Artificial Intelligence in Education
J.D. Moore et al. eds., IOS Press, 2001, pp. 210–221

Processing Language Input in the CIRCSIM-Tutor Intelligent Tutoring System

Michael Glass
EECS Dept. M/C 154
University of Illinois at Chicago, Chicago, IL 60612

Abstract. Language-based Socratic tutoring systems face the problem of handling unconstrained student input. By asking closed, short-answer questions, the CIRCSIM-Tutor dialogue-based intelligent tutoring system has rendered the free-text input understanding task more tractable. Even though the tutor's questions can be answered very simply, CIRCSIM-Tutor has preserved quite a number of the pedagogical advantages and opportunities of unconstrained student input. By using only simple language processing facilities the tutor is able to respond appropriately to a high percentage of student turns.

1 Introduction

CIRCSIM-Tutor (CST) is an intelligent tutoring system for teaching the baroreceptor reflex mechanism of blood pressure control to first-year medical students. What distinguishes CST from many other ITSs is its high reliance on natural language. Aside from a simple table that the student fills in, tutoring is accomplished exclusively by natural language dialogue. There are no diagrams, multiple-choice lists, or animated displays. The dialogue is under the tutor's control; the machine asks questions and the student answers with free text in imitation of the Socratic style of human tutoring.

Imitation of human Socratic tutoring was adopted by some of the earliest ITSs, for example SCHOLAR (Carbonell 1970), but they had limited capabilities to categorize student utterances beyond simply right/wrong, and limited ability to plan multi-turn dialogue acts. The resulting dialogue was not very human-like. Since then the field of intelligent tutoring systems has more often than not eschewed language-based dialogue-oriented Socratic tutoring.

Recently there has been considerable activity in dialogue-based tutoring, see for example the workshop papers collected by (Rosé and Freedman 2000). There is a better understanding of the structure and components of tutoring dialogues, as obtained by studying human dialogues in depth (Fox 1993, Graesser et al. 1995, VanLehn et al. 1998), and there have been advances in the machine planning for dialogue generation, for example (Freedman 2000). In order to carry out a conversation, a third big area that needs work is processing of student input.

Partly because of improvements in CIRCSIM-Tutor's ability to respond to more categories of unexpected student input (Zhou et al. 1999), and partly to be more robust in handling the vagaries of free-text input, the input understanding component of CIRCSIM-Tutor needed to be upgraded. The human tutorial dialogues that inform CIRCSIM-Tutor contain many instances of very short answers (Glass 1997), and the computer tutor is dependent on correctly categorizing such answers. This consideration drives the design of the input understander in the direction of robust extraction of simple concepts. We have discovered that we can get fairly good results in limited understanding of student utterances from very simple language processing techniques.

2 CIRCSIM-Tutor's Tutoring Dialogue

CIRCSIM-Tutor presents the student with a description of a perturbation that disturbs blood pressure in the human organism. For example, a patient might hemorrhage blood. The student is required to predict the direct effect on seven physiological variables, then predict how these variables will change as a result of action of the baroreceptor reflex. Variables include, for example, heart rate and central venous pressure. Predictions are qualitative: a variable increases, decreases, or does not change. After each set of seven predictions the tutor engages the student in remedial dialogue with the goal of having the student correct the incorrect predictions.

In the dialogues, the tutor's questions usually admit of short one- or two-word answers. The range of expected answers is limited, for example one or two of the seven variables in the prediction table. The students usually do not produce longer utterances than needed. The following is a sample of dialogue with the computer tutor, edited slightly for readability, showing a typical variety of short student answers:

T: What are the determinants of Stroke Volume?
S: inotropic state
T: Right. The other determinant represents preload. What is the other determinant of Stroke Volume?
S: cvp
T: Right, Stroke Volume is determined by Central Venous Pressure and Inotropic State. Which determinant is dominant in this case?
S: cvp
T: Correct, the important determinant of Stroke Volume in this case is Central Venous Pressure. Is the relationship from Central Venous Pressure to Stroke Volume direct or is it inverse?
S: inverse
T: Sorry, [gives physiological explanation]. What is the correct value of Stroke Volume?
S: down

In this dialogue the tutor's question are closed, the expected answers are quite short, and indeed the student input is usually short to match. Given that the range of acceptable student answers is small, it would be easier to have the student pick from a list of answers, a strategy that would bypass a somewhat messy and error-prone free-text input process.

Thus we need to explain why 1) we chose free-text input, and 2) we think it is useful to do so even when the expected student input is quite short.

We chose unconstrained free-text input for three principal reasons:

1. We want to rely on the student's recall memory instead of recognition memory if possible. Constraining the student's utterance, for example by presenting the student with alternative answers to pick from, gives away the answer to the student.
2. We have observed that experienced human tutors are more likely than inexperienced tutors to force the students to enunciate an answer, as opposed to the tutors enunciating it themselves (Glass 1999).
3. Human tutoring transcripts show that a number of unexpected simple answers stimulate useful tutoring. Free-text input gives the student opportunity to produce such unexpected answers.

In addition to our three particular motivations, there are general experimental and theoretical indications for preferring language-based dialogue as a tutoring modality. The self-explanation effect observed by (Chi et al. 1994) depends on the student verbalizing explanations in order to learn. Halliday and Vygotsky, approaching the issue from the different disciplines of linguistics and psychology, have developed respected language-based theories of learning (Wells 1999). Thus even though CIRCSIM-Tutor's conversation does not overtly prompt the student for complex language containing explanations, we believe that forcing the student to use language is on the right track.

3 The Categorization Task

The data for the analysis of input understanding comes primarily from fifty one- and two-hour tutoring sessions, where the tutor was one of two professors of physiology and the students were in the physiology class for first year medical students. From these transcripts we have obtained language phenomena of both the students and the tutors, as well tutoring methods and tactics. The tutoring sessions were conducted keyboard-to-keyboard with the participants in separate rooms. The software enforced turn-taking, the tutor and student could not interrupt each other or type simultaneously. Thus all the communication between tutor and student appears more or less cleanly in the log file, and the tutoring happened under the same communication conditions as with the computer tutor.

That short-answer dialogue is useful is supported by additional empirical evidence from counting the lengths of several thousand student dialogue turns in the human tutoring transcripts. Although the mean student turn is 7 words, the median is only 2 words, indicating that most student utterances are quite short.

Freedman (1997) makes the case that by virtue of free-text input CIRCSIM-Tutor can imitate the tutorial dialogue patterns of human tutors, even though the machine asks only short-answer questions while the human tutors often ask more open questions. Fundamentally the process is to categorize the student's utterance and use the categorized utterance to drive the preconditions in a dialogue planner. The more fine-grained the tutorial strategies in the planner, the more discrimination is needed in categorizing the student's utterances. For sophisticated tutoring just categorizing the student's answer as "right" and "everything else" is not enough.

Even though the CIRCSIM-Tutor input understanding process merely needs to categorize the student's utterance to a closed question, the fact that CIRCSIM-Tutor processes unconstrained input presents a number of challenges and opportunities to the categorization task.

There is considerable unexpected semantic variation that CIRCSIM-Tutor needs to understand and categorize.

A common type of unexpected input is a "near miss," where the student's response is not what the tutor expected, but it is nevertheless close enough that the tutor can introduce extra dialogue steps to bring the student from the near-miss answer to the desired answer. In our dialogues, a near-miss answer often takes the form of the name of a variable that is causally related to one of the desired answers. The near-miss variable might be one that the student learned in class but is not part of the computer tutor's normal tutorial discussion. Here is an example of a near miss from the human tutoring transcripts:

T: How does the reflex manage to lower TPR?

S: Dilation of blood vessels. [Physiologically related to desired answer]

T: And how does it accomplish that?

Common misconceptions are also evident in free-text student answers to CIRCSIM-Tutor's questions. In human dialogues, the tutor often diagnoses these misconceptions based on extremely little data, only a word or two. The tutor often seems to not engage in deep understanding of a student misconception, but rather rely on "stereotype" modeling of the student (Kay 2000). Here is an example:

T: First, what parameter determines the value of rap?

S: Venous return and peripheral resistance influences return.

T: Not in the way you seem to think.

Even when the student is simply wrong, there are hinting strategies that can be invoked in different contexts. Thus answers that are simply wrong can sometimes be subcategorized according to the type of hint the human tutors customarily use. To pick an easy example, in response to an answer that appears to be outside the domain of the immediate question, the tutor can inform the student what kind of answer is expected.

CIRCSIM-Tutor has an extensive catalog of common misconceptions as evinced by short answers, known wrong answers with corresponding hinting strategies, and other unexpected inputs that present tutorial opportunities. In general these were derived from close study of transcripts of human tutoring of the same physiological problems. It also has physiological knowledge to enable it to recognize a variety of physiologically near-miss answers. A complete classification of answers recognized by CIRCSIM-Tutor (Zhou et al. 1999) is as follows:

1. Correct
2. Partially correct answer, i.e. some part is correct and the rest is not
3. Near miss answer, pedagogically useful but not the desired answer
4. "I don't know" answer
5. "Grain of truth" answer, incorrect but indicative of a partially correct understanding of the problem
6. Misconception, a common confusion or piece of false knowledge
7. Other incorrect answers
8. Mixed answers, a combination of answers from the other categories

It will be noted that many of these varieties of unexpected student input are dependent on the student using recall memory and free-text input. They also provide opportunities for teaching, so it is important that CIRCSIM-Tutor handle them.

An additional motivation for improving the computer tutor to recognize a wider variety of student answers is to reduce the number of times CIRCSIM-Tutor denies a true-but-irrelevant answer. When the student's answer is factually correct, but not what the tutor was looking for, the tutor should not respond with "wrong." Here is an example where the machine successfully recognized a true-but-irrelevant answer:

T: What is the determinant of Central Venous Pressure?
S: blood volume
T: Blood Volume determines CVP. However, no change in blood volume occurs in this situation. [A worse answer: "wrong."]
T: Think about another determinant.

Recognizing reducing the number of these blunders was an important prerequisite for using CIRCSIM-Tutor with live classes of medical students.

4 Syntactic and Linguistic Considerations

One implication of relying on recall memory is that even when the student responds with the correct expected concept, the student might express that concept in an unusual way. For example, one concept that occurs frequently in the dialogues is the notion of control by the nervous system. The students sometimes say "autonomics" or "neural system." Sometimes they utter a bare adverb "neurally" or a simple sentence "it is neurally controlled." Sometimes they refer to one part of the nervous system, as in "sympathetics." The students can be linguistically productive, even in a very small domain.

To some extent this productivity is driven by substitution of sense for reference. Instead of naming a concept (reference), a student describes it (sense). By recognizing these locutions and responding with the correct term in its response, the tutor instructs and reinforces the proper use of medical language.

It must be noted that there are often extra words and syntax in the student input that do not contribute to the machine's understanding of the student's answer, complicating the parsing and understanding task. An interesting example is when the student hedges an answer, indicating uncertainty. We see this frequently in transcripts of human tutoring, but it also occurs occasionally in the computer tutor sessions, as when one student typed "increase, evidently." This kind of answer can profoundly change the syntax of the utterance. When "it increases" is expressed as "I think it increases" the main verb of the sentence changes from "increase" (the answer) to "think" (irrelevant).

Here is an attested example of several phenomena occurring in one student dialogue turn. This is from a transcript of a student using the computer tutor:

T: What is the determinant of Central Venous Pressure?
S: Central Venous Pressure is dependent on the volume of blood in the venous compartment. [Entire expected answer: "Cardiac Output."]

In this example we see:

- The student used a complete sentence containing more than just the single parameter that was requested.
- The student's answer is the physiological parameter normally denoted "CBV," here being described by the noun phrase "volume of blood in the venous compartment."
- CBV is causally related to the desired answer and physiologically correct, so this answer should not be denied.
- CBV is a near miss that presents a teaching opportunity.

Unfortunately in the example just cited the computer tutor did not recognize that the student was describing CBV, so the tutor classified it as a wrong answer.

5 Lower Level Understander Processing

In this paper we describe the architecture of and our experience with the lower levels of processing in the CIRCSIM-Tutor input understander. The final categorization of the student's utterance, which is carried out by the domain reasoning and student modeling components of the system, is not described here.

The fundamental idea in the input understander is to be as permissive as possible. It extracts whatever is needed from the student's input and ignores the rest. Only a little parsing is done; mostly the understander searches for words and phrases that could be answers to the question that CIRCSIM-Tutor just asked. It is possible to fool the understander, viz.:

T: Which determinant is dominant in this case?
S: Anything except HR.
T: HR is correct.

In practice, we are not worried about such behavior. If the input understander manages to correctly interpret the utterances of a person who is not trying to fool the system, it is performing its job. In the more than 3600 student turns the understander has processed during normal class use with medical students, the software has only once responded inappropriately because of this kind of misunderstanding.

The current version of the input understander sharply differs from earlier versions (Lee 1990; Seu 1992; Seu and Evens 1991) that used a Lexical Functional Grammar to parse the input. This produced an f-structure, representing the utterance in a kind of functional structure among the parts of the sentence (e.g. the verb and its complements). From this f-structure the input understander extracted features that matched the tutor's question. There were two reasons we became dissatisfied with this approach:

1. Extragrammaticality was sometimes not handled very well. Enhancing the grammar to handle new constructions was a nusiance.
2. With the inclusion of a variety of unexpected student answers the range of possible f-structures expanded. Building the f-structure and extracting the feature we wanted became a more involved, frequently pointless exercise.

To see why building the f-structure was often a pointless exercise, consider that perhaps the student typed only "increase." The parser, in an effort to find grammaticality, builds an f-structure for an imputed sentence that matches the dialogue context, for example "MAP increased in DR.” But the information the system needs is the original concept the student typed, “increase.”

Processing in the CIRCSIM-Tutor input understander consists of the following steps in sequence: lexicon lookup, spelling correction, processing by finite state transducers, lookup in concept ontologies, and finally matching to the question. We will discuss these in turn.

Lexicon lookup largely retrieves a meaning token and a part of speech. The lexicon contains many of the standard abbreviations used by the students, as well as a number of common phrasal entries. There are about 5000 words in the lexicon, counting all morphological and phrasal variants, but only 5% have meaning tokens. The remainder are needed for their parts of speech and for spelling correction purposes. The lexicon was initially created by extracting from the human tutoring transcripts all the correctly spelled words.

Spelling correction is invoked when lexicon lookup fails. There are many spelling errors and impromptu abbreviations in the student utterances; we cannot ignore them. For example, among the 1860 student turns in one recent batch of tutoring sessions, we observed the word “increase” to have been variously abbreviated or misspelled as “i,” “in,” “io,” “inc,” “incr,” “incrase,” “increasw,” and “ncrease.” For this purpose, we have been using the spelling corrector written by Elmi (Elmi 1994; Elmi and Evens 1998). To a large extent, spelling correction is aided by the fact that the recognition step that follows uses only little syntax. If “incrase” is spell-corrected to any form of “increase,” regardless of part of speech, it will probably be recognized.

In the new input understander the recognition mechanism is a cascade of finite state transducers (Roche and Schabes 1997). This approach has been popular in information extraction applications because it is robust and can be designed to extract from the input sentence only the information which is being sought. The robust parsers that are now available, for example (Rosé 2000), are attractive for processing unconstrained input. However there are aspects of student input that we would rather not parse, even though the technology might be available to parse them. Accordingly we coded a number of small machines for specific kinds of answers. One machine can recognize a variety of utterances about a neural mechanism, another recognizes utterances containing a parameter and a qualitative change, and so on.

The finite state transducers are often simple enough that they are just performing keyword lookup. This is appropriate, considering that often one- and two-word answers suffice. Nevertheless keyword lookup is not always sufficient, so the transducers do engage in a small amount of syntax recognition. For example there is a state machine to distinguish the copula “is” from its homograph, an abbreviation for the physiological parameter Inotropic State. Another transducer recognizes some simple prepositional phrases such as “volume of blood in the atrium”; the words “volume” and “blood” do not capture the concepts by themselves. Negation is recognized by some transducers, e.g., “not really changed” is recognized as meaning “unchanged.”

During syntactic processing the meaning tokens of the sentence are looked up in concept ontologies. One issue that arises while matching the student’s answer to the tutor’s question is that the meaning of an answer depends on the question that was asked. A certain physiological concept might be indicative of a misconception when it is an answer to one question, but similar physiological concepts might be category errors. The machine tries to respond helpfully to category errors, for example if it was expecting an answer such as “directly proportional,” and the student types the name of some unrelated physiological concept (such as a parameter name), CIRCSIM-Tutor will let the student know the kind of answer that is expected. By providing a first level of discrimination of student inputs, tuned for each question,

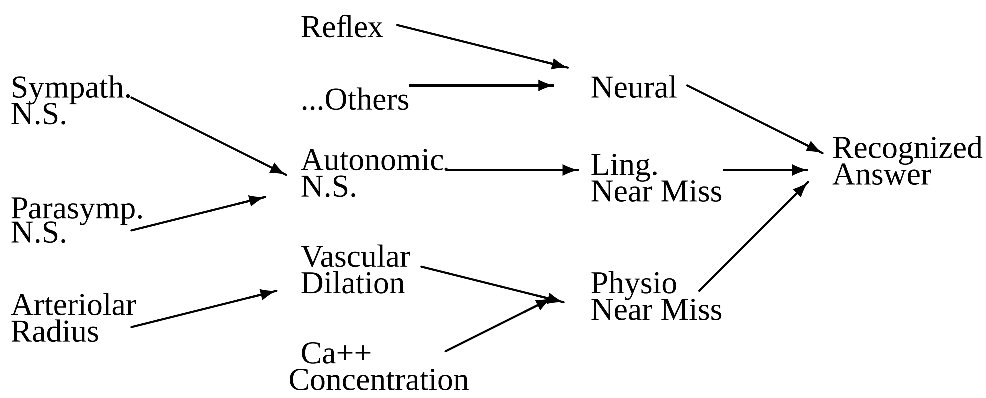

Figure 1: Example Part of Ontological Map for Neural Answers

the concept ontologies enable this problem to be caught before it reaches the higher-level tutorial functions where the misconceptions are recognized.

Another example where the ontology enhances matching the answer to the question arises because we have evidence that human tutors make small distinctions among the several different possible linguistically similar answers. If the student's utterance is sufficiently indirect, the tutor accepts it, but echoes back an answer using more desirable language. Freedman (1997) calls this a "linguistic near miss." Here is an example from a human tutoring session. The student's answer and the tutor's corrected answer in more desirable language are both emphasized:

T: How is TPR controlled?

S: *Sympathetic vasoconstriction.*

T: Right, TPR is primarily under *neural* control. We're talking about what happens before there are any neural changes.

When the student responds with a more direct answer, e.g. "nervous system," the tutor does not echo back more desirable language. Although CIRCSIM-Tutor's ontological maps make some distinctions identifying these linguistic near misses, the tutor does not in general respond to them at this time.

An extract from the ontological map for neural answers is shown in Figure 1. This shows that various more specific notions related to the nervous system are linguistic near misses, which are acceptable answers to the neural question. Also various physiological parameters are physiological near misses. Other physiological parameters, if given in answer to this question, are category errors, and would produce an informative message from the tutor similar to "please type a mechanism of control, such as neural or physical."

The final step in the input understander is to match the result of the previous processing (spelling corrector, transducers, ontological maps) to the question, producing a representation of the answer. This step is accomplished by *ad hoc* code for each type of question. This process could be less hard-coded in the future. Much of the processing is devoted to error checking. As an example, one kind of error it checks for is an answer with logical inconsistencies, such as "it goes up and down," which is not a possible answer to any of our questions. On the other hand, "it goes up and increases" would not be flagged as erroneous.

Table 1: Representative Samples of Input Understander Failure.

Student Input	Reason Not Recognized
“increaseed”	Spelling correction algorithm did not repair it, possible bug.
“istpr”	Parameters “is” and “tpr” are joined, but joins are not currently recognized by spelling correction.
“central venous volume”	Concept was missing from the lexicon.
“kiss my ass”	Expressions of frustration are not one of the tutorial planner’s answer categories, so are not recognized.
“in”	This was a too-drastic abbreviation for “inverse.”
“help”	The word “help” in the lexicon should have been marked with the same meaning as “I don’t know.”
“metabolic factors”	This domain concept is beyond CST’s knowledge, so the words have no assigned meaning in the lexicon.

6 Experience with the Input Understander

In 1998 and 1999 CIRCSIM-Tutor, incorporating its current understander, was used with two physiology classes of first-year medical students at Rush Medical College. There were 100 students in total, some of whom used the tutor in pairs while the rest used it individually, yielding 78 sessions. We recorded a total of 3664 student dialogue turns, of which 23 were garbled beyond human recognition. Of the remaining 3541 turns, only 36 were not handled properly by the input understander. Overall, the input understander failed to respond appropriately only 1% of the time. The garbled unusable turns include any turns for which we could not divine an intention of the student; all were no more than few characters long.

Of the usable turns, many were simply the plus, minus, and zero symbols (or misspellings thereof) used to indicate values of increased, decreased, and no change. These were correctly understood. The remaining 2644 turns were almost entirely alphabetic characters, and thus were subject to the full course of syntax recognition, spelling correction, and so on. The 35 failures represent 1.3% of these. Table 1 shows examples of unrecognized input, these errors and similar covered all the failures. Many of the errors are fixable, e.g. by adding vocabulary, but some would require substantial work.

7 Discussion

The approach we have taken for input understanding in CIRCSIM-Tutor is different than in some of the other recent ITS projects. The contrast with other projects shows some of the limitations and advantages of our approach.

The AutoTutor computer literacy tutor (Graesser et al. 1998) teaches the vocabulary and concepts of a basic computer literacy course. Tutorial planning is driven by a large base of curriculum scripts enhanced by a variety of other mechanisms; they guide the form and content of the tutor's responses through typically four exchanges while teaching one topic. Partly due to the subject content and partly informed by the study of human tutors, AutoTutor's questions can be akin to short essay questions on an exam. It asks open questions that admit of paragraph-long answers. This is appropriate for a topic where you want to know how well the student gets the general idea. The typically sixteen-word student answers can present a formidable challenge in understanding. AutoTutor brings a variety of technologies to bear on the problem, most notably applying Latent Semantic Analysis for approximate understanding of the longer student answers (Wiemer-Hastings et al. 1999). Although language is not the only mode in which AutoTutor communicates—it sports colorful diagrams and an animated face—it does seem to embody in a big way the basic principles that drove CIRCSIM-Tutor: at the core of the teaching process it is import to elicit language from the student.

It is not obvious whether the CST input understander technique would be suitable to the AutoTutor task. An argument that it might do well is the (Wiemer-Hastings et al. 1999) observation that their LSA approach performs only somewhat better than the best keyword-based approach can do. In favor of the LSA approach is that it is rather more robust and less brittle. An argument in favor of the CIRCSIM-Tutor approach is that its input understanding incorporates light parsing as well keyword lookup, while LSA is famously ignorant of syntax. Thus it is conceivable that a suitably scaled-up implementation of the CST input understander might perform creditably in AutoTutor. LSA's insensitivity to syntax has motivated the AutoTutor authors to experiment with ways to enhance it with syntactic information.

The biggest difficulty that I see in such a project is that the CST input understander was hand-coded to operate in a limited domain. It will not easily scale up. Large-scale systems based on finite state transducers are the state of the art in information extraction. They are built and trained mostly automatically, but training requires vast quantities of the same types of text that are to be processed. There is no corpus of a million words of student answers to the computer tutor's questions. The AutoTutor LSA semantic space was trained on professionally-written textbooks and articles. It learns the semantics latent in vocabulary co-occurrence, and the fact that LSA works at all shows the students seem to use many of the same words as the experts. AutoTutor might work so well because it did *not* learn the syntax of written professorial English, as this is not what students are likely to use.

Another recent relevant project in ITS input understanding is the addition of an input-understanding component to the PACT Geometry Tutor (Popescu and Koedinger 2000). From this tutor students learn proofs in secondary school geometry. The tutor is already well-established, but the authors have been motivated to enhance it by having the students write down explanations for the steps in their proofs. The motivations are much the same as in CIRCSIM-Tutor and AutoTutor: 1) a belief that forcing the students to think about and enunciate the material enhances learning, 2) a desire to have the tutor diagnose the student's answers and respond in a pedagogically useful way. The Geometry Tutor however places a different demand on its input understanding mechanism—it needs to represent in detail the student's explanations of logical and geometrical reasoning so the tutor can apply its own pedagogical and geometrical reasoning.

Thus in the Geometry Tutor it is not enough to know that the student has used some of the right words and classify the utterance according to a broad category. The input understander tries to parse as much of the student's utterance as it can and represent the result in a description logic. This demand is in many respects the opposite of what works so well in CIRCSIM-Tutor.

8 Conclusion

CIRCSIM-Tutor supports free-text natural language input in a Socratic intelligent tutoring system, but the examples of AutoTutor and the Geometry Tutor show that there is quite a range of input understanding jobs that might be entailed by such a tutor. In the case of CIRCSIM-Tutor the input understanding task is rendered more tractable by asking questions where short responses from the students are sufficient. For this task we have implemented a fairly simple input understanding mechanism that works quite well with classes of medical students.

We believe the very act of eliciting even simple answers from the students is good tutoring practice, a big step beyond not using any free-text input at all. From short free-text responses there are in our domain quite a few interesting phenomena in the realm of unexpected student utterances. It is important to recognize these phenomena in the text understanding process, partly so the machine tutor doesn't deny perfectly true student answers. The bigger reason is that even with these limited questions, free-text input provides the tutor with many teaching opportunities. Having a well-functioning input understander was necessary for a production-quality ITS.

9 Acknowledgments

The members of the CIRCSIM-Tutor group at the Illinois Institute of Technology all worked to improve the tutor to the quality where it could be used in the classroom on a non-experimental basis. Particular mention goes to Reva Freedman, who showed us how to think about all manner of issues, Yujian Zhou, who was responsible for improving its response to unexpected student inputs, and Martha Evens who has run and inspired the group since its inception.

This work was supported by the Cognitive Science Program, Office of Naval Research under Grant No. N00014-94-1-0338 to Illinois Institute of Technology. The content does not reflect the position or policy of the government and no official endorsement should be inferred.

References

Carbonell, Jaime R., 1970. AI in CAI: An Artificial Intelligence Approach to Computer Assisted Instruction, *IEEE Transactions on Man-Machine Systems*, vol. 11 no. 4, pp. 190–202.

Chi, Micheline T. H., Nicholas de Leeuw, Mei-Hung Chiu, and Christian LaVancher, 1994. "Eliciting Self- Explanations Improves Understanding," *Cognitive Science*, vol. 18, pp. 439–477.

Elmi, Mohammad Ali, 1994. *A Natural Language Parser with Interleaved Spelling Correction Supporting Lexical Functional Grammar and Ill-Formed Input*, Ph.D. diss., Department of Computer Science, Illinois Institute of Technology.

Elmi, Mohammad Ali and Martha W. Evens, 1998. Spelling Correction using Context. In Proceedings of the 17th International Conference on Computational Linguistics (COLING '98), Montréal, published together with 36th Annual Meeting of the Association for Computational Linguistics (ACL '98), pp. 360–364.

Fox, Barbara A, 1993. *The Human Tutorial Dialogue Project*. Hillsdale, NJ: Lawrence Erlbaum.

Freedman, Reva, 1997. Degrees of Mixed-Initiative Interaction in an Intelligent Tutoring System, AAAI 1997 Spring Symposium on Computational Models for Mixed-Initiative Interaction. Menlo Park, CA: AAAI Press.

Freedman, Reva, 2000. Plan-Based Dialogue Management in a Physics Tutor. In Proceedings of the Sixth Conference on Applied Natural Language Processing (ANLP 2000), Seattle.

Glass, Michael, 1997. Some Phenomena Handled by the CIRCSIM-Tutor Version 3 Input Understander. In Proceedings of the Tenth Florida Artificial Intelligence Research Symposium (FLAIRS '97), Daytona Beach, FL. Menlo Park, CA: AAAI Press.

Glass, Michael, 1999. *Broadening Input Understanding in an Intelligent Tutoring System*, Ph.D. diss., Dept. of Computer Science, Illinois Institute of Technology.

Graesser, Art, Stan Franklin, Peter Wiemer-Hastings, and the Tutoring Research Group, 1998. Simulating Smooth Tutorial Dialogue with Pedagogical Value. In Proceedings of the 11th International Florida Artificial Intelligence Research Symposium (FLAIRS '98), Sanibel Island, FL. Menlo Park, CA: AAAI Press

Graesser, A. C., N. K. Person, and J. P. Magliano, 1995. Collaborative Dialogue Patterns in Naturalistic One-to-one Tutoring. *Applied Cognitive Psychology*, vol. 9, pp. 1–28.

Kay, Judy, 2000. Stereotypes, Student Models, and Scrutability. In Giles Gauthier, Claude Frasson, and Kurt VanLehn, eds., Intelligent Tutoring Systems: 5th International Conference (ITS 2000), Montréal, pp. 19–30. Berlin: Springer-Verlag.

Lee, Yoon-Hee. 1990. *Handling Ill-Formed Natural Language Input for an Intelligent Tutoring System*, Ph.D. diss., Illinois Institute of Technology.

Popescu, Octav and Kenneth Koedinger, 2000. Towards Understanding Geometry Explanations. In Papers from the AAAI Fall Symposium on Building Dialogue Systems for Tutorial Applications, North Falmouth, MA. Menlo Park, CA: AAAI Press Technical Report FS-00-01.

Roche, Emmanuel and Yves Schabes, 1997. *Finite-State Language Processing*. Cambridge, MA: MIT Press.

Rosé, Carolyn Penstein, 2000. A Framework for Robust Semantic Interpretation. In Proceedings of the Sixth Applied Natural Language Processing Conference (ANLP '00) Seattle.

Rosé, Carolyn Penstein, and Reva Freedman, eds., 2000. Papers from the AAAI Fall Symposium on Building Dialogue Systems for Tutorial Applications, North Falmouth, MA. Menlo Park, CA: AAAI Press Technical Report FS-00-01.

Seu, Jai Hyun, 1992. *The Development of an Input Understander for an Intelligent Tutoring System Based on a Sublanguage Study*. Ph.D. diss., Illinois Institute of Technology.

Seu, Jai Hyun and Martha Evens, 1991. Understanding Ill-Formed Input to an Intelligent Tutoring System in an LFG Framework. In Proceedings of the Third Midwest Artificial Intelligence and Cognitive Science Society Conference, Southern Illinois University, Carbondale, pp. 36–40.

VanLehn, K., S. Siler, C. Murray, and W. Bagget, W., 1998. What Makes a Tutorial Event Effective? In Proceedings of the Twenty-first Annual Conference of the Cognitive Science Society, Madison, WI. Mahwah, NJ: Lawrence Erlbaum.

Wells, Gordon, 1999. *Dialogic Inquiry*, Cambridge, UK, Cambridge University Press, pp. 3–50.

Wiemer-Hastings, Peter, Katja Wiemer-Hastings, and Art Graesser, 1999. Improving an Intelligent Tutor's Comprehension of Students with Latent Semantic Analysis. In AI in Education 1999, Le Mans, France, pp. 535–542. Amsterdam: IOS Press.

Zhou, Yujian, Reva Freedman, Michael Glass, Joel A. Michael, Allen A. Rovick and Martha W. Evens. 1999. "What Should the Tutor Do When the Student Cannot Answer a Question?" Proceedings of the 12th International Florida Artificial Intelligence Symposium (FLAIRS '99), Orlando, FL.