then.
Tu: Yes. A traffic jam caused by everybody piling into the same area at once.

This dialogue has the tutor and student switching back and forth, even in mid-sentence, between the cardiovascular and the car domains. The *T-tutors-via-analogy* method is so rich and varied that we have not decomposed it into topic-level goals.

- *T-tutors-via-algebraic-approach:* Although the final, SS stage, predictions can be obtained by examining causal relations between the final values of the variables, it is also possible to obtain a correct answer for a single variable by examining its behavior through time. For example, if a variable is unchanged in DR and down in RR it must be down in SS. We call this method *T-tutors-via-algebraic-approach* because the predictions are represented in writing by plus, minus, and zero symbols, and the reasoning is sometimes explained as a kind of algebraic manipulation.

[K25: 160]

Tu: Note that in SS you can apply some simple logic. If DR is up and RR is up then SS MUST be up…
Tu: Only when DR and RR are different (one up, one down) can you not use logic. In these cases SS almost always follows what happened in DR (because the reflex can't fully correct).

Notice that in this example the tutor gives a rule of thumb for using the algebraic approach (when DR and RR differ, SS follows the DR) along with an explanation as to why the rule of thumb works.

- *T-tutors-via-negative-reflex-concept:* This method is applicable to negative reflex systems in general. It involves two topics: *T-invokes-teleology* reminds the student of the purpose of the baroreceptor reflex and *T-invokes-reflex-consequence* reminds the student that negative reflex systems in the body do not fully compensate for the effects of the perturbation.

[K48: 180–188]

Tu: What does the reflex attempt to accomplish? [T-invokes-teleology]
St: Regulate MAP.
Tu: So, if MAP is increased by the bad pacemaker what will the reflex attempt to do?
St: Decrease MAP.
Tu: And will the reflex completely correct the error that is present?
[T-invokes-reflex-consequence]
St: No.
Tu: So, why is MAP increased in SS?
St: Because the reflex has not completely decreased it.
Tu: Right.

Nested methods. The methods described earlier provide schemata for correcting one variable or exploring one anomaly. After every topic in the method has been successfully communicated to or elicited from the student, the method is completed. But what happens when a topic is not successfully communicated? In particular, what happens when the student gives a wrong or near-miss answer to a question? One action the tutor can take is to correct the topic the student had a problem with, then proceed with the original method. Typically a nested method is a schema that corrects one topic in response to an unexpected answer. Often a nested method refers to a more detailed physiological model. We call these nested methods because they are annotated as being entirely within the tutoring of the topic that provoked the incorrect response. Implementation of this feature, which is found in tutoring systems Atlas–Andes (Rosé et al., 2001), Why2-Atlas (VanLehn, Jordan, Rosé, & the Natural Language Tutoring Group, 2002), Ms. Lindquist (Heffernan, 2001; Heffernan & Koedinger, 2002), AutoTutor (Graesser, Olney, Haynes, & Chipman, 2005), and the CAPE tutor (Freedman, 2001; Freedman et al., 2004), is a way of tailoring responses to the student's individual needs.

- *T-moves-toward-PT:* "PT" in the name of this method stands for "prediction table." This method is used when the student's answer is physiologically pertinent but not what the tutor was looking for. The tutor then tries to obtain an answer to the original question, namely one of the prediction table variables or the baroreceptor reflex itself. In this example a *T-tutors-mechanism* topic elicits a near-miss response, which is rescued by a *T-moves-toward-PT* nested method.

[K12: 37–40]

[*T-tutors-mechanism* begins]

Tu: What is the primary mechanism of control of TPR?

St: Radius of arterioles. [This is pedagogically useful but not the desired answer.]

[*T-moves-toward-PT* nested method occurs here]

Tu: Yes. And what is the primary mechanism by which arteriolar radius is controlled?

St: Sympathetics.

[*T-tutors-mechanism* ends]

T-moves-toward-PT can be employed in the service of a number of other topics whenever the tutor or student has introduced a deeper conceptual model involving nonprediction table variables. Eventually the reasoning must return to the core variable that started the discussion.

- *T-tutors-via-deeper-concepts:* This method tutors in terms of a more detailed physiological model after the tutor has failed to elicit a correct answer from the student.

[K11: 65–68]

Tu: How about the influence of a change in CO on RAP?

St: I [=increased] CO -> I venous pressure

Tu: I CO decreases the volume of the central veins by transferring larger volumes of blood into the arterial system (that's why the MAP goes up —larger arterial blood volume). If the volume of blood in the central veins decreases, what would happen to central venous pressure?

St: D [=decreased]

- *T-tutors-equation:* To correct student misconceptions about a causal relation between variables, the tutor often asks for the related equation using *T-tutors-equation.*

[K24: 96–98]

Tu: Can you write the equation relating MAP, CO, and TPR?

St: MAP = CO × TPR.

Tu: Right.

This method is useful because it establishes much of the information needed for qualitative reasoning: which variables determine the target variable, and whether the relations are direct or inversely proportional. It also serves as an alternate way to prompt the student, as the following example shows.

[K33: 118–121]

Tu: What are the determinants of MAP?
St: SV, Contractility
Tu: Try again, write an equation that reads MAP = …… .
St: CO, TPR

Nonremediative methods. These methods do not remediate any specific incorrect prediction but they serve other pedagogical or narrative purposes. As noted earlier, the *T-explores-anomaly* method is also used in this way.

- *T-prompts-start:* This method is used at the beginning of a stage to help the student get started.

[K3: 63–64]

Tu: Begin making your predictions starting with the variables that are first affected by the reflex.
St: Vasodilation would cause TPR d [=decreases].
Tu: Good.

- *T-summarizes:* The tutor sometimes summarizes progress, especially when the student has made several errors in one stage.

[K38: 72]

Tu: So, let me summarize the results that we have been discussing. (You might want to write these down if you haven't kept up as we have been conversing.) TPR is up (the drug does that directly) and hence MAP is up. This causes SV to decr and thus CO is down. The decrease in CO causes RAP to incr. And CC and HR as neural variables don't change.

This example summarized about 50 turns of dialogue. Shorter summaries sometimes occur within the tutoring of a single *T-tutors-variable* goal. In some situations the tutor elicits a summary, item by item, using a DLR strategy (Sanders, 1995).

Primitive goals. The lowest level goals of a schema are the primitive goals *T-elicits* and *T-informs,* each corresponding roughly to a single tutor question or statement. The *T-elicits* goal is used when we want the student to participate actively by answering a question. With *T-informs* the tutor gives some information to the student. Following are some examples.

- *T-elicits info = var-value*

[K10: 53]

Tu: So what's your prediction of CC in the DR?

- *T-informs info = DR-info, attitude = remind*

[K12: 35]

Tu: Remember that we are dealing with the short period before you get a reflex response.

- *T-informs narrative-mode = summary*

[K27: 72]

Tu: So, we have HR down, SV up and CO down.

The various attributes annotated on the primitive and higher level goals are described next.

Categories of Attributes

As noted earlier, the SGML attributes in the examples of primitive goals shown earlier sketch the content to be uttered (the *info* attribute in the aforementioned examples) and aspects of the form and style of the utterance (the *attitude* = *remind* and *narrative-mode* = *summary* attributes in those examples). The *remind* attitude shows the tutor's intention and the *summary* narrative mode specifies a particular narrative form.

Here are some of the attributes that we have used in coding the transcripts. We have organized them according to Halliday's three kinds of meaning (Halliday, 1985).

Attributes related to experiential meaning. These attributes describe the propositional content of the utterance.

- *Variable:* We have seven core variables: HR, IS, TPR, CO, MAP, CVP, SV. There are other variables that appear infrequently. Some older transcripts use CC in place of IS and RAP in place of CVP.
- *Information:* This attribute specifies the bulk of the propositional content delivered by one topic. Some examples are:

[K10: 41]

info = var-value	You predicted that CC would go up.

[K10: 45]

info = starling's law	Increased filling (preload) does increase ventricular contractile performance; but this is the cardiac length/tension relationship (Starling's Law), not contractility.

- *Time-qualifier:* The time qualifier specifies a stage of the reflex response, DR, RR, or SS.

Attributes related to textual meaning. These attributes describe the narrative form and structure of the utterance.

- *Attempt Number:* This number specifies how many attempts have been made to satisfy the current topic goal within a method, or to satisfy the current method. The default value is 1. The attempt number is an attribute of narrative structure; additional attempts after the first are usually expressed differently than the first attempt.
- *Discourse Marker:* Discourse markers are connectives that specify the rhetorical relation between utterances. For example, *since* signals that the following utterance is in support of some other utterance. The *and* in the following example indicates that the following utterance is the next logical step in a chain of reasoning.

[K48: 48]

DM = "and"	And during DR what changes in ANS activity occur?

[K14: 45]

DM = "since"	Since you predicted that CO I, what will RAP do?

Discourse markers are consistent with our goal structure annotations (Kim et al., 2000), often serving to introduce topic-level goals and to show the rhetorical structure of the tutor's argument.

- *Narrative-mode:* Narrative mode is not shown in the unmarked case. *Narrative-mode = summary* is used to indicate that the text is used as a summary. *Narrative-mode = reference* indicates that the tutor is referring to something the student said, rather than asserting a new proposition.

[K25: 62]

narrative-mode = summary — So, in DR HR is up, CO is up, but SV is down.

[K14: 53]

narrative-mode = reference — You predicted that RAP would in fact go down.

- *Context-setting clause:* The context-setting clause is most frequently used to restate information that is already known, in order to provide context for the next utterance. Because it usually does not add any propositional content to the conversation we have classified it as specifying textual meaning.

[K25: 60]

context-setting-clause =
“if CVP is down” — So, if CVP is down what happens to RAP and SV?

Attributes related to interpersonal meaning. Interpersonal meaning is most often marked on primitive goals that directly engage a student, either a question put to the student or the first sentence of the response addressing the student’s answer. Usually they do not change the propositional content of an utterance, but just how it is expressed.

- *Softener:* This attribute is used when the tutor recasts questions in more polite or face-saving language. For example, *softener = “do you know”* converts the question “how is TPR controlled” to “do you know how TPR is controlled?” Other typical softeners we have observed are “do you think that” and “can you tell me.” More examples can be found in Kim et al. (1998a) and Kim (2000).
- *Attitude:* Here is an example of a question/answer/response exchange in service of a *T-tutors-determinants* topic goal, illustrating several values of the *attitude* attribute:

[K20: 34–36]

Tu: What are the determinants of SV? [T-elicits info = determinants, var = sv]
St: Determinants are end-diastolic volume, afterload i.e., MAP, …. [S-ans]
Tu: Well that’s partly correct. [T-ack]
EDV is certainly a determinant. [T-informs attitude = bolster-answer]
Afterload (i.e., aortic pressure) is important but only when….
[T-informs attitude = qualify-answer]

In this example, the *bolster-answer* attitude is expressed by the word “certainly,” which distinguishes the tutor’s stance toward the student as different than a simple recitation of fact. The tutor's acknowledging that afterload is an important determinant in other situations, instead of merely rejecting it, earns the next utterance a *qualify-answer* attitude.

Many of our attitude attributes are similarly marked by characteristic words or phrases. Here are more examples:

[K14: 49]

attitude = rephrase-question "What I was asking is …"

[K22: 46]

attitude = repeat-answer "You are correct, both of these would alter RAP." Without the *repeat-answer* attitude, the tutor issues a bare acknowledgment.

[K10: 43]

attitude = remind "But remember that we are dealing with the period before there can be any neural changes." The word "remember" is a good marker for this attitude.

DISCUSSION

Hierarchical Versus Linear Description

Analysis of educational dialogues began with an exchange structure view (Sinclair & Coulthard, 1975), where utterances are characterized as cycles of initiate, response, and feedback, and much of the text is described as a linear sequence of these moves. By concentrating on pairs of turns, the exchange structure view permits precise tracking of the way each speaker adapts to the other as the dialogue evolves. This linear, sequential style of analysis, combined with the insight of Clark's (1996) work on joint actions, has enabled us to understand both the production of more varied and more appropriate acknowledgments (Brandle, 1998), and the decision to leave some implicit.

As we noted in the description of our annotation model, however, when dialogue is annotated for the purpose of informing computer generation, some description of goal structure is essential. More recent markup schemes such as Discount (Pilkington, 1999) and DAMSL (Allen & Core, 1997) enhance the traditional exchange structure approach by building up goal structures in layers on top of the exchange structure. Wells (1999) also used the initiate/response/feedback cycle as the basis for higher levels of structure (moves, sequences, and episodes), although his scheme is not intended to be used for computation. In our scheme the content-based hierarchical nature of the tutor's goals is fundamental. The linear exchange structure view is most useful to us in understanding the immediate responses to student turns, so we annotate these in the transcripts by interpolating annotations of the student's responses and the tutor's acknowledgments at the points where they occur.

Generalizing The CIRCSIM = Tutor Goals

The method-level goals were derived for tutoring in a particular problem domain. Thus some of them have small niches of applicability whereas others are more broadly useful. Three questions you can ask about a method level goal are (a) What is its scope of applicability? (b) Does it capture a restricted example of a more generally useful goal? and (c) What kind of information is needed to use it?

The term DLR was coined by Sanders (1995) in his dissertation to describe a generalization of Michael and Rovick's method of delivering interactive explanations, summaries, and remediation of misconceptions. Cawsey (1992) and Fox (1993) described the same type of behavior in the tutors that they studied. *T-does-neural-DLR* describes a DLR, a fairly general tutoring pattern. Because it has most of the facts built in for a particular piece of remedial material in our domain, it has no applicability outside of the autonomic nervous system. Very little extra information is needed to utilize *T-does-neural-DLR* in a tutoring system. A less specific DLR method would have greater applicability, but would require more domain information every time it was used.

The methods *T-tutors-via-determinants* and *T-moves-forward* appeal directly to the underlying causal model of the domain and should apply in any domain that uses qualitative causal reasoning. *T-tutors-via-determinants* involves a kind of backward chaining that is common in many types of expert systems. *T-moves-forward* is designed to push the student forward along the causal chain. The two methods will apply in any domain where the goal is to teach the student to solve problems using qualitative causal reasoning, such as physics, electronics, process control systems, or troubleshooting methods for equipment. For example, we have observed the same methods in tutoring dialogues in the domain of respiratory physiology.

T-tutors-equation and *T-tutors-via-deeper-concepts* also apply to teaching in qualitative causal reasoning domains. *T-tutors-equation* serves to remind students of qualitative relations by appealing to their prior knowledge of a quantitative model. *T-tutors-via-deeper-concepts* allows the tutor to invoke any domain relation that might persuade the student of the correct answer. For example, in our transcripts the tutors appeal not only to ancillary physiological variables but also to related anatomical structures. These methods are not specific to the baroreceptor reflex; the tutor needs information from the domain model to apply them.

Employing *T-explores-anomaly* and *T-shows-contradiction* requires a subject area where the reasoning is falsifiable. Any scientific, technical, or mathematical field should qualify.

T-tutors-via-negative-reflex-concept and *T-tutors-via-algebraic-approach* embody arguments pertinent to simple negative feedback or closed loop control systems common in nature and in technology. One reason for building CIRCSIM-Tutor is to reinforce in medical students the correct reasoning about the many regulatory mechanisms in the human body. As we have described them, these tutorial methods are specialized for domains similar to ours: not every feedback system's response is conceptualized as three stages nor is every feedback system regulated by a neural reflex. The generalized versions of these methods would require only a little domain-specific information to be fully useful.

The methods *T-moves-toward-PT, T-prompts-start, T-summarizes,* and *T-diagnoses-error* all represent common tutorial dialogue moves in task-oriented dialogues. *T-moves-toward-PT* in our markup requires the name of the physiological variable being tutored. It is a specific case of the more general tutorial goal of moving from the current discussion to answering the original question put to the student. Teaching by using analogies (Kurtz, Miao, & Gentner, 2001), coded as *T-tutors-via-analogy,* also occurs in many domains.

Comparison With Other Analyses of Tutoring Strategies

Graesser, Person, and Magliano's (1995) catalog of pedagogical strategies. In the course of studying "normal" tutors (e.g., upper level students teaching lower level students), Graesser et al. (1995, p. 497) searched for evidence of these eight components of effective pedagogy drawn from current theories in the literature:

1. Active student learning
2. Sophisticated pedagogical strategies
3. Anchored learning in specific examples and cases
4. Collaborative problem solving and question answering
5. Deep explanatory reasoning
6. Convergence toward shared meanings
7. Feedback, error diagnosis, and remediation
8. Affect and motivational strategies

They found that Items 3, 4, and 5 were most prominent in the tutoring dialogues that they examined, whereas the others were "underdeveloped, defective, or virtually non-existent" (p. 497).

We also find 3, 4, and 5 to be prominent. The tutoring sessions are structured by solving specific cases (Item 3). The tutors often employ deep physiological and anatomical reasoning and try to ensure that students understand the topic at that level (Item 5). Although "collaborative problem solving" in a classroom setting most often refers to having small groups of students work together, the salient feature is that the several parties are engaging in a conversation where the shared conversational goal is the construction of a solution. Our dialogues map onto this description nicely (Item 4), as did the dialogues of Graesser's study.

Our tutors Joel Michael and Allen Rovick frequently manifest Socratic teaching, one of the sophisticated pedagogical strategies (Item 2). In a controlled comparison of our tutors with less experienced tutors similar to the "normal" tutors, we found that our experienced tutors devote a much larger fraction of their utterances to asking questions of the students (Glass et al., 1999). We take this to be quantitative evidence that the experienced tutors use a more Socratic style. Although many of the method and inner-method goals such as *T-shows-contradiction* and *T-tutors-via-analogy* do not a priori need to be expressed in a Socratic style, they are usually wielded as efforts to Socratically coax correct answers out of students. We would also assert that teaching by analogy is a cognitively sophisticated strategy. Lulis et al. (2004a, 2004b) discussed and tabulated the analogies found in our transcripts.

Our tutors frequently diagnose and remediate student misconception (Item 7). Indeed, *T-diagnoses-error* is among the tutoring methods in our catalog. Although our tutors often attend to student affect (Item 8), many of the motivational strategies that Lepper, Woolverton, Mumme, and Gurtner (1993) described have no opportunity to appear in our sessions. For example, the confidence-building strategy of describing the first problem as difficult, even though it is easy, cannot occur when there are only one or two problems in a 1-hr session. Active student learning (Item 1), as a pedagogical technique, entails that the student have control over the pedagogical agenda. In our dialogues they do not. The only time the student has control is when seizing the conversational initiative. In our dialogues these are temporary interruptions. Evidence of convergence toward shared meanings (Item 6) is hard to quantify. According to this theory, tutoring ideally should result in a meeting of the minds. Graesser et al. (1995) reported that they were "pessimistic," noting that sometimes tutor and student even had divergent goals. We have not tried to measure this phenomenon in our transcripts.

In total, in our dialogues with expert tutors we find strong use of Items 2 and 7 in addition to 3, 4, and 5, and much weaker expressions of 1 and 8. The "normal" tutors of (Graesser et al., 1995) were knowledgeable students without extensive tutoring experience. It may be that tutoring expertise is manifested partly in the ability to use sophisticated strategies more often (in particular, Socratic tutoring or similar ones) and to diagnose and remediate student misconceptions.

Tutoring strategies identified by Chi, Siler, Jeong, Yamauchi, and Hausmann (2001). Chi et al. (2001) studied 11 novice tutors in the biomedical domain and coded each substantive statement by their tutors into one of eight categories (p. 489):

1. Giving explanations
2. Giving direct (positive or negative) feedback
3. Reading text sentences aloud
4. Making self-monitoring comments
5. Answering questions that the students asked
6. Asking content questions
7. Scaffolding with generic and content prompts
8. Asking comprehension-gauging questions (such as "Is this starting to stick?")

Almost all of the activities of our expert tutors belong to Categories 2, 5, 6, and 7. They rarely give explanations except in the context of direct feedback (Category 2) or when answering questions of the student (Category 5). Instead, they elicit explanations from the student. They do not read text sentences aloud, but they do ask students to read problem statements and instructions. They never make self-monitoring statements to students, though they sometimes make them to each other or to the CIRCSIM-Tutor team.

Category 8 is the most controversial. The Coach system of Winkels and Breuker (1990) has strategies called "check understanding" and "check acquisition," which are expanded into questions like "Do you understand what I just said?" and "Are you still with me?" CIRCSIM-Tutor does not ask such comprehension gauging questions because Rovick and Michael believe they are a waste of time. The only valid way to check for understanding, they say, is to ask a substantive question. Comparison of their tutoring sessions with novice tutoring sessions shows that novice tutors ask such questions all the time, but experts much less so (Glass et al., 1999). Graesser (1993) gave a solid foundation to our expert tutors' intuition by demonstrating that only the "A" students tend to accurately report whether they understand the material or not, whereas Graesser et al. (1995) illustrated that the answers to comprehension gauging questions do not correlate well with the truth.

Wells's categories of functions. Wells (1999) coded educational dialogues using a system based on an exchange structure view (Sinclair & Coulthard, 1975). The top level in his scheme is an episode, each episode is segmented into sequences, sequences are segmented into exchanges, and exchanges are segmented into moves. Possible moves include *Initiate, Respond,* and *Follow-up.* Exchanges are further structured in relation to each other; for example, they are marked as nuclear or dependent on others. Moves and exchanges have functions and many of those functions (Wells, 1999, pp. 337–338) correspond to the lower levels of goals described here and in our article on student initiatives (Shah et al., 2002). For example, Shah et al.'s conversational repair corresponds to Wells's *Request Repeat* and *Repetition.*

Wells thus builds up a structured assemblage of goals describing the text, much as we do. That many of our method and topic goals are domain-specific, whereas Wells's are not, elucidates the difference. When it comes to hierarchical labeling, Wells annotates conversational structure, we annotate pedagogical content goals. Our content-specific tutoring method goals would largely be lumped together by Wells as *Give Suggestion.* On the other hand, Wells devotes much attention to the protocols of dialogue, such as *Bid* (request to speak) and *Nominate* (the next speaker) that are obviated by our two-person computer-enforced turn-taking setup. Although we do annotate the student's responses and the tutor's acknowledgment of those responses, equivalent to Wells's *Respond,* these elements are not included in our hierarchical structure.

Comparison With Strategies in Other Intelligent Tutoring Systems

Dialogue schemata in Atlas–Andes. The current tutoring system with goals most like ours is the Atlas–Andes tutor developed at the University of Pittsburgh under the direction of VanLehn (Rosé et al., 2001). Andes is a model-tracing physics tutor and Atlas–Andes is an expanded version with a natural language dialogue component. The planning and natural language dialogue generation components are based on the APE dialogue planner (Freedman, 2000b; Freedman et al., 2004; Freedman et al., 2000). Parsing is done by the CARMEL parser (Freedman et al., 2000; Rosé, 2000). In Atlas–Andes, discourse schemata corresponding to the method level in our dialogues are called Knowledge Construction Dialogues (KCDs). KCDs are designed to teach principles in a Socratic way in a problem-solving environment. KCDs also include plans for responding to various student errors. If a student draws an acceleration vector in the wrong direction, the KCD might begin: "What is the definition of acceleration?" If the student responds appropriately, the next question might be about the values of the terms used in the definition. If, instead, the student is totally confused, the tutor might choose a different KCD to teach the same material via analogy (Freedman, 2000a; Rosé et al., 2001). KCDs also include topic-level utterances aimed at individual errors and misconceptions. For example, if a student solving an elevator problem believes that the acceleration and the velocity must go in the same direction, the system might reply, "But if the acceleration went the same direction as the velocity, then the elevator would be speeding up." Some topic-level utterances are aimed at discourse goals rather than at pedagogical goals. For example, to indicate that the tutor wants the student to give another answer, Atlas–Andes might say, "Try again."

Tutoring strategies in Heffernan's Ms. Lindquist. Heffernan (2001; Heffernan & Koedinger, 2002) developed a KCD-like system for tutoring algebra via natural language dialogue. His system, Ms. Lindquist, is focused on helping students learn to solve algebra word problems. It concentrates on symbolization, or the act of translating problem statements into algebraic notation. Thus it has a problem-solving focus, although the style of causal reasoning is different from our dialogues. Ms. Lindquist has four general strategies that seem to correspond to our method-level goals: Concrete Articulation (doing an example with numbers before trying to symbolize), Explain Verbally, Decomposition and Substitution, and Study Worked Example. These strategies are rather different than the strategies seen in Atlas–Andes or in our dialogues, perhaps due to differences in the symbolization task and the algebra domain. Heffernan's topic-level goals, on the other hand, seem to be closer to the kind of dialogue that we have seen. Some of the goals that Heffernan uses (Heffernan, 2001, p. 98, Tables 4 and 5) include:

- Op 7: Asking a student to recall a definition
- Op 14: Positive feedback on parts that are correct (an option in our *T-ack*)
- Op 15: Simple feedback on an identifiable bug category (our *T-does-neural-DLR*)
- Op 16: Asking the student to figure out what subgoal to set
- Op 17: Socratic technique showing a contradiction from a student's error (our *T-shows-contradiction*)

Dialogue moves in AutoTutor. The original AutoTutor (Graesser, Person, Harter, & the Tutoring Research Group, 2001; Person, Graesser, Kreuz, Pomeroy, & the Tutoring Research Group, 2001) is a script-based intelligent tutoring system that was designed to teach basic computer literacy. Scripts and enhancements have been written for other subjects as well. The newest version of AutoTutor employs a sophisticated dialogue planning engine (Graesser et al., 2005) that we discuss briefly later. AutoTutor uses the following dialogue moves:

1. Major question, including *why* and *how* questions
2. Short feedback
3. Pumping for more information, for example, asking "What else?"
4. Prompting for specific information, such as making an assertion with a blank in it
5. Hinting
6. Elaborating on a student answer
7. Splicing in correct content after a partially correct student answer
8. Summarizing

The AutoTutor script contains a list of the major content questions along with the desired answers. It also contains the possible questions in each of the other categories and their correct answers.

Because the AutoTutor content is stored in a script, AutoTutor dialogue moves do not refer to specific content. This makes AutoTutor easier to retarget to new domains, but it also restricts the types of dialogue moves that early versions of AutoTutor could make. Thus AutoTutor had no equivalent for the multi-turn directed lines of reasoning that are characteristic of CIRCSIM-Tutor, such as *T-does-neural-DLR* and *T-tutors-via-determinants.* Similarly, AutoTutor did not have the capacity to handle arbitrarily nested goals. AutoTutor does, however, have a built-in multi-turn dialogue strategy that consists of posing a question, eliciting better and better answers, then summarizing. AutoTutor and CIRCSIM-Tutor thus both achieve the net effect of putting a question on the table, then asking questions until the student gives the correct answer or the tutor provides it. Truly open-ended pumping is rarely if ever attested in our transcripts because the original questions are rarely open-ended.

Many of the AutoTutor dialogue moves are attested in our transcripts but are not explicit goals. For example "give hint" is not an explicit goal in our annotation scheme, however particular kinds of tutoring goals such as *T-shows-contradiction, T-tutors-via-negative-reflex-concept,* and *T-tutors-via-deeper-concepts* can look like hints when manifested in dialogue where some of their component topics are informed and some are elicited.

CONCLUSIONS

This article analyzes the goal structure of human tutoring dialogues in physiology. These dialogues were collected in order to construct a dialogue-based intelligent tutoring system, CIRCSIM-Tutor. The tutors were professors of physiology who were also expert tutors, and the students were first year medical students. In this article we analyzed the hierarchical goal structure of the tutorial conversation. We described the major goals that we found and the tactics used to realize those goals. In addition to propositional content, we described the textual and interpersonal attributes needed to represent the goals. We compared our approach to approaches based on an exchange structure view. We described the relation between the goals we found and the goals identified in other analyses of human tutorial dialogues, especially those of Graesser et al. (1995), Chi et al. (2001), and Wells (1999). Finally, we compared our goal structure with the goals used in several prominent intelligent tutoring systems, including Atlas–Andes (Rosé et al., 2001), Ms. Lindquist (Heffernan, 2001; Heffernan & Koedinger, 2002), and AutoTutor (Graesser et al., 2005; Person et al., 2001).

The original CIRCSIM-Tutor had a dialogue planner that was based on a finite state machine; current developmental versions are based on advanced hierarchical planning as described earlier. The original AutoTutor was designed with a single fixed tutoring method, later a finite state dialogue model was added, and now it contains a sophisticated planner that can handle nested multiturn dialogue schemata similar to those seen in this article. Similarly, the dialogue-based physics tutor Atlas–Andes (Rosé et al., 2001) also contains a hierarchical plan manager. Larsson and Traum's (2000) widely adopted dialogue planning infrastructure, TrindiKit, is capable of building multistep plans of hierarchical goals, abandoning or modifying those plans in the same manner as human tutors do while simultaneously supporting finite state exchange structure dialogue moves. Although the annotation of hierarchical multi-turn tutorial goals that we describe here does not model all the phenomena found in tutorial dialogue, it seems that the trend in tutorial dialogue generation systems is to increasingly incorporate such models as a component of dialogue planning.

Further issues of goal structure analysis, pedagogical questions, and linguistic questions suggest interesting avenues for further research into the annotation of tutorial dialogues. We have not fully analyzed how the tutor chooses among the method goals and decides when to abandon them and when to use nested methods. Pedagogically, we are interested in learning when and how the tutors respond to correctable student errors and misconceptions, generalizing the work of Zhou et al. (1999a, 1999b). Linguistically, we need to know how to generate coherent text, given that the tutor may typically amalgamate an acknowledgment, a reply to an incorrect student utterance, and a continuation of a tutorial method into a single turn. Tutorial goal structure is only part of a very large picture.

ACKNOWLEDGMENTS

This work was partially supported by the Cognitive Science Program, Office of Naval Research under Grant 00014–00–1–0660 to Stanford University as well as Grants N00014–94–1–0338 and N00014–02–1–0442 to Illinois Institute of Technology, N00014–03–1–0037 to Valparaiso University, and N00014–02–1–0164 to North Carolina A & T State University. The content does not reflect the position or policy of the government and no official endorsement should be inferred.

We are most grateful to Art Graesser and two anonymous reviewers, whose guidance made all the difference.

REFERENCES

Allen, J., & Core, M. (1997). *DAMSL: Dialogue act markup in several layers* (Tech. Rep.). Retrieved July, 2005, from the University of Rochester, Department of Computer Science: http://www.cs.rochester.edu/research/cisd/resources/damsl

Austin, J. L. (1962). *How to do things with words*. Oxford, England: Oxford University Press.

Brandle, S. S. (1998). *Using joint actions to explain acknowledgment in tutorial discourse: Applications to intelligent tutoring systems.* Unpublished doctoral dissertation, Illinois Institute of Technology.

Bratman, M. (1987). *Intentions, plans, and practical reason.* Cambridge, MA: Harvard.

Carbonell, J. R. (1970). AI in CAI: An artificial intelligent approach to computer-aided instruction. *IEEE Transactions on Man-Machine Systems, 11*(4), 190–202.

Cawsey, A. (1992). *Explanation and interaction: The computer generation of explanatory dialogues.* Cambridge, MA: MIT Press.

Chi, M., Siler, S. A., Jeong, H., Yamauchi, T., & Hausmann, R. G. (2001). Learning from human tutoring. *Cognitive Science, 25,* 471–533.

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

Dickinson, C. J., Goldsmith, C. H., & Sackett, D. L. (1973). MACMAN: A digital computer model for teaching some basic principles of hemodynamics. *Journal of Clinical Computing, 2*(4), 42–50.

Di Eugenio, B., & Glass, M. (2004). The kappa statistic: A second look. *Computational Linguistics, 32*(1), 95–101.

Fox, B. (1993). *The human tutorial dialogue project: Issues in the design of instructional systems*. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Freedman, R. (1996). *Interaction of discourse planning, instructional planning, and dialogue management in an interactive tutoring system.* Unpublished doctoral dissertation, Northwestern University.

Freedman, R. (1997). Degrees of mixed-initiative interaction in an intelligent tutoring system. In *Computational models for mixed initiative interaction: Papers from the AAAI Spring Symposium, Palo Alto* (pp. 44–49). Menlo Park, CA: AAAI Press.

Freedman, R. (2000a). Plan-based dialogue management in a physics tutor. In *Proceedings of the Sixth Applied Natural Language Processing Conference (ANLP 2000), Seattle* (pp. 52–59). New Brunswick, NJ: Association for Computational Linguistics.

Freedman, R. (2000b). Using a reactive planner as the basis for a dialogue agent. In J. Etheredge & B. Manaris (Eds.), *Proceedings of the 13th International Florida Artificial Intelligence Research Society Conference (FLAIRS 2000), Orlando* (pp. 203–208). Menlo Park, CA: AAAI Press.

Freedman, R. (2001). An approach to increasing programming efficiency in plan-based dialogue systems. In J. D. Moore, C. L. Redfield, & W. L. Johnson (Eds.), *Proceedings of the 10th Conference on Artificial Intelligence in Education (AI-Ed 2001), San Antonio* (pp. 200–209). Amsterdam: IOS Press.

Freedman, R., & Evens, M. W. (1996). Generating and revising hierarchical multi-turn text plans in an ITS. In C. Frasson, G. Gauthier, & A. Lesgold (Eds.), *Intelligent tutoring systems: Third International Conference (ITS '96), Montreal* (Lecture Notes in Computer Science No. 1086, pp. 632–640). Berlin: Springer.

Freedman, R., Haggin, N., Nacheva, D., Leahy, T., & Stilson, R. (2004). Using a domain-independent reactive planner to implement a medical dialogue system. In *Dialogue systems for health communication: Papers from the AAAI Fall Symposium, Palo Alto* (pp. 24–31). Menlo Park, CA: AAAI Press.

Freedman, R., Rosé, C. P., Ringenberg, M., & VanLehn, K. (2000). ITS tools for natural language dialogue: A domain-independent parser and planner. In G. Gauthier, C. Frasson, & K. VanLehn (Eds.), *Proceedings of the Fifth International Conference on Intelligent Tutoring Systems (ITS 2000), Montreal* (Lecture Notes in Computer Science No. 1839, pp. 433–442). Berlin: Springer.

Freedman, R., Zhou, Y., Glass, M., Kim, J. H., & Evens, M. W. (1998). Using rule induction to assist in rule construction for a natural-language based intelligent tutoring system. In *Proceedings of the 20th Annual Conference of the Cognitive Science Society, Madison* (pp. 362–367). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Gentner, D. (1998). Analogy. In W. Bechtel & G. Graham (Eds.), *A companion to cognitive science* (pp. 107–113). Oxford, England: Blackwell.

Glass, M. (1999). *Broadening input understanding in a language-based intelligent tutoring system.* Unpublished doctoral dissertation, Illinois Institute of Technology.

Glass, M. (2001). Processing language input for an intelligent tutoring system. In J. D. Moore, C. L. Redfield, & W. L. Johnson (Eds.), *Proceedings of the 10th Conference on Artificial Intelligence in Education (AI-Ed 2001), San Antonio* (pp. 210–221). Amsterdam: IOS Press.

Glass, M., Kim, J. H., Evens, M. W., Michael, J. A., & Rovick, A. A. (1999). Novice vs. expert tutors: A comparison of style. In U. Priss (Ed.), *Proceedings of the 10th Midwest Artificial Intelligence and Cognitive Science Conference (MAICS '99), Bloomington, IN* (pp. 43–49). Menlo Park, CA: AAAI Press.

Graesser, A. C. (1993). Dialogue patterns and feedback mechanisms during naturalistic tutoring. In *Proceedings of the 15th Annual Conference of the Cognitive Science Society, Boulder* (pp. 126–130). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Graesser, A. C., Olney, A., Haynes, B. C., & Chipman, P. (2005). AutoTutor: A cognitive system that simulates a tutor that facilitates learning through mixed-initiative dialogue. In C. Forsythe, M. L. Bernard, & T. E. Goldsmith (Eds.), *Cognitive systems: Human cognitive models in systems design.* Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Graesser, A. C., Person, N. K., Harter, D., & the Tutoring Research Group (2001). Teaching tactics and dialog in AutoTutor. *International Journal of Artificial Intelligence in Education, 12,* 257–279.

Graesser, A. C., Person, N. K., & Magliano, J. P. (1995). Collaborative dialogue patterns in naturalistic one-to-one tutoring. *Applied Cognitive Psychology, 9,* 495–522.

Grosz, B. J. (1977). Representation and use of focus in a system for understanding dialogs. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence (IJCAI '77), Cambridge, MA* (pp. 67–76). San Francisco: Morgan Kaufmann.

Grosz, B. J., & Sidner, C. L. (1986). Attention, intentions and the structure of discourse. *Computational Linguistics, 12*(3), 175–204.

Halliday, M. A. K. (1985). *An introduction to functional grammar.* London: Longman.

Halliday, M. A. K., & Hasan, R. (1976). *Cohesion in English.* London: Longman.

Heffernan, N. (2001). *Intelligent tutoring systems have forgotten the tutor.* Unpublished doctoral dissertation, Carnegie Mellon University.

Heffernan, N., & Koedinger, K. (2002). An intelligent tutoring system incorporating a model of an experienced human tutor. In S. Cerri, G. Gouardères, & F. Paraguaçu (Eds.), *Proceedings of the Sixth International Conference on Intelligent Tutoring Systems (ITS 2002), Biarritz* (Lecture Notes in Computer Science No. 2363, pp. 596–608). Berlin: Springer.

Hume, G., Michael, J. A., Rovick, A. A., & Evens, M. W. (1996). Hinting as a tactic in one-on-one tutoring. *Journal of the Learning Sciences, 5*(1), 23–47.

Kim, J. H. (2000). *Natural language analysis and generation for tutorial dialogue.* Unpublished doctoral dissertation, Illinois Institute of Technology.

Kim, J. H., Freedman, R., & Evens, M. W. (1998a). Relationship between tutorial goals and sentence structure in a corpus of tutoring transcripts. In M. W. Evens (Ed.), *Proceedings of the Ninth Midwest Artificial Intelligence and Cognitive Science Conference (MAICS '98), Dayton, OH* (pp. 124–131). Menlo Park, CA: AAAI Press.

Kim, J. H., Freedman, R., & Evens, M. W. (1998b). Responding to unexpected student utterances in CIRCSIM-Tutor v. 3: Analysis of transcripts. In D. J. Cook (Ed.), *Proceedings of the 11th International Florida Artificial Intelligence Research Society Conference (FLAIRS '98), Sanibel Island, FL* (pp. 153–157). Menlo Park, CA: AAAI Press.

Kim, J. H., Glass, M., Freedman, R., & Evens, M. W. (2000). Learning the use of discourse markers in tutorial dialogue for an intelligent tutoring system. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society, Philadelphia* (pp. 262–267). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Kurtz, K., Miao, C., & Gentner, D. (2001). Learning by analogical bootstrapping. *Journal of the Learning Sciences, 10,* 417–446.

Larsson, S., & Traum, D. R. (2000). Information state and dialogue management in the TRINDI dialogue move engine toolkit. *Natural Language Engineering, 6,* 341–362.

Lepper, M. R., Woolverton, M., Mumme, D. L., & Gurtner, H. (1993). Motivational techniques of expert human tutors: Lessons for the design of computer-based tutors. In S. P. Lajoie & S. J. Derry (Eds.), *Computers as cognitive tools* (pp. 75–105). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Li, J., Seu, J. H., Evens, M. W., Michael, J. A., & Rovick, A. A. (1992). Computer dialogue system (CDS): A system for capturing computer-mediated dialogues. *Behavior Research Methods, Instruments, and Computer (Journal of the Psychonomic Society), 24,* 535–540.

Lulis, E., Evens, M. W., & Michael, J. A. (2004a). How human tutors employ analogy to facilitate understanding. In *Proceedings of the 26th Annual Conference of the Cognitive Science Society, Chicago* (pp. 861–866). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Lulis, E., Evens, M. W., & Michael, J. A. (2004b). Implementing analogies in an electronic tutoring system. In J. Lester, R. M. Vicari, & F. Paraguaçu (Eds.), *Intelligent tutoring systems: Seventh International Conference (ITS 2004), Maceió, Brazil* (Lecture Notes in Computer Science No. 3220, pp. 751–761). Berlin: Springer.

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

Person, N. K., & Graesser, A. C. (2003). Fourteen facts about human tutoring: Food for thought for ITS developers. In V. Aleven, U. Hoppe, J. Kay, R. Mizoguchi, H. Pain, F. Verdego, et al. (Eds.), *AIED 2003 Supplemental Proceedings* (pp. 751–761). Sydney, Australia: University of Sydney School of Information Technologies.

Person, N. K., Graesser, A. C., Kreuz, R. J., Pomeroy, V., & the Tutoring Research Group. (2001). Simulating human tutor dialog moves in AutoTutor. *International Journal of Artificial Intelligence in Education, 12,* 23–39.

Pilkington, R. M. (1999). *Analysing educational discourse: The DISCOUNT scheme* (Tech. Rep. 99/2). Leeds, England: University of Leeds, Computer Based Learning Unit. Retrieved July 2005, from the University of Birmingham, School of Education, http://www.education.bham.ac.uk/aboutus/profiles/curped/pilkington/docs/DISCoun99.htm

Reiter, E., & Dale, R. (1997). Building applied natural-language generation systems. *Journal of Natural Language Engineering, 3,* 57–87.

Reiter, E., & Dale, R. (2000). *Building natural language generation systems.* Cambridge, England: Cambridge University Press.

Reiter, E., & Sripada, S. (2002). Should corpora texts be the gold standards for natural language generation? In *Proceedings of the Second International Conference on Natural Language Generation (INLG 2002), Harriman, NY* (pp. 97–104). New Brunswick, NJ: Association for Computational Linguistics.

Reiter, E., Sripada, S., & Robertson, R. (2003). Acquiring correct knowledge for natural language generation. *Journal of Artificial Intelligence Research, 18,* 491–516.

Rosé, C. P. (2000). A framework for robust semantic interpretation. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL 2000), Seattle* (pp. 311–318). New Brunswick, NJ: Association for Computational Linguistics.

Rosé, C. P., Jordan, P., Ringenberg, M., Siler, S., VanLehn, K., & Weinstein, A. (2001). Interactive conceptual tutoring in Atlas–Andes. In J. D. Moore, C. L. Redfield, & W. L. Johnson (Eds.), *Proceedings of the 10th Conference on Artificial Intelligence in Education (AI-Ed 2001), San Antonio* (pp. 256–266). Amsterdam: IOS Press.

Rovick, A. A., & Brenner, L. (1983). HEARTSIM: A cardiovascular simulation with didactic feedback. *Physiologist, 26,* 236–239.

Rovick, A. A., & Michael, J. A. (1986). CIRCSIM: An IBM-PC computer teaching exercise on blood pressure regulation. In *Proceedings of the 30th Conference of the International Union of Physiological Sciences, Vancouver* (p. 318).

Rovick, A. A., & Michael, J. A. (1992). The prediction table: A tool for assessing students' knowledge. *American Journal of Physiology, 263 (Advances in Physiology Education, 8),* S33–S36.

Sanders, G. (1995). *Generation of explanations and multi-turn discourse structures in tutorial dialogue based on transcript analysis.* Unpublished doctoral dissertation, Illinois Institute of Technology.

Schiffrin, D. (1988). *Discourse markers.* Cambridge, England: Cambridge University Press.

Searle, J. R. (1969). *Speech acts*. Cambridge, England: Cambridge University Press.

Shah, F., Evens, M. W., Michael, J. A., & Rovick A. A. (2002). Classifying student initiatives and tutor responses in human keyboard-to-keyboard tutoring sessions. *Discourse Processes, 33,* 23–52.

Sinclair, J. M., & Coulthard, R. M. (1975). *Towards an analysis of discourse: The English used by teachers and pupils*. Oxford, England: Oxford University Press.

Spitkovsky, J., & Evens, M. W. (1993). Negative acknowledgments in natural language tutoring. In *Proceedings of the Fifth Midwest Artificial Intelligence and Cognitive Science Conference (MAICS '93), Chesterton, IN* (pp. 41–45).

VanLehn, K., Jordan, P., Rosé, C. P., & the Natural Language Tutoring Group (2002). Architecture of Why2-Atlas: A coach for qualitative physics essay writing. In S. Cerri, G. Gouardères, & F. Paraguaçu (Eds.), *Proceedings of the Sixth International Conference on Intelligent Tutoring Systems (ITS 2002), Biarritz* (Lecture Notes in Computer Science No. 2363). Berlin: Springer.

Wells, G. (1999). *Dialogic inquiry.* Cambridge, England: Cambridge University Press.

Winkels, R., & Breuker, J. (1990). Discourse planning in intelligent help systems. In C. Frasson & G. Gauthier (Eds.), *Intelligent tutoring systems: At the crossroads of artificial intelligence and education* (pp. 124–139). Norwood, NJ: Ablex.

Woolf, B. P. (1984). *Context-dependent planning in a machine tutor.* Unpublished doctoral dissertation, University of Massachusetts, Amherst.

Zhou, Y. (2000). *Building a new student model to support adaptive tutoring in a natural language dialogue system.* Unpublished doctoral dissertation, Illinois Institute of Technology.

Zhou, Y., Freedman, R., Glass, M., Michael, J. A., Rovick, A. A., & Evens, M. W. (1999a). Delivering hints in a dialogue-based intelligent tutoring system. In *Proceedings of the 16th National Conference on Artificial Intelligence (AAAI '99), Orlando* (pp. 128–134). Menlo Park, CA: AAAI Press.

Zhou, Y., Freedman, R., Glass, M., Michael, J. A., Rovick, A. A., & Evens, M. W. (1999b). What should the tutor do when the student cannot answer a question? In A. Kumar & I. Russell (Eds.), *Proceedings of the 12th FLAIRS Conference, Orlando* (pp. 187–191). Menlo Park, CA: AAAI Press.