Understanding Students’ Explanations in Geometry Tutoring

Octav Popescu, Vincent Aleven, and Kenneth Koedinger
Human-Computer Interaction Institute
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213 USA
octav@cmu.edu, aleven@cs.cmu.edu, koedinger@cmu.edu

Abstract

Precise Natural Language Understanding is needed in Geometry Tutoring to accurately determine the semantic content of students’ explanations. The paper presents an NLU system developed in the context of the Geometry Explanation Tutor. The system combines unification-based syntactic processing with description logics based semantics to achieve the necessary accuracy level. Solutions to specific semantic problems dealing with equivalence of semantic representations are described. Experimental results on classification accuracy are also presented.

1 Introduction

The Geometry Cognitive Tutor is designed to help high school and middle school students learn geometry. As a kind of Cognitive Tutor (Anderson et al. 1995), the system is based on an underlying cognitive model, implemented as an ACT-R production system (Anderson and Lebiere 1998), of both novice and ideal student knowledge. This model is used to monitor student performance and to provide assistance just when students need it and in a context that demands it. Currently the Geometry Cognitive Tutor is in regular use (two days per week) in about 350 schools around the US.

The tutor proposes problems to students and checks their solutions step by step. It can also provide context-sensitive hints at each step in solving the problem, as needed. The students are asked to compute various elements of the problem (mostly angle measures) by applying the theorems and definitions they have learned. In case the results are correct, the students are also asked to provide a justification for their results. In the version of the tutor that is currently in use in schools, the justification consists in choosing the right theorem or definition out of a menu. The choice is accepted or rejected by the tutor, accordingly.

The use of this tutor in combination with classroom instruction has been shown to lead to improvements in students’ test scores over traditional classroom instruction alone (Koedinger et al. 1997). Experiments have also shown that the use of menu-based justifications help students learn with greater understanding over a tutor that does not ask for justifications (Aleven and Koedinger 2002).

However there is room for improvement. Classroom observation shows that some students try to game the system by choosing each item in the menu in turn, until one is accepted. Even when this is not the case, it is possible that simply recognizing the name of the correct theorem or definition out of a menu does not imply that the student is fully knowledgeable about the actual content of the theorem involved. Thus it is plausible that asking the students to express the content of these theorems and definitions in their own words as a form of self-explanation could lead to a deeper level of their understanding.

To verify this hypothesis we are currently developing the Geometry Explanation Tutor. This version is based on the Geometry Cognitive Tutor, with the modification that the justification consists of a natural language sentence that expresses the content of the theorem or definition used to derive the result, in free form. The tutor checks the semantic content of the explanation and provides feedback on its correctness and completeness. In case the explanation quality is deemed not good enough, the student is allowed to refine it, until it becomes acceptable.

Thus, one of the main problems that the Geometry Explanation Tutor faces is to determine with accuracy the semantic content of students’ utterances. There are many different ways to express the same semantic content, which have to be recognized as being equivalent.

The determination of equivalence relations has to work reliably over variation of syntactic structure, variation of content words, or a combination of both. For example, the sentences below all express the same geometry theorem, about the measures of angles formed by other angles.

> An angle formed by adjacent angles is equal to the sum of these angles.
>
> The measure of an angle formed by other angles is equal to the sum of the measures of those adjacent angles.
>
> An angle's measure is equal to the sum of the two adjacent angles that form it.
>
> The sum of the measures of two adjacent angles is equal to the measure of the angle formed by the two angles.
>
> The measure of an angle formed by two adjacent angles is equal to the sum of the measures of the two angles.
>
> If adjacent angles form an angle, its measure is their sum.
>
> When an angle is formed by adjacent angles, its measure is equal to the sum of those angles.

The process has also to be consistent, so no unwarranted conclusions are derived from the text, and robust, in an environment of imprecise or ungrammatical language, as uttered more often than not by high school students. Many times this content equivalence relies on inferences specific to the domain of discourse. Our hypothesis is that such a high-precision recognition process needs to be based on contextual information about the domain of discourse modeled in a logic system.

2 The System's Architecture

The system's overall architecture is presented in Figure 1 below. The *interface module* takes the input sentence from the tutor, word by word, in real time, and after some preprocessing and spelling checking, it passes it to the chart parser. It also passes the results back to the tutor. The *chart parser* is the main engine of the system. It uses linguistic knowledge about the target natural language from the *unification grammar* and the *lexicon*. The parser used currently is LCFlex, a left-corner active-chart parser developed at the University of Pittsburgh (Rosé and Lavie 1999). The parser calls the *feature structure unifier* in order to process restrictions attached to grammar rules and build *feature structures* for each phrase successfully recognized. These feature structures store lexical, syntactic, and semantic properties of corresponding words and phrases. The parser uses an *active chart* that serves as a storage area for all valid phrases that could be built from the word sequence it received up to each point in the process.

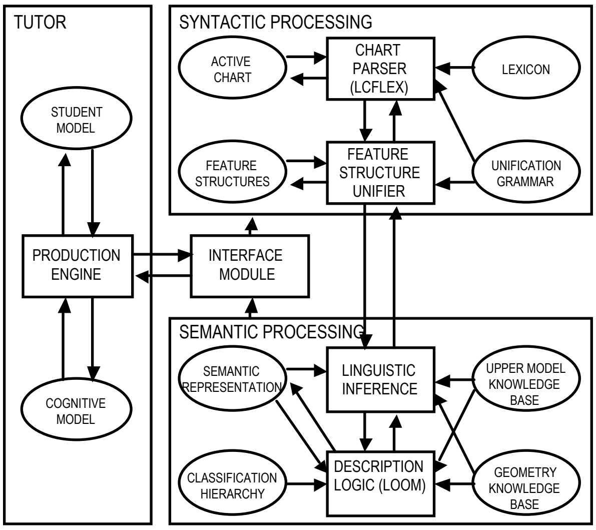

Figure 1. System Architecture

Some of the restrictions in the grammar are directives to the *description logic system*, currently Loom (MacGregor 1991). The logics system relies on a model of the domain of discourse, encoded as concepts, relations, and production rules, in the two *knowledge bases.* Concepts and relations stand for predicates in the underlying logic. Production rules perform additional inferences that are harder to encode into concepts and/or relations.

The *linguistic inference* module mediates the interaction between the feature structure unifier and the description logics system. This module is responsible for performing semantic processing that is specific to natural language understanding, like compositional semantics, resolving metonymies and references, and performing semantic repairs.

Based on this knowledge base, the logic system builds compositionally a model-theoretic *semantic representation* for the sentence, as a set of instances of various concepts connected through various relations. An instance corresponds to a discourse referent in the sentence. The logic system performs forward-chaining classification of resulting instances, and also ensures semantic coherence of the semantic representation.

The logic system then uses a classifier to evaluate the semantic representation against a *classification hierarchy* of valid representations for geometry theorems. The results of the classification are passed back to the tutor.

1.1 Example of Compositional Build of the Semantic Representation

To see how the compositional building of semantic representations works, let's consider the last step in parsing the sentence:

```
The measure of a right angle is 90 degrees.
```

A set of simplified knowledge base definitions necessary for building its representation, in Loom's definitional language, is:

```
(defconcept Configuration
  :is-primitive (:and Thing (:all participant Thing)))
(defconcept Being&Having
  :is-primitive (:and Configuration
                      (:at-most 2 participant)))
(defconcept Ascription
  :is (:and Being&Having (:exactly 1 attribuend)
                         (:exactly 1 attribute))
(defproduction ascription-production
  :when (:detects (Ascription x))
  :do ((combine-instances (x attribute)
                          (x attribuend))))
(defconcept Geometry-Unit
  :is (:and Unit (:one-of 'degree 'radian 'meter 'foot)))
(defconcept Angle-Unit
  :is (:and Geometry-Unit (:one-of 'degree 'radian)))
(defconcept Geometry-Measure
  :is (:and Measure (:the unit Geometry-Unit)))
(defconcept Angle-Measure
  :is (:and Geometry-Measure (:the unit Angle-Unit)))
(defconcept Geometry-Object
  :is-primitive (:and Spatial
                      (:all measure Geometry-Measure)))
(defproperty Right :domain Geometry-Object)
(defconcept Angle
  :is-primitive (:and Geometry-Object
                      (:the measure Angle-Measure)))
(defconcept Right-Angle :is (:and Right Angle))
```

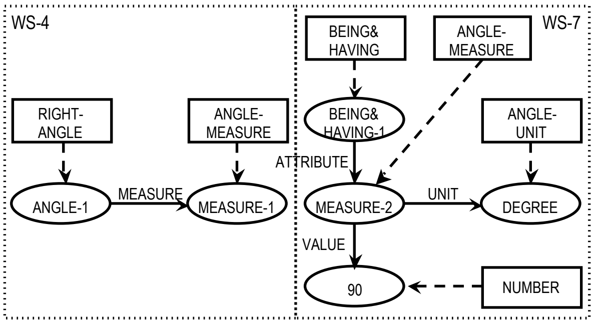

Figure 2. Example of Semantic Representation

Based on these definitions and the rules of the grammar, the system builds the representation below for the subject of the example above, expressed in Loom's assertional language:

```
(tell (:about measure-1 (:create Angle-Measure)))
(tell (:about angle-1 (:create Right-Angle)
        (measure measure-1)))
```

The system also builds this structure for the verb phrase:

```
(tell (:about measure-2 (:create Angle-Measure)
        (unit 'degree) (value 90)))
(tell (:about being&having-1 (:create Being&Having)
        (attribute measure-2)))
```

The two structures are illustrated in Figure 2. Then the parser applies the grammar rule for clauses, given below in simplified form. *Connect-semantics* will assert an *attribuend* relation between instances *being&having-1* and *measure-1,* relation specified in the lexicon as the semantic role of the verb's subject.

```
(<Cl> ==> (<NP> <VP>)
      ((x0 = x2)
       ((x0 subject) = x1)
       ((x0 semantics) <=
         (connect-semantics (x2 semantics)
           (x2 subject sem-role) (x1 semantics)))))
```

Loom then classifies *being&having-1* as an instance of the more specific concept *Ascription,* and this classification triggers production *ascription-production.* The production will combine the two measure instances, *measure-1* and *measure-2*, into a single instance, resulting in the structure below:

```
(tell (:about measure-1 (:create Angle-Measure)
        (unit 'degree) (value 90)))
(tell (:about angle-1 (:create Right-Angle)
        (measure measure-1)))
(tell (:about being&having-1 (:create Ascription)
        (attribute measure-1) (attribuend measure-1)))
```

The structure is shown in Figure 3.

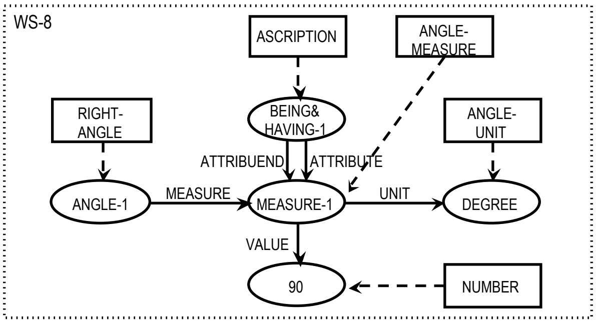

Figure 3. Resulting Semantic Representation

This structure is then classified against a hierarchy of concept definitions representing classes of possible explanations. A few of them are shown in Figure 4.

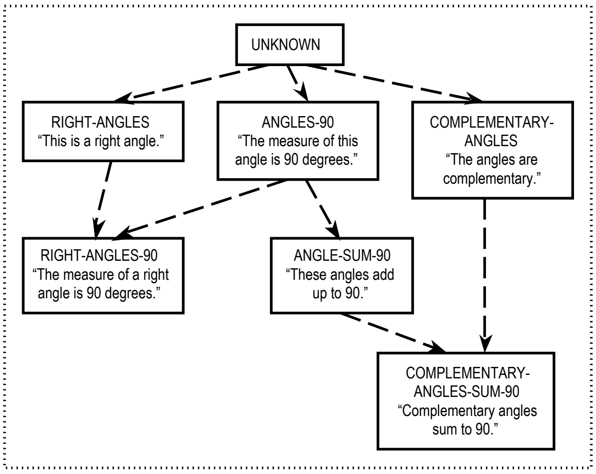
UNKNOWN
RIGHT-ANGLES
"This is a right angle."
ANGLES-90
"The measure of this angle is 90 degrees."
COMPLEMENTARY-ANGLES
"The angles are complementary."
RIGHT-ANGLES-90
"The measure of a right angle is 90 degrees."
ANGLE-SUM-90
"These angles add up to 90."
COMPLEMENTARY-ANGLES-SUM-90
"Complementary angles sum to 90."

Figure 4. Partial Classification Hierarchy

3 Specific Problems in Students' Explanations

The basic approach we take to the content equivalence problem is to provide the right inference rules to make the logic system derive the same semantic representation for all sentences that are semantically equivalent. Below we present how this is done in some specific cases.

1.2 Variation of Syntactic Structure

Even when the choice of content words is the same, the same meaning can be conveyed through a variety of syntactic structures. Some cases, like that of passive versus active constructs, can be taken care of in the grammar. Other cases require specific knowledge about the domain of discourse. One such situation is that of prepositional phrases attached in different places in the sentence, without changing the meaning, like in these examples:

In a triangle angles opposite to congruent sides are congruent.

Angles opposite to congruent sides in a triangle are congruent.

Angles in a triangle opposite to congruent sides are congruent.

Angles opposite to congruent sides are congruent in a triangle.

The solution in our system comes as a concept definition that identifies the container relation at the assertion level, and percolates it down to involved objects.

```
(defconcept Ascription-Location
  :is (:and Ascription (:at-least 1 belongs-to))
  :implies
          (:and (:relates belongs-to attribuend belongs-to)
                (:relates belongs-to attribute belongs-to))))
```

A similar case is that of using constructs specific to the domain of discourse.

The measures of these two angles are equal.

These two angles are equal in measure.

Knowledge about the semantics of "equal" and "measure" is involved in determining that "equal in measure" means the same thing as "measures ... are equal". We can model this knowledge by defining a rule that will identify cases of "equal in some measurable quantity" and will generate a structure with the meaning of "equal quantity".

```
(defconcept Equal-in
  :is (:and Equal (:some belongs-to Measure)))
(defproduction equal-in-production
  :when (:detects (Equal-in ?object))
  :do (combine-semantics ?object
                          (?object belongs-to)))
```

The use of relative and subordinate clauses can also lead to a large variety of syntactic structures without a significant change in meaning:

The sum of the measures of complementary angles is 90 degrees.

If angles are complementary, then the sum of their measures is 90 degrees.

The measures of the angles sum to 90 degrees, because they are complementary angles.

Complementary angles are angles whose measures sum to 90 degrees.

These sentences all express the same theorem about complementary angles using respectively a single clause sentence, a conditional clause, a subordinate clause, or a relative clause. Because the semantic representation we build does not keep any trace of the original syntactic structure, such variations are automatically ignored. For example, the structure built for the first sentence is:

```
(tell (:about measure-1 (:create Geometry-Measure)))
(tell (:about angle-1 (:create Angle)
        Complementary (measure measure-1))
(tell (:about sum-1 (:create Sum)
        (value 90) (unit 'degree) (term measure-1))
(tell (:about being&having-1 (:create Ascription)
        (attribute sum-1) (attribuend sum-1))
```

Ignoring the conditionality, the structures for the two clauses in the second sentence are:

```
(tell (:about angle-1 (:create Angle) Complementary))
(tell (:about being&having-1 (:create Ascription)
        (attribute angle-1) (attribuend angle-1))
(tell (:about thing-1 (:create Thing)
        (measure measure-1)))
(tell (:about measure-1 (:create Geometry-Measure)))
(tell (:about sum-1 (:create Sum)
        (value 90) (unit 'degree) (term measure-1))
```

```
(tell (:about being&having-2 (:create Ascription)
        (attribute sum-1) (attribuend sum-1))
```

All that is needed to achieve semantic equivalence is a reference resolution mechanism that identifies referents at the semantic level with their antecedents. In the example above the system would solve *thing-1* to *angle-1*.

1.3 Variation of Content Words

Many times differences in the content words used in the sentence do not make any difference at the meaning level. An obvious case is that of synonyms. However there are cases when different words are used as synonyms only in certain contexts. For instance:

Angles ABC and BAC are equal.

Angles ABC and BAC are congruent.

Versus:

The measures of angles ABC and BAC are equal.

*The measures of angles ABC and BAC are congruent.

Here the synonymy holds only when the objects involved in the relation are geometry objects, and it is not allowed when they are measures. We can make this distinction by defining "congruent" as a specialized case of "equal":

```
(defrelation equal-to
  :is-primitive (:and relation (:domain Thing)
                               (:range Thing))
  :characteristics :symmetric)
(defrelation congruent-to
  :is (:and equal-to (:domain Geometry-Object)
                     (:range Geometry-Object)))
```

Moreover, we can add a production rule that will perform the inference that if the measures of some objects are equal, then the objects themselves are congruent. This rule will make the third sentence above be recognized as equivalent to the first two sentences.

```
(defconcept Equal-Measure
  :is (:and Measure (:at-least 1 equal-to)))
(defproduction equal-measure-production
  :when (:detects (Equal-Measure ?measure))
  :do (connect-semantics
          (?measure measure-of) congruent-to
          (?measure equal-to measure-of)))
```

A related phenomenon is that of using very generic functional words in usual language to denote specific relations among the concepts of the domain.

The angles of a linear pair sum to 180.

The angles that form a linear pair sum to 180.

The angles that are elements of a linear pair sum to 180.

In these examples the angles are actually the elements of the linear pair. However in the first two sentences the relation is expressed either through a preposition, or through a generic verb like "form". Recovering the explicit relation and thus being able to determine that the three examples above are semantically equivalent requires once again a model of the domain of discourse. We can model this first by defining the *element-of* relation as a more specific version of the generic relation *belongs-to* expressed by "of". This definition will make the system build the same representation for the first sentence as for the third one.

```
(defconcept Set :is-primitive Thing)
(defrelation element-of
  :is (:and belongs-to (:domain Thing) (:range Set)))
```

Second, we can define a production rule that recognizes a "form" configuration and asserts a "belongs-to" relation between the arguments, thus generating for the second sentence the same representation as for the first one:

```
(defconcept Form-Configuration
  :is (:and Generalized-Possession
            (:the part Thing)
            (:the whole Thing)))
(defproduction form-configuration-production
  :when (:detects
            (Form-Configuration ?configuration))
  :do (connect-instances (?configuration part)
                         belongs-to
                         (?configuration whole)))
```

Another similar situation is that when students use the definition of a concept expressed in terms of more generic concepts, instead of its name.

Adjacent angles on a line sum to 180 degrees.

Linear angles sum to 180 degrees.

The ability to recognize such examples as being semantically equivalent, with the right degree of generality, is conditioned by the possibility to model the definitions of those specific concepts within the framework of the system. This case can be dealt with by defining "linear angles" as "adjacent angles on a line":

```
(defrelation adjacent-to
  :is-primitive (:and relation
                      (:domain Geometry-Object)
                      (:range Geometry-Object)))
(defconcept Adjacent-Angle
  :is (:and Angle (:at-least 1 adjacent-to)))
(defconcept Angle-on-Line
  :is (:and Angle (:some location Line)))
(defconcept Linear-Angle
  :is (:and Adjacent-Angle Angle-on-Line))
```

1.4 Syntactic Ambiguity

Syntactic ambiguity in many cases does not reflect semantic ambiguity. One such possibility is prepositional phrase attachment. That is, following only the grammar rules, many times a prepositional phrase could be an adjunct/ argument of several preceding components. A deeper look at those alternative attachments reveals that most of them can be discarded because they do not result in a meaningful sentence. However, in absence of detailed knowledge about the meaning of the words in the sentence and their possible interactions, an NLU approach would not be able to disambiguate among them.

The sum of the measures of the three interior angles in a triangle is equal to 180 degrees.

The subject in this example contains three prepositional phrases: "of the measures", "of the three interior angles", and "in a triangle". While the first one can only be attached to one place: the noun "sum", the second one already can be attached to two places: "sum" or "measures", and the third one can be attached to three places: "sum", "measures", or "angles", resulting in a total of 6 different valid parses. By adding appropriate restrictions to the definitions of the concepts involved, our approach can make some of these combinations invalid during the parsing process. In our example we can restrict sums to only apply to elements that are measures, and thus eliminate the attachment of prepositional phrase "of the three interior angles" to "the sum". And then we can restrict the containment relation to have geometry objects on both sides, and thus eliminate the attachment of "in a triangle" to either "the sum" or "the measures".

```
(defconcept Sum
 :is-primitive (:and Measure (:all term Measure)))
(defconcept Object-in-Location
 :is (:and Object (:some location Geometry-Object))
 :implies Geometry-Object)
```

1.5 Reference Resolution Disambiguation

The presence of anaphora in students' explanations results in cases where sentences with different sets of words are semantically equivalent. Recognizing the semantic equivalence of such cases leads to the necessity to have an accurate reference resolution mechanism, which allows us to build the right semantic representation for the sentence.

The resolution of referents to antecedents is done in our system at the semantic level. That is we simply try to merge the semantic representation of the referent with that of the antecedent. This mechanism has the advantage that the logic system will make sure that all semantic constraints associated with the two discourse referents are enforced, so that elements that are incompatible will fail the merge. This takes care both of the number restrictions, as well as all other semantic features, like taxonomic compatibility between the concepts involved.

Finding the right referent for an anaphor is not always easy. Syntactic criteria can help with disambiguation among candidates, but there are cases where they cannot lead to a unique antecedent. Adding semantic constraints to the solution can increase the accuracy considerably.

If the lengths of two sides of triangles are equal, then the measures of the angles opposite them will also be equal.

In this example there are five possible candidates as antecedent for the pronoun "them": "the lengths", "two sides", "a triangle", "the measures", and "the angles". Constraints of the Binding Theory implemented in our system eliminate "the angles", since "them" is a personal pronoun that has to be free within its local domain. Constraints on number eliminate "a triangle", as being singular, while "them" is plural. Then semantic constraints attached to the definition of relation "opposite" can eliminate both "the lengths" and "the measures", by asking that geometry objects can oppose only other geometry objects:

```
(defconcept Object-opposite
  :is (:and Geometry-Object
            (:some opposite-to Thing))
  :implies (:all opposite-to Geometry-Object))
```

4 Performance Evaluation

As a measure of the system's ability to understand students' explanations we evaluated the accuracy of the classification of these sentences with respect to the hierarchy of explanation classes. The evaluation used a set of 700 sentences representing actual explanations provided by high school students during an experimental study in 2003. The classification task consists in associating each sentence with one or more of 200 fine-grained categories, a difficult task even for humans. We used the kappa statistic (Cohen 1960) to measure the inter-rater reliability between the system and two human raters. We used three different measures. First, a "set equality" measure, where two sets of classes match only if they are identical. Second, an "overlap" measure, where two sets are considered to partially match if they share some subset. And third, a "weighted overlap", which takes into account the relative semantic distance between different classes in assessing the match between two sets of categories. The results in Table 1 show the system to work reasonably well, although not at human level.

		κ	Actual Agreement	Chance Agreement	s_κ
Set equality					
	Human-Human	0.84	0.84	0.034	0.014
	System-Human	0.65	0.66	0.025	0.018
Overlap					
	Human-Human	0.87	0.88	0.040	0.012
	System-Human	0.73	0.74	0.033	0.016
Weighted overlap					
	Human-Human	0.92	0.94	0.30	0.0087
	System-Human	0.81	0.87	0.30	0.012

Table 1. Agreement between the system and human raters.

Regarding the hypothesis question of whether replacing menu-based justifications with natural language justifications helps students have a better understanding of geometry, we do not have a definitive answer yet. Some experimental results based on the same study seem to show (Aleven et al. 2004) that while students' ability to express their knowledge was improved considerably, students' performance on actual problem solving was not affected significantly. There are a number of possible causes for that, so further studies are needed.

5 Conclusions

We present a natural language understanding system that combines unification-based syntactic processing with logic-based semantics. The system is used in conjunction with a Geometry Cognitive Tutor to help students better understand geometry. The approach we take allows for an elegant solution to the problem of determining equivalence between various ways to express the same meaning. Study results show that the system works reasonably well on classifying students' explanations on a grid of about 200 fine-grained categories, although there is space for further improvement. One particular problem is robustness in the face of ungrammaticality. Also the question of whether natural language explanations improve students' understanding of geometry still waits for a definitive answer.

Acknowledgements

This work was supported by NSF ITR/IPE, NSF grant No. EIA-0113864, "Tutoring explanation and discovery learning: Achieving deep understanding through tutorial dialog."

References

Vincent Aleven and Kenneth R. Koedinger 2002. An Effective Meta-cognitive Strategy: Learning by Doing and Explaining with a Computer-Based Cognitive Tutor. *Cognitive Science*, 26(2), 147-179.

Vincent Aleven, Amy Ogan, Octav Popescu, Cristen Torrey, Kenneth R. Koedinger 2004. Evaluating the Effectiveness of a Tutorial Dialogue System for Self-Explanation. In Proceedings of the 7th International Conference on Intelligent Tutoring Systems, Macelo, Brasil.

John R. Anderson, Albert T. Corbett, Kenneth R. Koedinger, and Ray Pelletier 1995. Cognitive tutors: Lessons learned. In *The Journal of the Learning Sciences*, 4:167-207.

John R. Anderson and Christian Lebiere 1998. *The Atomic Components of Thought.* Hillsdale, NJ: Erlbaum.

Jacob Cohen 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20, 37-46.

Kenneth R. Koedinger, John J. Anderson, William H. Hadley and Mary A. Mark 1997. Intelligent tutoring goes to school in the big city. In *International Journal of Artificial Intelligence in Education*, 8:30-43.

Robert MacGregor 1991. Using a description classifier to enhance deductive inference. In Proceedings of the Seventh IEEE Conference on AI Applications, 141-147, Miami, FL.

Carolyn Penstein Rosé and Alon Lavie 1999. LCFlex: An efficient robust left-corner parser, User's manual, University of Pittsburgh.