A Tutorial Dialogue System with Knowledge-Based Understanding and Classification of Student Explanations

Vincent A.W.M.M. Aleven, Octav Popescu, and Kenneth R. Koedinger

Human-Computer Interaction Institute
School of Computer Science
Carnegie Mellon University
aleven@cs.cmu.edu, octav@cmu.edu, koedinger@cs.cmu.edu

Abstract

We are engaged in a research project to create a tutorial dialogue system that helps students to explain the reasons behind their problem-solving actions, in order to help them learn with greater understanding. Currently, we are pilot-testing a prototype system that is able to analyze student explanations, stated in their own words, recognize the types of omissions that we typically see in these explanations, and provide feedback. The system takes a knowledge-based approach to natural language understanding and uses a statistical text classifier as a backup. The main features are: robust parsing, logic-based representation of semantic content, representation of pedagogical content knowledge in the form of a hierarchy of partial and complete explanations, and reactive dialogue management. A preliminary evaluation study indicates that the knowledge-based natural language component correctly classifies 80% of explanations and produces a reasonable classification for all but 6% of explanations.

1 Introduction

The use of intelligent instructional software is a very promising avenue for improving education. One type of such software, Cognitive Tutors, have been shown to yield a standard deviation improvement in student learning outcomes over traditional instruction (Anderson, et al, 1995; Koedinger, et al., 1997). In spite of this success, Cognitive Tutors and other types of intelligent tutoring systems leave room for improvement relative to human (one-on-one) tutors, who yield a two standard deviation improvement over traditional classroom instruction on average (Bloom, 1984).

A limitation of many current systems is that they teach "at the problem-solving level", meaning that they provide assistance in the context of problem solving, but engage students only indirectly in thinking about the reasons behind the solution steps. They do not ask students to explain their reasoning, for instance, "why did you do this step?" or "what rule can you apply next and why?" or "what does this rule really say?" Students' understanding may improve significantly if the software engaged them in dialogue about such questions.

The cognitive science literature provides some indirect evidence that such dialogues will be instructionally effective. For example, students who study worked-out examples or expository text learn with greater understanding to the extent that they explain the materials to themselves ("self-explanation") (Chi, et al., 1989; 1994). Similarly, students are more likely to discover good solution methods when they explain their problem-solving steps to themselves (Berardi-Coletta, et al., 1985; Gagne & Smith, 1962).

Recently, a number of interesting tutorial dialogue systems have been developed (Rose and Freedman, 2000). A recent evaluation study showed that "knowledge construction dialogues" implemented in a computer tutor can enhance student learning (Rose, at al, in press). Yet few if any dialogue systems focus on having students explain. So far, this has been done only in systems with more traditional user interfaces (Aleven et al., 1999; Conati & VanLehn, 2000).

In developing a dialogue system that helps students to state and improve explanations, major open questions are: What type of tutoring strategies and dialogue phenomena need to be modeled in order to help students state good explanations? What kind of architecture is needed to support these tutorial strategies? Do students indeed learn with greater understanding as a result? In the current project we explore the viewpoint that a system that tutors at the explanation level needs to have a very sophisticated understanding of students' explanations but a less sophisticated dialogue management module. We have developed a prototype dialogue system that helps students state general explanations of their problem-solving steps. Our development effort so far has focused on (a) the system's natural language understanding (NLU) component and (b) identifying and representing the "pedagogical content knowledge" needed to respond to the types of omissions we frequently see in students' explanations.

In this paper, we present the architecture our system. We present results from a preliminary evaluation of the NLU component and our plans for further development of the system's dialogue management component. Finally, we discuss opportunities for extending our system with multi-modal interaction.

2 Analysis of Student Explanations

The design of the Geometry Explanation Tutor, our tutorial dialogue system, has been influenced by our analysis of

Table 1: Examples of Explanations

1.	two adjacent angles that form a line add to 180
2.	the sum of the measures of a linear pair of angles is 180 degrees
3.	linear pairs sum to 180
4.	angles formed by 2 lines are congruent
5.	adjacent angles add to 180
6.	base angles in a triangle are equal
7.	the angles in a triangle are 180
8.	180 minus two other angles
9.	triangle has 180 degrees
10.	angle addition of adjacent angles is sum

several small corpora of written explanations of geometry theorems and definitions, most of them by high-school students (Aleven & Koedinger, 2000; Popescu & Koedinger, 2000). The main findings are as follows: First, there are many ways of correctly stating the same geometry theorem in English—the range of lexical and structural variety is even greater than we expected. For example, explanations 1-3 in Table 1 all are valid explanations of the linear pair theorem. Second, students' failed attempts at stating geometry rules are far more often incomplete than they are incorrect. Examples 4-7 in Table 1 illustrate incomplete ways of stating various geometry rules. These statements are overgeneralizations, because they omit one or more conditions of a general rule. Third, students sometimes state the arithmetic operations by which they found the result (example 8). This of course does little to justify why these operations were the correct thing to do. Fourth, students' abbreviate in ways that violates semantic constraints (examples 1, 3, 5, 7, 9). In these examples, the students talk about angles (or even triangles) when strictly speaking they mean angle measures. Finally, students sometimes use language that is rather sloppy—or semantically ill-formed (example 10).

It follows from this analysis that a tutoring system that tutors at the explanation level must have a robust NLU component that can deal with ill-formed explanations and must be able to respond intelligently to the types of incomplete explanations that students produce.

3 The Geometry Explanation Tutor

The success of many recent NLU systems has shown that a good level of natural language understanding is feasible in a limited semantic domain, even with disfluent input. Several natural language systems for limited domains such as weather services, travel reservations, or the scheduling of meetings have achieved over 80% accuracy in understanding spontaneously produced utterances (Gates et al., 1996; Polifroni, et al., 1997; Ward, 1990). However, geometry tutoring poses challenges beyond those tackled by the applications mentioned above. First, wrong, irrelevant, or hedging responses to the student can be confusing. Therefore, the system must produce analyses

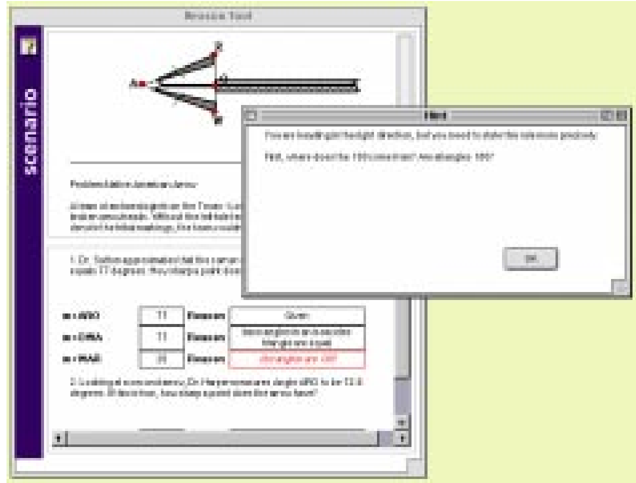

Figure 1: The Geometry Explanation Tutor

that are highly accurate. Further, the system must not only determine what the user intended to say, but also whether she said that with sufficient precision. Finally, when dealing with novices in a domain, one can expect a high proportion of semantically ill-formed utterances.

There are many standard NLU techniques available for tackling limited domains. These techniques include a variety of knowledge-based techniques, like syntactic and semantic analysis or semantic concept spotting using a parser that can skip unrecognized input (Ward, 1990; Gavalda, 1998). There are also a variety of statistical techniques (latent semantic analysis, hidden markov models, statistical language models, mutual information (Jurafsky et al, 1997; Manning & Schütze, 1999). We opted for a knowledge-based approach to NLU, rather than a purely statistical approach, primarily because we expected it to be better at producing detailed and accurate analyses of student input (Popescu & Koedinger 2000). However, in a limited sense, we do try to take advantage of the complementary strengths of statistical and knowledge-based approaches: We use a statistical text classifier as a backup.

Our prototype dialogue system is based on an existing Cognitive Tutor for geometry problem solving, the PACT Geometry Tutor (Aleven, et al., 1999). This tutor was developed, by our research group, as an integrated part of a full-year high-school geometry course. Like all other Cognitive Tutors, this tutor assigns problems to students on an individual basis. It monitors students as they work through these problems, giving guidance on intermediate steps and solutions. The tutor provides just-in-time feedback in response to common difficulties that students have. Upon students' request, it provides context-sensitive hint messages. This is made possible by having a cognitive model that represent the skills of an ideal student in the given domain, represented in the form of production rules. The model enables the tutor to analyze the student's problem-solving activities in terms of underlying skills, and to estimate how well each individual student masters these skills (Anderson, et al, 1995). The PACT Geometry Tutor is in regular use in about five schools, most of them in the Pittsburgh area, and is being marketed nationwide.

The Geometry Explanation Tutor was created by adding dialogue capabilities to the PACT Geometry tutor (see Figure 1). The current prototype tutor engages students in a restricted form of dialogue in order to help them state general explanations that justify their problem solving steps (Popescu & Koedinger, 2000; Aleven, et al., in press). The system has been pilot-tested with our colleagues and staff as well as with 2 high-school students. Below is an example of an actual dialogue with the system, in which a hypothetical student explains the isosceles triangle theorem. This theorem (stated somewhat informally) says that the base angles in an isosceles triangle are congruent. The tutor feedback helps the student in refining an explanation that initially is imprecise. The annotations shown in italics are explained below.

1. **Student:** angles are equal
Classified as: CONGRUENT-ANGLES
2. **Tutor:** Are any two angles congruent? What tells you that there are two congruent angles here?
3. **Student:** angles in a triangle are equal
Classified as: CONGRUENT-ANGLES-IN-TRIANGLE
4. **Tutor:** That is a bit imprecise. Are ALL angles in ALL triangles congruent?
5. **Student:** angles in an isosceles triangle are equal
Classified as: CONGRUENT-ANGLES-IN-ISOSCELES-TRIANGLE
6. **Tutor:** Correct, except for one thing: are all three angles in an isosceles triangle congruent?
7. **Student:** base angles in an isosceles triangle are equal
Classified as: ISOSCELES-TRIANGLE-BASE-ANGLES
8. **Tutor:** CORRECT.

3.1 System Architecture

The Geometry Explanation Tutor is based on the standard Cognitive Tutor architecture (Anderson et al., 1995), augmented with a NLU component (see Figure 2). The task of the NLU component is to build a semantic representation of the student's input sentence and to classify that representation with respect to a hierarchy of explanation categories, discussed below. The task of the Cognitive Tutor module is to decide how to respond to the student, based on the classification of the explanation.

The knowledge base of the system's NLU component contains an ontology of the domain as well as the hierarchy of explanation categories mentioned above. The knowledge base is implemented using the Loom term description system (MacGregor, 1991). The ontology covers the material of the Angles unit of the tutor curriculum, which deals with the geometric properties of angles. The ontology contains concepts representing geometry objects such as angles and lines, as well as relations such as congruency,

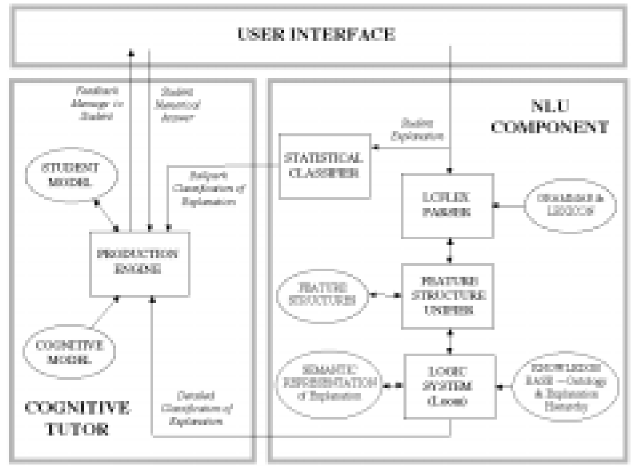

Figure 2: Architecture of the Geometry Explanation Tutor

adjacency, etc. Currently, the knowledge base contains definitions for about 310 concepts and 90 relations.

The explanation categories in the knowledge base represent ways of stating each geometry rule correctly, as well as frequently occurring ways of stating rules incorrectly. The categories are based to a large degree on the analysis of our corpora of student explanations. We have identified about 140 explanation categories related to the 25 geometry rules that make up the tutor's Angles unit. An excerpt of this hierarchy is shown in Figure 3. Each node represents a class of explanations that have the same meaning, but may have vastly different surface forms. A canonical example of a sentence that falls in each category is shown in each node. Explanation categories at the bottom of the hierarchy represent correct and complete ways of stating the isosceles triangle rule. Explanations categories higher up in the hierarchy represent progressively more incomplete ways of stating this rule. Each node in the hierarchy has attached to it a feedback message that would be appropriate to present to the student when an explanation is classified under the given category.

Student input is parsed using a unification-based approach. We employ the LCFLEX parser, an active chart parser (Rose & Lavie, 1999) in combination with a feature structure unifier. We have developed a grammar of about 200 rules. The Loom classifier is used to test the coherence of candidate semantic representations with respect to the semantic constraints expressed in the system's domain ontology. When a semantic representation has been constructed, the Loom classifier automatically classifies it with respect to the explanation categories.

The Cognitive Tutor module determines the tutor's response to the student, based on the classification of the explanation. For explanation steps, the tutor runs its cognitive model to determine which geometry rule needs to be explained. It then selects an appropriate feedback message, taking into account whether the student's explanation is a (a) full statement of the relevant geometry

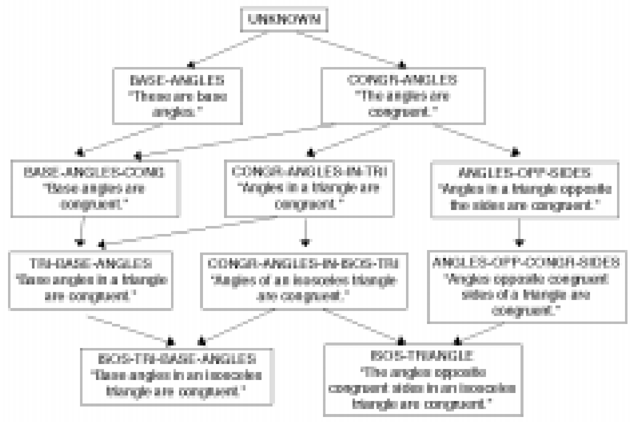

Figure 3: Excerpts from the explanation hierarchy, represented in the system's knowledge base

rule, (b) a partial statement, (c) only the name of a geometry rule, or (d) focuses on the wrong geometry rule.

For example, category ISOSCELES-TRIANGLE-BASE-ANGLES (shown at the bottom left in Figure 3) represents one way of correctly and completely stating the isosceles triangle theorem. The student's explanation in step 7 of the dialogue shown above is classified as an instance of this category. Explanations in this category get the thumbs up from the tutor (step 8), provided of course that the isosceles triangle theorem indeed justifies the step being explained. The tutor determines this using its cognitive model. As an example of a category of partially correct explanations, category CONGRUENT-ANGLES-IN-TRIANGLE represents statements saying that the angles in triangles are equal (see Figure 3, middle). In response to explanations in this category, the tutor suggests that the statement by the student is an overgeneralization (steps 3 and 4 in the dialogue shown above). It does so simply by printing the feedback message attached to the category under which the student explanation was classified, CONGRUENT-ANGLES-IN-TRIANGLE.

We have begun to experiment with the integration of a statistical Naïve Bayes text classifier (Mitchell, 1997, Ch. 6). When the knowledge-based NLU component fails to produce an analysis, the statistical classifier is used to determine whether the student's explanation is in the ballpark (i.e., focuses on the correct geometry rule). If so, the tutor will print a feedback message saying that the student appears to be on the right track. This helps in dealing with unexpected input. We are looking for further ways to leverage the statistical text classifier.

3.2 Knowledge-Based NLU

Two specific problems that were addressed in developing the knowledge-based NLU component are the resolution of anaphora and the resolution of metonymy. The system currently resolves references within sentences. It does so as it builds the semantic representation of a sentence, using knowledge attached to the grammar rules. This enables the system to resolve most of the anaphoric references we observed in our corpora. The system currently does not deal with referents following a pronoun, choice among multiple possible referents (at least not when syntactic and semantic constraints do not uniquely identify the referent), references across sentences or dialogue turns, and references to particular elements in the geometry problem under consideration (such as "the previous angle"). However, in the current system, none of these problems are particularly acute, since students are asked only to provide general statements of geometry rules. They are not supposed to refer to specific objects in the problem or refer back to entities previously mentioned in the dialogue. The experience so far indicates that students have little trouble following these directions.

Metonymy is the phenomenon where one refers to an entity using a closely related entity (Jurafsky & Martin, 2000). In the geometry domain, students frequently use abbreviations like "the angles in a triangle sum to 180", where they mean that the measures, not the angles themselves, sum to 180 degrees. The challenge in dealing with metonymy is to build semantic representations that conform to semantic constraints, when the input sentence does not. The NLU component takes care of this during the semantic processing of sentences. When the NLU component detects that the input violates semantic constraints, it tries to recover missing structure by searching for a conceptual link in the "semantic vicinity" of the concepts that are involved (e.g., angles and sum). This enables it to deal with cases of metonymy that involve sets and measures, which together cover the vast majority of the instances of metonymy we have seen in the corpora. For example, the following examples can be handled: "the angles sum to 180", "vertical angles are equal" and "a linear pair sums to 180" (double metonymy).

3.3 Classification of Student Explanations

As mentioned, Loom's classifier is used to classify the semantic representation of explanations with respect to the system's hierarchy of explanation categories. For example, the sentence "It's a triangle, so the sum of its angles is 180" is a complete and correct statement of the triangle sum rule. The semantic representation of this sentence should therefore be classified under the category ANGLES-SUM-OF-TRIANGLE-180-REASON. Let us see how the system takes care of this.

The NLU component builds the semantic representation shown in Figure 4. This representation consists of a number of instances of concepts defined in the knowledge base. These concepts (ANGLE, TRIANGLE, etc.) form part of the system's geometry ontology. The two instances of concept BEING&HAVING shown in Figure 4 correspond to the two clauses in the sentence. The verb "to be" is represented using an instance of BEING&HAVING whose attribute and attribuend are the same. One of these instances is also an instance of the REASON concept. Since the sentence was given as an explanation of an

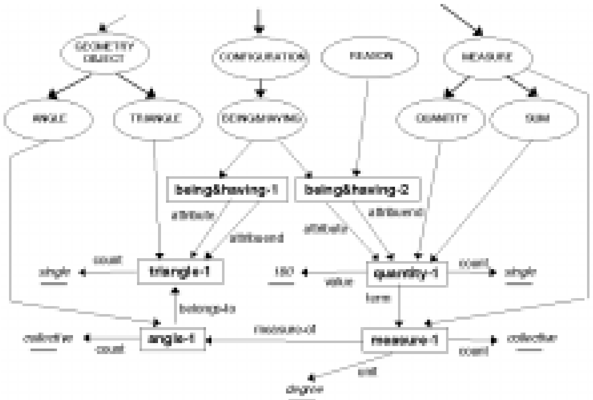

Figure 4: Semantic structure built by the NLU component for the input sentence: “It's a triangle, so the sum of its angles is 180” Ovals denote concepts, rectangles denote individuals, and underlined text denotes a constant.

answer, the NLU component has asserted that it is an instance of the REASON concept. The other individuals in Figure 4 correspond in obvious ways to the entities mentioned in the sentence. The belongs-to link from angle-1 to triangle-1 reflects the fact that the system has resolved the pronoun “its”. In order for the Loom classifier to recognize that an individual is an instance of a concept, this individual has to satisfy the conditions expressed in the definition of this concept. The definition, for concept ANGLES-SUM-OF-TRIANGLE-180-REASON, in Loom's terminological language, is as follows:

```
(defconcept angles-sum-of-triangle-180-reason
:is (:and reason
        (:some topic
          (:and sum
              (:the value value-180)
              (:some term
                (:and measure
                  (:some measure-of
                        (:and angle
                              (:some vertex-of triangle))))))))
```

Translated somewhat loosely into English, the definition says that an ANGLES-SUM-OF-TRIANGLE-180-REASON is a reason that says that the sum of some measure(s) of some angle(s) of some triangle is 180 degrees. This definition relies on a number of other definitions, such as those of the concept sum and the relation vertex-of, which in turn rely on other definitions:

```
(defconcept sum
 :is-primitive (:and measure
          (:at-least 1 term)
          (:all term measure)))

(defrelation vertex-of
 :is (:and belongs-to
        (:domain angle)
        (:range polygon)))
```

Now we are in a position to understand why Loom classifies instance being&having-2 under concept ANGLES-SUM-OF-TRIANGLE-180-REASON. As required by the concept definition, being&having-2 is an instance of concept REASON. Further, the definition requires a topic relation. Figure 4 does not show any topic relations, but attribute and attribuend relations are in fact defined as topic relations. Finally, individual quantity-1 is an instance of SUM and satisfies all other restrictions placed on the value and term relations stated in the definition of ANGLES-SUM-OF-TRIANGLE-180-REASON: some term of quantity-1 is a measure of an angle that belongs to a triangle. Interestingly, that inference depends on the presence of the belongs-to link from angle-1 to triangle-1. Without anaphora resolution, this example would not have been classified correctly.

3.4 Dialogue Management

Currently, the system's response in each dialogue turn is based only on the classification of the student's last explanation attempt. No further context is taken into account. This way, the tutor can respond to the types of omissions we often see in students’ explanations, and under the right circumstances can even produce a sense of coherent dialogue, as illustrated in the example dialogue. However, because of its simple dialogue management scheme, the system's range of dialogue strategies is currently very limited. For example, the system has no memory of what went on before in the dialogue. If in successive attempts at explaining a geometry rule, a student regresses (e.g., types an explanation that was worse than the previous attempt), the system currently blithely gives a locally-appropriate feedback message without any knowledge of the global lack of progress. Similarly, when the student stagnates (i.e., when consecutive unsuccessful attempts at explaining a geometry rule are classified under the same category), the system will simply repeat its previous feedback message, in spite of the fact that this message apparently did not help the student. It would be better if the system would try an alternative approach to help the student. Finally, the tutor is not able to engage in multi-turn strategies or to lead students through a directed line of reasoning, as human tutors often do. This makes it difficult to have the system engage in such venerable mathematical strategies as showing counterexamples, when students (as they tend to do) state overly general rules.

We plan to address these limitations insofar as this turns out to be necessary in order to help students learn with greater understanding. It will be necessary to extend the system so that it maintains a dialogue history. Further, the system probably needs a dialogue planning mechanism. We are currently investigating several alternatives (Freedman, 2000; Heffernan & Koedinger, 2000; Larsson, et al, 1999). In extending the system, we plan to be guided by results of frequent preliminary evaluation studies, adding more sophisticated facilities only when the data suggest that they are needed. Exactly how far we will have to push the system's dialogue management is an interesting open question. Ultimately, our goal is to find out how best to help students learn through self-explanation.

Table 2: Classification accuracy (number of explanations classified) of the knowledge-based NLU component

Correct	520	80%
Overly General	91	14%
Incorrect	37	6%
Total	648	100%

4 Preliminary Evaluation

We conducted a preliminary evaluation of the classification accuracy of the knowledge-based NLU component. As test data we used a corpus of 648 explanations collected during a session in which about 20 of our colleagues and staff worked on the system. All explanations gathered during this session were labeled by hand by two authors of the paper, who assigned each explanation to the most specific category in the explanation hierarchy to which it belongs. New categories were invented as needed. A total of 92 categories were represented in the data set. The system's knowledge base was then extended to include definitions for many of the new categories. The system was run to classify the 648 explanations.

As shown in Table 2, the system classified 80% of the explanations correctly. Of the correctly classified explanations, 80% (420 out of 520) fall under categories of full or partial explanations. The rest were references, meaning that the student stated only the name of a geometry rule. The system classified a further 14% of the explanations under categories that were too general, although not strictly wrong. The remaining 6% of explanations were either not classified or under categories that were unrelated to the correct category.

An accuracy score of 80% is very encouraging, especially given the fact that we are dealing with a very fine-grained classification task, where small differences between categories are the rule rather than the exception. There are two caveats. First, the accuracy results were obtained with a data set that was used during the development of the system. Second, the data set used was obtained with subjects who are more advanced than the students in the target population (high-school students). More work is needed before we expect to see the same accuracy score with students from the target population.

5 Opportunities for Multimodal Interaction

Geometry involves both verbal and visual modes of learning. Geometry tutoring is therefore a very interesting domain to explore the advantages of multimodal interaction in an intelligent tutoring system. Multimodal interaction opens up the opportunity to have the system alternate between visual and verbal tutoring strategies, which is likely enhance the system's pedagogical effectiveness. For example, when students state overly general explanations (as they tend to do) a multimodal system could communicate this fact using purely verbal strategies (such as those illustrated for example in steps 4 and 6 of the example dialogue shown above). But it could also show diagrams of counterexamples. ("You are saying that the base angles in a triangle are congruent. But here is a triangle none of whose angles are congruent.")

In addition, in a multimodal system students could alternate between verbal and visual ways of expressing explanations. While it is plausible that students will learn with greater understanding as they explain their problem-solving steps in their own words—the hypothesis that is the focus of our current research—sometimes it may be more effective if students explain geometry knowledge with a picture. For example, instead of saying "the measure of an angle formed by two adjacent angles is equal to the sum of the measures of those angles" it may be easier for students to draw a diagram of two adjacent angles (using a drawing tool integrated in the tutoring environment) and say: "whenever you have two angles next to each other like this, you can sum the two to get the third one". In deciding how to follow up on that explanation, the tutor could then choose between visual and verbal strategies.

Further, it is quite likely that multimodal interaction will streamline the interaction between tutor and students and possibly make it more robust (Oviatt, 1999). We plan to extend our dialogue system to handle dialogue about problem solving and to handle explanations about how geometry rules apply to a given problem. We foresee that much of the student-system interaction will take place in a diagram window and a dialogue window, rather than in the worksheet window shown in Figure 1. Both student and system will freely mix references to geometry objects expressed in the traditional mathematical way ("angle ABC"), referring expressions in English ("the segment perpendicular to the highlighted part"), references expressed by clicking on the relevant object in the diagram. It is an interesting question how this kind of multimodal interaction will influence learning and in particular, whether students will come away with qualitatively different knowledge as a result of such interaction.

From an educational/cognitive science point of view, little work has been done to understand the nature of the interactions between perceptual and verbal learning systems. Geometry is an interesting domain to investigate these interactions. Critical research questions include the following: To what extent and in what way does an integrated system of visual and verbal representations develop and contribute to understanding? Do these representations provide mutual support in the learning process? A multimodal system seems an excellent vehicle to explore such questions.

6 Conclusion

We are working to investigate how an intelligent tutoring system can best help students learn through self-explanation. We have developed a prototype tutorial dialogue system, the Geometry Explanation Tutor, that helps students state explanations of geometry rules in their own words and help them improve their explanation through dialogue.

Architecturally, we explore the viewpoint that a system that helps students learn through self-explanation must have a sophisticated NLU component and can get by with a less sophisticated dialogue management component. Thus, we opted for a knowledge-based approach to NLU. The results of a preliminary evaluation study are very encouraging. The NLU component produced accurate classifications for 80% of explanations and produced somewhat reasonable classifications for all but 6% of explanations. These results suggest that adopting a knowledge-based approach to NLU was a good choice. The system's dialogue management component takes a simple “classify and react” approach. In each dialogue cycle, it produces a response based primarily on the classification of the student's explanation. At times, this produces reasonable dialogue. However, it is likely that more will be needed in order to help *all* students state accurate and complete explanations.

We plan to improve both the NLU component and the dialogue management component. In doing so, we will let ourselves be guided by the results from further pilot studies, so as to focus our efforts on adding functionality that is most likely to make a difference in student learning. We will conduct an empirical study to evaluate whether explaining in one's own words is the best way to learn through self-explanation.

7 References

Aleven, V. & Koedinger , K. R. (2000). The Need for Tutorial Dialog to Support Self-Explanation. In C. P. Rose & R. Freedman (Eds.), *Building Dialogue Systems for Tutorial Applications, Papers of the 2000 AAAI Fall Symposium* (pp. 65-73). Menlo Park, CA: AAAI Press.

Aleven V., Popescu, O. & Koedinger, K. R. (in press). Towards tutorial dialog to support self-explanation: Adding natural language understanding to a cognitive tutor. Accepted for publication in the *Proceedings of the 10th International Conference on AI and Education, AIED 2001.*

Aleven, V., Koedinger, K. R. & Cross, K. (1999). Tutoring Answer Explanation Fosters Learning with Understanding. In S. P. Lajoie & M. Vivet (Eds.), *Proceedings of of the 9th International Conference on AI and Education, AIED-99,* (pp. 199-206). Amsterdam: IOS Press.

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & R. Pelletier, (1995). Cognitive tutors: Lessons learned. *The Journal of the Learning Sciences,* 4,167-207.

Berardi-Coletta, B., Buyer, L. S., Dominowsky, R. L., & Rellinger, E. R. (1995). Metacognition and Problem-Solving: A Process-Oriented Approach. *Journal of Experimental Psychology: Learning Memory, and Cognition,* 21 (1) 205-223.

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

Chi, M. T. H., Bassok, M., Lewis, M. W., Reimann, P., & Glaser, R. (1989). Self-Explanations: How Students Study and Use Examples in Learning to Solve Problems. *Cognitive Science,* 13, 145-182.

Chi, M. T. H., de Leeuw, N., Chiu, M., & Lavancher, C. (1994). Eliciting Self-Explanations Improves Understanding. *Cognitive Science,* 18, 439-477.

Conati C. & VanLehn K. (2000). Toward Computer-Based Support of Meta-Cognitive Skills: a Computational Framework to Coach Self-Explanation. *International Journal of Artificial Intelligence in Education,* 11.

Freedman, R. (2000). Using a Reactive Planner as the Basis for a Dialogue Agent. *Proceedings of the Thirteenth Florida Artificial Intelligence Research Symposium (FLAIRS 2000).* Orlando.

Gagné, R. M., & Smith, E. C. (1962). A Study of the Effects of Verbalization on Problem Solving. *Journal of Experimental Psychology,* 63(1), 12-18.

Gates, D., Lavie A., Levin, L., Waibel, A., Gavalda, M., Mayfield, L., Woszczyna, M., & Zhan, P. (1996). End-to-end Evaluation in JANUS: a Speech-to-speech Translation System. In the *Proceedings of the Workshop on dialogue.* European Conference on Artificial Intelligence. Budapest.

Gavalda, M. (1998). The Soup Home Page. http://www.is.cs.cmu.edu/ISL.speech.parsing.soup.html.

Heffernan, N. T. & Koedinger, K. R. (2000). Intelligent Tutoring Systems are Missing the Tutor: Building a More Strategic Dialog-Based Tutor. In C. P. Rose & R. Freedman (Eds.), B*uilding Dialogue Systems for Tutorial Applications, Papers of the 2000 AAAI Fall Symposium* (pp. 14-19). Menlo Park, CA: AAAI Press.

Jurafsky, D., Bates, R., Coccaro, N., Martin, R., Meteer, M., Ries, K., Shriberg, E., Stolcke, A., Taylor, P., Van Ess-Dykema, C. (1997b). Automatic Detection of Discourse Structure for Speech Recognition and Understanding. In the *Proceedings of the IEEE Workshop on Speech Recognition and Understanding.*

Jurafsy, D. & Martin, J. H. (2000). *Speech and Language Processing.* Upper Saddle River, NJ: Prentice Hall.

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

Larsson, S., Bohlin, P., Bos, J., & Traum, D. (1999). TRINDIKIT 1.0 Manual. Göteborg University. http://www.ling.gu.se/research/projects/trindi

MacGregor, R, (1991). The Evolving Technology of Classification-Based Knowledge Representation Systems. In J. Sowa (Ed.), *Principles of Semantic Networks: Explorations in the Representation of Knowledge.* San Mateo, CA: Morgan Kaufmann.

Manning C. D. & Schütze, H. (1999). *Foundations of Statistical Natural Language Processing.* Cambridge, MA: The MIT Press.

Mitchell, T, 1997. *Machine Learning.* McGraw-Hill.

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

Polifroni, J., Seneff, S., Glass, J., & Hazen, T.J. (1997). Evaluation Methodology for a Telephone-based Conversational System. In *Proceedings of the First International Conference on Language Resources and Evaluation.* (pp. 42-50) Granada, Spain.

Popescu, O., & Koedinger, K. R. (2000). Towards Understanding Geometry Explanations In C. P. Rose & R. Freedman (Eds.), *Proceedings of the AAAI 2000 Fall Symposium, Building Dialog Systems for Tutorial Applications* (pp. 80-86). Menlo Park, CA: AAAI Press.

Rose C. P. & Freedman R. (Eds., 2000). Building Dialogue Systems for Tutorial Applications, Papers of the 2000 AAAI Fall Symposium. Menlo Park, CA: AAAI Press.

Rose, C. P., Jordan, P., Ringenberg, M., Siler, S., VanLehn, K., & Weinstein, A. (in press). Interactive Conceptual Tutoring in Atlas-Andes. Accepted for publication in the *Proceedings of the 10th International Conference on AI and Education, AIED 2001.*

Rose, C. P. & Lavie, A. (1999). LCFlex: An Efficient Robust Left-Corner Parser. User's Guide, Carnegie Mellon University.

Ward, W. (1990). The CMU Air Travel Information Service: Understanding spontaneous speech. In *Proceedings of the DARPA Speech and Language Workshop.*