Dialogue Analysis to Inform the Development of a Natural-language Tutoring System for Physics

Sandra Katz
Learning Research and Development Center
University of Pittsburgh
katz@pitt.edu

Patricia Albacete
Learning Research and Development Center
University of Pittsburgh
palbacet@pitt.edu

Pamela Jordan
Learning Research and Development Center
University of Pittsburgh
pjordan@pitt.edu

Diane Litman
Learning Research and Development Center
University of Pittsburgh
litman@cs.pitt.edu

Abstract

Several cognitive scientists attribute the effectiveness of tutorial dialogue to its interactive nature. Although this view has empirical support, the notion of "interactivity" is too vague to guide the development of natural-language tutorial dialogue systems. This paper describes our attempts to operationalize particular forms of *interactivity*: tutor abstraction and specification of student dialogue moves, and tutor prompts for specification. We describe and illustrate the process by which we specified decision rules for abstraction and specification in automated tutorial dialogues about physics. Correlational analyses suggest that particular types of interactive abstraction and specification relations predict student learning, as measured by pretest to posttest gain score—for example, tutor prompts for the student to specify the individual forces that comprise the net force. Since particular kinds of abstraction and specification relations are associated with particular decision rules, these findings are guiding our selection of rules to implement in a tutorial dialogue system for physics.

1 Introduction

Researchers in cognitive science and the development of intelligent tutoring systems have made significant progress in identifying features of human tutorial dialogue that predict learning (e.g., Boyer et al., 2010; Chi et al., 2001; Ward et al., 2009). Several studies of tutorial dialogue converge on an important finding: that it is not so much what tutors do that is important, nor what students do, but how (and how frequently) the student and tutor respond to each others' conversational moves—in other words, the degree to which the tutorial dialogue is *interactive* (e.g., Chi et al., 2001; Chi, 2009; Graesser, Person, & Magliano, 1995; van de Sande & Greeno, 2010). This finding presents a challenge to developers of natural-language tutorial dialogue systems: to operationalize this vague notion of *interactivity* sufficiently enough to simulate it. The goal of this paper is to describe our analyses of a corpus of human-human tutorial dialogues in physics that we have conducted in order to model two forms of interactivity: tutors' *specification* and *abstraction* of students' dialogue moves. Specification involves taking what one's dialogue partner said to a lower level of granularity (e.g., shifting focus from *acceleration* to *average acceleration*), while abstraction is the reverse.

At the lexical level, this type of interactivity is achieved through *cohesive ties*—the same types of relations that contribute to the connectedness of a written text such as synonymy, paraphrase, and word repetition (Halliday and Hasan, 1976). Abstraction and specification are often signaled by hypernym/hyponym ties. However, at other times they are not signaled as such, and might require

inference on the listener's part. For example, in the tutorial dialogue excerpt shown in Table 1, the student needs to infer that the tutor's phrase, "a change in velocity," abstracts over the student's clause, "final velocity is larger than the starting velocity."

Andes Problem: Calculate the speed at which a hailstone, falling from 9000 meters out of a cumulonimbus cloud, would strike the ground, presuming that air friction is negligible. **Reflection Question:** How do we know that we have an acceleration in this problem? **Student:** because the *final velocity is larger than the starting velocity*, 0. **Tutor:** Right, *a change in velocity* implies acceleration.

Table 1: A reflective dialogue about an Andes problem, with related dialogue segments in italics.

The ultimate goal of our project, the Rimac Project,[1] is to develop a natural-language dialogue system for physics that abstracts and specifies from the student's preceding turn when appropriate. Specifically, we are developing automated "reflective dialogues" (e.g., Katz et al., 2003) that scaffold students in co-constructing explanations about the concepts and principles associated with quantitative problems they just solved in the Andes physics tutoring system (VanLehn et al., 2005). Our focus on abstraction and specification is driven by empirical research which shows a correlation between the frequency of these dialogue acts, particularly at the lexical level (i.e., hypernym/hyponym relations) and learning (Ward & Litman, 2008; Ward et al., 2009).

In order to simulate abstraction and specification during tutorial dialogue in a way that promotes learning, we have focused our analyses of human tutorial dialogues on the following questions: (1) When do human tutors abstract and specify what students say, or prompt students to do the same? (2) Does tutor abstraction/specification, taken as a whole, predict student learning, or only particular types of abstraction/specification relations? If the latter, does student ability level mediate the effectiveness of particular types of tutor abstraction and specification moves? Although tutors and students abstract and specialize each others' dialogue contributions, these questions focus on the tutor because we are interested in modeling human tutors' behavior in our reflective dialogue system.

As discussed in Lipshultz et al. (2011), Machine Learning (ML) is one approach that we are taking to model tutors' abstraction and specification of students' dialogue contributions. To summarize this line of work, ML analyses were conducted on a corpus of human-human physics tutorial dialogues that were tagged for interactive hypernym/hyponym relations (Ward et al., 2009). Our goal was to model tutor abstraction and tutor specification in terms of several types of features: student characteristics (e.g., gender, pretest score), features of the Andes problem-solving sessions that preceded tutorial dialogues (e.g., the frequency of various types of system help that the student invoked; the number of correct and incorrect problem-solving entries in the tutor interface), and features of the dialogue context (e.g., the position of tutor abstractions or specifications in reflective dialogues, such as at the beginning or end of dialogues). We found that contextual features produce the most predictive models, and we identified some interesting patterns. For example, tutors tend to abstract early in a reflective dialogue, when students are having difficulty responding to the tutoring system's questions about a just-solved Andes problem. These abstractions seem to be aimed at ensuring that the student understands the basic concepts needed to answer the automated tutor's reflection question. Then, as the dialogue progresses, specification becomes more frequent than abstraction, as tutors probe students for precision—e.g., to specify units and direction for a vector quantity, when the student only states its magnitude.

Although these automated analyses are helping us to specify decision rules that will allow us to simulate tutor abstraction and specification, they are limited in two main ways. First, the ML patterns only capture cases of tutor abstraction and specification that are signaled by lexical,

[1] Rimac is the name of a river whose source is in the Andes. Its name is a Quechua word meaning "talking;" hence the nickname for Rimac, "talking river." We thus considered Rimac to be well-suited to a dialogue system embedded within the Andes tutoring system.

hypernym/hyponym ties. Second, our feature set is restricted to data that is readily available (e.g., gender) or automatically detectable (e.g., number of student help requests during problem solving). Consequently, the models of tutor abstraction and specification produced by these ML analyses cover a restricted set of cases.

To extend and refine these models, we retagged the dialogue corpus to include cases of inter-speaker abstraction and specification relations that are not necessarily signaled by hypernym/hyponym ties (e.g., Table 1) and reflect a broader meaning of abstraction and specification: any instance of raising or lowering the level of granularity of a dialogue partner's moves, respectively. We then specified the discourse context in which each case of tutor abstraction or specification occurred, searched for patterns across cases, and expressed these patterns as general decision rules. To date, this process has led to a set of 24 general rules, examples of which are shown in Table 2. Identification of cases of abstraction, and formulation of rules for abstraction, are in progress. This paper describes and illustrates this manual approach to modeling tutor abstraction and specification—in contrast to the automated approach described in Lipshultz et al. (2011).

1. Example of *Specification for Understanding*
The tutor may prompt the student to define a concept that the student seems to not understand, or one that is needed to understand a particular aspect about another, more central concept.
Number of cases: 13
Local Context (triggering conditions): student answered reflection question incorrectly
Extended Context: occurs early in a set of reflection questions; may be useful for diagnosing student understanding about a topic
Exceptions (rule constraints): 1) tutor defined the concept during the previous reflection question, 2) tutor defined the concept himself, while giving an explanation during the current reflection question, or 3) student answered the reflection question correctly with a "yes/no" response, but gave an incorrect explanation when prompted
2. Example of *Specification for Precision*
When the student provides a numeric value without units, the tutor will specify by providing the missing units, or prompt the student to do so. (The latter is more common.)
Number of cases: 14
Local Context (triggering conditions): student provides a quantity; units are missing
Extended Context: tutor provides units (instead of prompting) when: (1) dialogue has been going on for awhile or (2) student has done well throughout dialogue; missing units are only error
Exceptions (rule constraints): student has trouble understanding concepts addressed in reflection question. (Presumably, tutor does not want to burden the student with details until student grasps these concepts.)
3. Example of *Abstraction*
When the student instantiates a physical principle or law correctly, the tutor may generalize by stating the corresponding principle/law.
Number of cases: to be determined
Local Context (triggering conditions): student applies principle/law, but does not identify it
Extended Context: occurs irrespective of dialogue length, number of mistakes student made while solving problem, dialogue stage (early, middle, or late), time spent on previous reflection questions
Exceptions (rule constraints): (1) tutor discussed corresponding principle/law in a previous dialogue or (2) student instantiated principle/law using generic terms instead of specific values

Table 2: Examples of three types of abstracted decision rules

2 Corpus

Our corpus comes from a previous study on the effectiveness of reflection questions after a physics problem-solving session within the Andes physics tutoring system (Katz, Allbritton & Connelly, 2003). The same corpus was used for the automated (ML) analyses described in Lipshultz et al. (2011). Students taking introductory physics courses at the University of Pittsburgh first took a physics pretest, with 9 quantitative and 27 qualitative physics problems. Following the pretest, students reviewed a workbook chapter developed for the experiment and received training on using Andes. Although there were three conditions in the experiment, we focused our analyses on the Human Feedback (HF) condition, since we are interested in building interactive dialogues. (See Katz, Allbritton, & Connelly, 2003, for additional information.). Students in each condition began by solving a problem in Andes. After completing the problem, students in the HF condition were presented with a conceptually oriented "reflection question," as illustrated in Tables 1 and 4. After the student entered his or her answer, he or she began a typed dialogue with a human tutor. This dialogue continued until the tutor was satisfied that the student understood the correct answer.

Three to eight reflection questions were asked per problem solved in Andes, twelve problems total. After completing these problems and their associated reflective dialogues, students took a posttest that was isomorphic to the pretest and counterbalanced. The main finding of the study was that students who answered reflection questions learned more than students who solved more Andes problems.

There were 16 students in the HF condition (4 male, 12 female). Fifteen students participated in all 60 reflection question dialogues; one student only participated in 53, producing a total of 953 dialogues. There are a total of 2,218 student turns and 2,135 tutor turns in these dialogues. The average number of turns across reflective dialogues is 4.6, ranging from 1.1 turns for simple reflection questions to 11.4 turns for the most complex questions.

The HF condition dialogues were analyzed in Ward et al. (2009) to determine which cohesive ties correlate with learning. As noted previously, hypernym/hyponym ties predicted student pretest to posttest gains. The same corpus was retagged in the current study to identify cases in which tutor abstraction and specification occur independent of lexical, hypernym/hyponym ties (e.g., Table 1), and to determine if these forms of abstraction and specification also predict student learning.

3 Tagging Scheme

Within each of the 953 reflective dialogues, all student and tutor turns were first manually parsed into clauses. We then searched for interactive abstraction and specification relations at the exchange level—that is, between a tutor's dialogue turn and the subsequent student turn, or the reverse. Finally, we tagged the following features of each identified abstraction/specification relation:

- ***Type:*** abstraction or specification
- ***Direction:*** did the tutor abstract/specify the student's previous turn, or the reverse (S→T vs. T→S, respectively)? Alternatively, did the tutor prompt the student for a specification, which the student then provided (T→S)?
- ***Solicited?:*** was the student's or tutor's abstraction/specification in the second turn of the exchange solicited or initiated? (yes or no)
- ***Correct?***: if solicited, was the abstraction or specification in the second turn of the exchange correct? (yes or no. This feature applies to student and tutor replies, because tutors sometimes make mistakes!)
- ***Subtype:*** the particular type of abstraction/specification relation. We used Mann and Thompson's (1988) set of six types of Elaboration relations from Rhetorical Structure Theory (RST) as a framework for classifying inter-speaker abstraction/specification ties, as well as one other RST relation (*term:definition*). These relations are defined and illustrated in Table 3. Note that these relations are bi-directional—for example, *set:member* or *member:set*, depending on the order in which they occur in a dialogue exchange.

Subtype and Definition	Example
set:member—physics concepts and subconcepts	acceleration: instantaneous, average, and constant acceleration
abstract:instance—a general physics concept or principle and a specific instantiation of this concept/principle	The mass of a body times its acceleration equals the (vector) sum of all the forces on that body: m*a = t-(m*g) [m=mass, a=acceleration, t=tension, g=gravity, and m*g = weight]
whole:part—vectors and their components	velocity: horizontal velocity (velocity in the vertical direction is the other component or "part" of this vector)
process:step(s)—a problem-solving goal and the steps required to achieve this goal	find average acceleration: $v_f - v_i/t_0 - t_1 \rightarrow 15\text{-}(\text{-}1)/62=.26$ m/s^2 [v_f = final velocity, v_i = initial velocity, t_1= final time and t_0=initial time]
object:attribute—typically applies to vectors and their attributes; also applies to qualitative aspects of the physical situation	velocity: magnitude, direction, units; motorcycle:speeding up
generalization:specific—a more precise restatement of a vague or general phrase	not accelerating: acceleration = 0
term:definition—a physics concept and its meaning	average acceleration: $a=(v_f\text{-}v_i)/(t_1\text{-}t_0)$

Table 3: Abstraction/specification subtypes

These "subtypes" characterize various ways in which students and tutors jointly construct explanations (Chi, 2009). For example, the tutor might prompt the student to specify the meaning of Newton's Second Law (a *term:definition* relation), or the type of acceleration exhibited in a given physical situation (a *set:member* relation). Conversely, the tutor might tell the student that the student's equation illustrates Newton's Second Law (an *instance:abstract* relation), or specify this law after the student names it (a *term:definition* relation).

One researcher tagged approximately half of the corpus for these subtype relations and another researcher tagged the remaining half. To test for agreement, they independently tagged all of the dialogues for one problem (approximately 8% of the corpus). The kappa for inter-rater reliability was .86, which is considered strong.

Although RST has typically been used to describe the hierarchical rhetorical relations within a single speaker's text (spoken or written), we have found this taxonomy to also be useful for describing inter-speaker abstraction and specification relations within tutorial dialogues.

4 Abstraction and Specification

As noted previously, identification of abstraction relations is in progress. Among all 575 tagged cases of specification, 87% represent the student specifying a more general term or phrase in the tutor's previous turn. Most of these student specifications were solicited; students rarely initiated a specification of a tutor's dialogue move. These observations prompted us to examine these T→S specification relations more closely.

As illustrated by the sample of abstracted decision rules shown in Table 2, there are two types of tutor prompts for specification. These types are distinguishable by function. In one type, which we call *specification for understanding*, the student makes an error, or verbally demonstrates a misconception or poor understanding about a concept. The tutor responds by taking the conversation up a level of abstraction, in order to focus on the concepts that the student lacked. For example, in the dialogue excerpt shown in the left column of Table 4, the student's response to the reflection question indicates that this student does not fully understand the meaning of "net force." The tutor digresses for a moment, by prompting the student to define this concept. This is a *term:definition* relation; that is, the tutor states a term and prompts the student to define it.

The second type of tutor prompt for specification is what we call *specification for*

precision. Throughout the corpus, tutors prompted students to be more precise when students' responses were partially correct but incomplete—most commonly, when a student stated a correct quantity, but omitted units and/or direction.

To date, we have specified 23 decision rules that cover all 575 tagged cases of specification, and an additional rule that fits the cases of abstraction tagged so far. Seventeen rules were classified as *specification for understanding*; six as *specification for precision.* The process of deriving these abstracted rules will be described and illustrated next.

CASE 1	**CASES 2 and 3**
Andes Problem: A model airplane hangs from two strings S1 and S2 which are attached to the ceiling. String S1 is inclined at 45 degrees, and string S2 is inclined at 60 degrees. If the tension in string S1 is 50 N: A) Find the mass of the airplane; B) Find the tension in string S2. **Reflection Question:** Is there a net force in either the x or y direction? (*correct answer is no, because acceleration equals 0)* **S1:** yes, but it adds up to zero **T1:** Let's digress for a moment, then re-evaluate your answer. *Can you say what "net force" means?* **S2:** it is *the sum of all forces acting on an object…* **Description of specification in this particular dialogue:** The tutor asks the student to define "net force" because the student's partially correct response to the reflection question signals confusion.	**Andes Problem:** A motorcyclist races along a flat road with an initial velocity of 1.0 meters per second. At the finish line, 62 seconds later, he reaches a velocity of 15.0 meters per second. Find the magnitude of the average acceleration. **Reflection Question:** Suppose the problem had specified that the motorcyclist had started out with his velocity in the opposite direction (backwards) but the same magnitude (1 m/s). Would we still have had the same answer for the magnitude of the average acceleration? (*correct answer is no)* **S1:** yes, but it adds up to zero **T1:** *what's the definition of velocity?* **S2:** *the change in displacement over time* **T2:** so is velocity a vector or scalar? **S3:** vector **T3:** *what is a vector?* **S4:** *a scalar with direction* **Description of specifications in this particular dialogue:** In the first relation (T1→S2), the tutor prompts the student to define *velocity* because the student answered the reflection question incorrectly. This concept is central, because initial velocity is the changed variable in this "what if" question. In the second relation (T3→S4), the tutor presumably prompts the student to define *vector* in order to draw the student's attention to a particular aspect of velocity, namely its direction, which is crucial for answering the question correctly.

Table 4: Three cases of tutor prompts for definition that led to the first abstract rule shown in Table 2. Text that illustrates this specification relation is shown in italics.

5 Generating Decision Rules for Abstraction and Specification

Decision rules such as those illustrated in Table 2 were derived by a four-step process. First, we described the immediate, local context in which each tagged case of tutor specifications and abstractions occurred. Second, similar cases were grouped together and analyzed with the goal of finding more general ways in which to describe the corresponding abstraction or specification and its context. Third, the extended context of each related case was analyzed, in order to refine the abstracted form of the rules derived from step two. The "extended context" encompasses the dialogue corresponding to the whole reflection question in which each abstraction or specification occurred,

as well as those from previous reflection questions for the same problem. Finally, in order to further refine the abstracted rule, we searched for circumstances in which the tutor chose *not* to abstract or specialize, even when the context was similar to others in which he or she had done so.

To illustrate this process, we will show how we developed the first rule specified in Table 2. Each step of the process is reflected in particular aspects of the rule description shown in this table—that is, the abstracted form of the rule, description of its local and extended context and triggering conditions, and constraints/exceptions.

Step 1: Describing the context of each case of abstraction and specification. We found 13 cases of the tutor eliciting the definition of a concept—that is, T→S *term:definition* relations. Three of these cases are illustrated in Table 4, with the relations of interest shown in italics. For each case, we described the particular context in which this form of specification (*term:definition*) occurred. To the maximum extent possible, this description attempted to operationalize the local triggering conditions. For example, for Case 1 in Table 4, the student's "confusion" is operationalized as answering the reflection question incorrectly—in particular, saying that there is a net force.

Step 2: Abstracting over related cases. Examination of all cases of the tutor prompting the student to define a term revealed that the term is not always the central concept in a reflection question, as it is in cases 1 and 2 shown in Table 4. Alternatively, the tutor may prompt the student to define a concept that is required in order to understand the central concept. For example, in the third case shown in Table 4 (T3→S4), the tutor prompts the student to define *vector*, after the student has given an incorrect definition of the central concept, velocity. After examining all 13 cases of the tutor prompting the student to define a concept, an abstracted rule for this relation was specified as shown in Table 2 (repeated here for convenience): *The tutor may prompt the student to define a concept that the student seems to not understand, or one that is needed to understand a particular aspect about another, more central concept.*

Step 3: Examining the extended context of a rule. The goal of this step is to find other factors that may influence the tutor's decision to use the rule. During this step, we observed that 7 out of the 13 instances of tutor prompts for definitions took place while students were solving an early reflection question—in particular, the second reflection question that the tutoring system presented to them. This suggests that tutors solicited the definition of concepts in order to assess students' knowledge, during this early phase of instruction.

Step 4: Identifying exceptions. Finally, we searched for instances in which the tutor did not use this particular form of specification (*term: definition*), even though the immediate context was similar to others in which it was used. The aim of this step is to identify rule constraints. During this analysis, we found that the tutor chose not to solicit the definition of a concept in three situations, as specified in Rule 1 of Table 2.

6 Correlations Between Abstraction and Specification, and Learning

As noted previously, the frequency of inter-speaker hypernym/hyponym relations predicted student learning in this corpus of tutorial dialogues (Ward et al., 2009). In order to determine if the frequency of abstraction and specification relations which were not necessarily signaled by hypernym/hyponym ties also predict learning, we performed correlational analyses of the frequency of these tagged relations and learning, as measured by pretest to posttest gain scores—specifically, total, quantitative, and qualitative gain scores. These analyses were done for all students combined, and separately for low and high pretest students, classified according to a median split. There were seven high pretest students, nine low pretest students.[2] The data was first normalized by the total number of turns (student turns + tutor turns) per reflection question, in order to control for dialogue length.

Contrary to our expectations, we found no statistically significant correlation between gain

[2] The numbers are uneven because the two pretest scores in the middle of the distribution were identical. Both students who had these scores were assigned to the "low pretest" group.

scores and the total number of specifications or abstractions (cases tagged to date), regardless of direction (S→T or T→S). This led us to consider whether certain types of relations (abstraction/specification subtypes) are stronger predictors of learning than others. Towards this end, we performed a preliminary correlational analysis between the frequency of subtype relations and gain score. Since particular subtypes are associated with particular rules—for example, *term:definition* relations are associated with Rule 1 in Table 2—these analyses also indicate which decision rules are likely to predict learning, and are therefore the most important to implement within our dialogue system.

We found that for all students considered together, *object:attribute* relations in which the tutor prompts the student to specify the units of a value (e.g., 5 m/s) predicts gain score on quantitative test items (R(14)=.584, *p*=.018). This relation is most closely associated with abstracted Rule 2 in Table 2, suggesting that the tutor's attention to units might improve precision, which in turn improves quantitative problem-solving performance. This finding also indicates that the tutor should prompt students for missing units, instead of providing them himself.

Another significant finding for all students considered together is that *whole:part* (and *part:whole*) relationships predict quantitative gain (R(14)=.633, *p*=.008). *Whole:part* relationships mainly occurred when the tutor asked students to specify the individual forces that make up the net force. Perhaps this prompt increased students' understanding of Newton's Second Law, and was reflected in higher quantitative gain scores.

For the subgroup of low pretest students, there was a statistically significant correlation between *member:set* relations and qualitative gains (R(7)=.706, *p*=.034). This relation was mainly found in instances of the following abstracted rule: *When the student writes or talks about an equation, the tutor may ask the student to specify the meaning of a variable in that equation that the student shows evidence of not understanding, with respect to the situation at hand.* This suggests that making students ponder about the meaning of variables in equations—for example, what "F" is in F=m*a—enhances students' understanding of the concepts associated with these variables. Furthermore, it might help students comprehend the relationships between concepts that are expressed in mathematical formulae.

For the subgroup of high pretest students, there was a statistically significant correlation between *process:step* relations and qualitative gain (R(5)=.863, *p*=.012). This relation was found, for example, in instances of the abstracted rule: *When two quantities Q1 and Q2 are related, and the student has difficulty with Q1 or with the relationship between them, the tutor might ask the student to specify Q2, which may be simpler, or the tutor might ask the student to specify the relationship itself.* This indicates that to aid students in understanding a new concept, or one which they are having difficulty with, it might be useful to have them reflect on a known concept that is related to the one being taught. For example, Q1 is acceleration and Q2 is net force, and these concepts are related through Newton's Second Law (F=m*a). If a student is having trouble comprehending the concept of net force but understands the concept of acceleration, prompting the student to specify acceleration (with respect to the current problem) and to explain Newton's Second Law may help the student understand the concept of net force.

To our surprise, we also found several statistically significant negative correlations between certain types of relations and learning. One of them was the frequency of *abstract:instance* (or *instance:abstract*) relations, with respect to overall gain score, when all students were considered (R(14)= -.610, *p*=.012) and when only low pretest students were considered (R(7)= -.671, *p*=.048). When only low pretest students were considered, there was also a negative correlation between the frequency of *abstract:instance* (and *instance:abstract*) relations and qualitative gain scores (R(7)= -.716, *p*=.030). Similarly, the frequency of *generalization:specific* (and *specific:generalization*) relations was negatively correlated with overall gains (R(7)= -.757, *p*=.018) and qualitative gains (R(7)= -.667, *p*=.050) among low pretest students.

One possible interpretation of these negative correlations is that the more these relations are used, the less students learn. Another possible interpretation is that the degree to which these types of relations take place in the dialogues is an indicator of the level of difficulty that students have with understanding the concepts associated

with just-solved problems. For example, over the course of several dialogues, the tutor may have repeatedly asked the student to give the numerical value of the concepts involved in solving these problems (*abstract:instance* relations) or may have given more precise statements of vague utterances made by the student (*generalization:specific* relations) because the student had persistent difficulty with solving the problems and/or answering the reflection questions, and these difficulties were not resolved by posttest time. This hypothesis warrants further investigation.

7 Conclusion

Observations of skilled teachers and tutors indicate that tutoring systems should explain *with* students, not *to* them (e.g., van de Sande & Greeno, 2010). The work described in this paper takes a step towards operationalizing how such co-constructed explanations evolve during human tutoring. To a large extent, human tutoring is patterned, and can be specified as decision rules such as those illustrated in Table 2. Our initial correlational analyses suggest that some of these rules might be more important to simulate than others.

In our current work, we are developing reflective dialogues for Andes that implement these rules. Since these rules are closely coupled with particular types of abstraction/specification relations, evaluations of our dialogue system will allow us to test hypotheses about specific types of interactive tutoring events that support learning.

Acknowledgments

The authors thank Christine Wilson for assistance with data analysis, and the anonymous reviewers for their comments. The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305A100163 to the University of Pittsburgh. The opinions expressed are those of the authors and do not represent the views of the Institute or the U.S. Department of Education.

References

Kristy E. Boyer, Robert Phillips, Amy Ingram, Eun Y. Ha, Michael Wallis, Mladen Vouk, and James Lester. 2010. Characterizing the Effectiveness of Tutorial Dialogue with Hidden Markov Models. In Proceedings of ITS 2010: 55-64.

Michelene T.H. Chi, Stephanie A. Siler, Heisawn Jeong, Takesi Yamauchi, and Robert Hausmann. 2001. Learning from Human Tutoring. Cognitive Science, 25: 471-533.

Michelene T.H. Chi. 2009. Active-Constructive-Interactive: A Conceptual Framework for Differentiating Learning Activities. Topics in Cognitive Science 1: 73-105.

Arthur C. Graesser, Natalie K. Person, and Joseph P. Magliano. 1995. Collaborative Dialogue Patterns in Naturalistic One-on-One Tutoring. Applied Cognitive Psychology, 9:359-387.

Michael A.K. Halliday and Ruqaiya Hasan. 1976. Cohesion in English. Longman, London.

Sandra Katz, David Allbritton, and John Connelly. 2003. Going Beyond the Problem Given: How Human Tutors use Post-Solution Discussions to Support Transfer. International Journal of Artificial Intelligence in Education, 13(1): 79-116.

Michael Lipschultz, Diane Litman, Pamela Jordan, and Sandra Katz. 2011. Predicting Changes in Level of Abstraction in Tutor Responses to Students. In Proceedings of the 24th International Florida Artificial Intelligence Research Society Conference (FLAIRS-2011).

William C. Mann and Sandra A. Thompson. (1988). Rhetorical Structure Theory: Toward a Functional Theory of Text Organization. Text, 8:243-281.

Carla van de Sande and James G. Greeno. 2010. A Framing of Instructional Explanations: Let Us Explain *With* You. Instructional Explanations in the Disciplines, 2010(2): 69-82.

Kurt VanLehn, Collin Lynch, Kay Schultz, Joel A. Shapiro, Robert Shelby, Donald Treacy, Anders Weinstein, and Mary Wintersgill. 2005. The Andes Physics Tutoring System: Lessons Learned. International Journal of Artificial Intelligence in Education, 15(3): 1-47.

Arthur Ward and Diane Litman. 2008. Semantic Cohesion and Learning. In Proceedings of the 9th International Conference on Intelligent Tutoring Systems (ITS): 459-460.

Arthur Ward, John Connelly, Sandra Katz, Diane Litman, and Christine Wilson. 2009. Cohesion, Semantics, and Learning in Reflective Dialogue. In Proceedings of the Workshop on Natural Language Processing in Support of Learning: Metrics, Feedback, and Connectivity. Held with the 14th International Conference on Artificial Intelligence in Education (AIED) 2009.