Logic-Based Natural Language Understanding for Cognitive Tutors

OCTAV POPESCU, VINCENT ALEVEN, KEN KOEDINGER
Human Computer Interaction Institute
Carnegie Mellon University

Abstract

High-precision 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 Cognitive Tutor. The system combines unification-based syntactic processing with description logics based semantics to achieve the necessary accuracy level. The paper describes the compositional process of building the syntactic structure and the semantic interpretation of NLU explanations. It also discusses results of an evaluation of classification performance on data collected during a classroom study.

1 Explanations in Geometry Tutoring

The Geometry Cognitive Tutor assists students in learning by doing as they work on geometry problems on the computer. Currently the Geometry Cognitive Tutor is in regular use (two days per week) in about 500 schools around the US.

In previous evaluation studies Koedinger et al. (1997) have shown that the tutors are successful in raising high school students' test scores in both algebra and geometry. However, there is still a considerable gap between the effectiveness of current cognitive tutor programs and the best human tutors (Bloom, 1984).

Cognitive Tutors pose problems to students and check their solutions to these problems step by step. They can also provide context-sensitive hints at each step in solving the problem, as needed. However, prior Cognitive Tutors do not ask students to explain or justify their answers in their words. On the other hand human tutors often engage students in thinking about the reasons behind the solution steps. Such "self-explanation" has the potential to improve students' understanding of the domain, resulting in knowledge that generalizes better to new situations. This difference might also be the main explanation beneath the gap mentioned above. To verify this hypothesis, the next generation of intelligent cognitive tutors needs to be able to carry tutoring dialogs with students at the explanation level.

Some of the current intelligent tutoring systems, like Autotutor (Wiemer-Hastings et al, 1999), Circsim-Tutor (Glass, 2000), and Atlas/Andes (Rosé et al, 2001), do have natural language processing capabilities. However, these systems rely on either statistical processing of language, identifying keywords in language, or some level of syntactic analysis. None of these approaches seem to achieve the degree of precision in understanding needed in a highly formalized domain such as geometry tutoring.

One of the main problems that the Geometry Tutor faces is to determine with accuracy the semantic content of students' utterances. Natural language allows for many different ways to express the same meaning, all of which have to be recognized by the system as being semantically equivalent. The determination of semantic equivalence 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 (the Angle Addition Theorem).

```
An angle formed by adjacent angles is equal to the sum of these
angles.
The measures of two adjacent angles sum up to the measure of
the angle that the 2 angles form.
An angle's measure is equal to the sum of the two adjacent
angles that compose it.
The sum of two adjacent angles equals the larger angle the two
are forming.
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 made up of two adjacent angles is equal
to the sum 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.
An angle is equal to the sum of its adjacent parts.
Two adjacent angles, when added together, will be equal to the
whole angle.
The sum of the measures of adjacent angles equals the measure
of the angle formed by them.
Two adjacent angles added together make the measure of the
larger angle.
```

The process also has 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.

The paper presents an NLU system we have built to test this hypothesis. The next section describes the overall architecture of the system, and illustrates the main interpretation mechanism. Section 3 discusses the results of an evaluation we completed based on data from a recent classroom study.

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 correction, it passes it to a chart parser.

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 Carnegie Mellon University and University of Pittsburgh (Rosé and Lavie, 2001). The parser takes words of a sentence one by one and combines them in larger phrase structures, according to

rules in the unification grammar. It then calls the *feature structure unifier* in order to process restrictions attached to grammar rules and build *feature structures (FS)* 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.

Some of the restrictions in the grammar are directives to the *description logics system*, currently Loom (MacGregor, 1991). The logic 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.
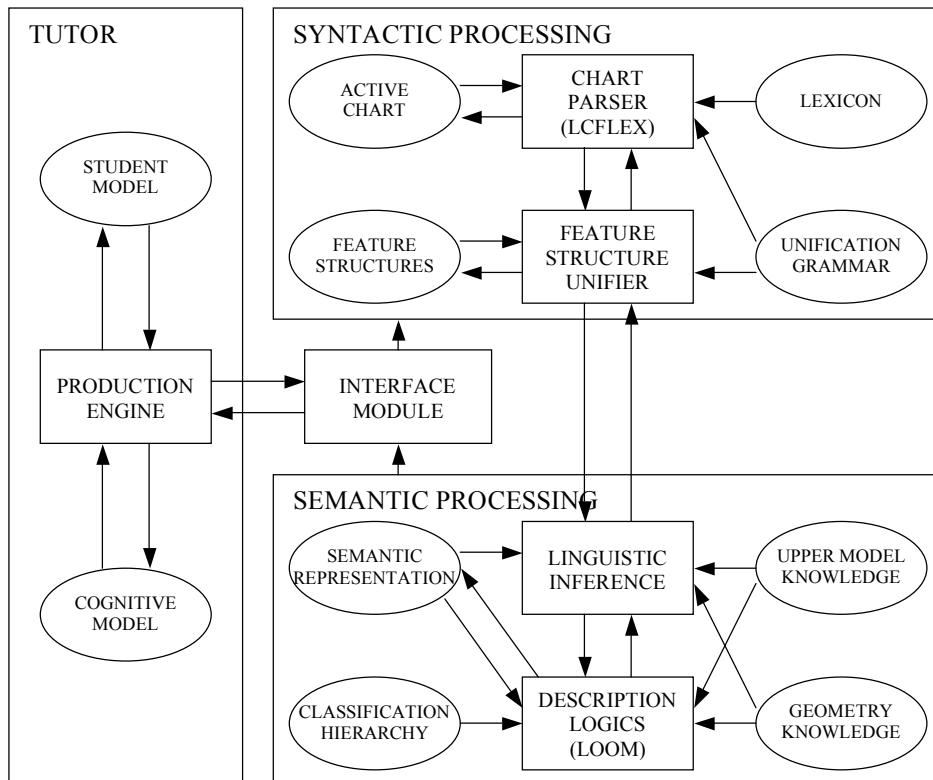

Figure 1. System Architecture

The logic system then uses a classifier to evaluate the semantic representation against a *classification hierarchy* of valid logic definitions of full geometry theorems, as well as of many incomplete ways to state them. The results of the classification are passed back to the tutor by the interface module. Based on that, the tutor generates an appropriate feedback to the student's input (Aleven et al, 2003).

2.1 Linguistic Inference

The linguistic inference module creates the interface between the feature structure unifier and the description logics module. In the process, it also performs two additional inference processes that rely on a combination of linguistic context and domain knowledge: reference resolution and metonymy resolution.

The interaction between syntax and semantics is mediated by semantic restriction statements attached to rules in the unification grammar. These statements ensure that the right semantic representation is built compositionally from representations for right-hand side components, and that a reference to the built representation is kept in the feature structure. The statements are interpreted by the feature structure unifier as special constructs, which require Lisp function calls.

At each step in the parsing process the semantic representation for a new phrase (the left-hand side non-terminal) is generated through one of four different methods:

- It is newly created by calling function `create-semantics`, in case the component is one of the lexical elements. A special case is when it is created as a new measure, out of components that are a numeric value and possibly a measure unit.

- It is created through the combination (or merging) of the representations of two components through a call to `combine-semantics`.

- It is created by connecting two components through a semantic relation, through a call to `connect-semantics`, when one component is a functional role of the other.

- It is combined in a sequence of representations through a call to `collect-semantics`, when the component representations are not directly logically connected, like in some multi-clause sentences.

Specific information about concepts and relations to be used in the semantic interpretation process are derived from lexical entries for the corresponding words.

2.2 Compositional Building of Semantic Representations

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 simplified ontology of concepts necessary for building its representation (part of the domain's ontology) is shown in Figure 2. The ontology is divided into a domain-independent UPPER MODEL KB and a domain-specific GEOMETRY KB.

Based on these concepts and the rules of the grammar, the system takes the subject of the example above ("The measure of a right angle"), and after a number of steps creates

for it an arc labeled `<NP>` in the chart, whose feature structure's semantics links to the representation illustrated on the left in Figure 3. The system also takes the verb phrase ("is 90 degrees") and creates another arc labeled `<VP>`, whose semantics representation is illustrated on the right in Figure 3.
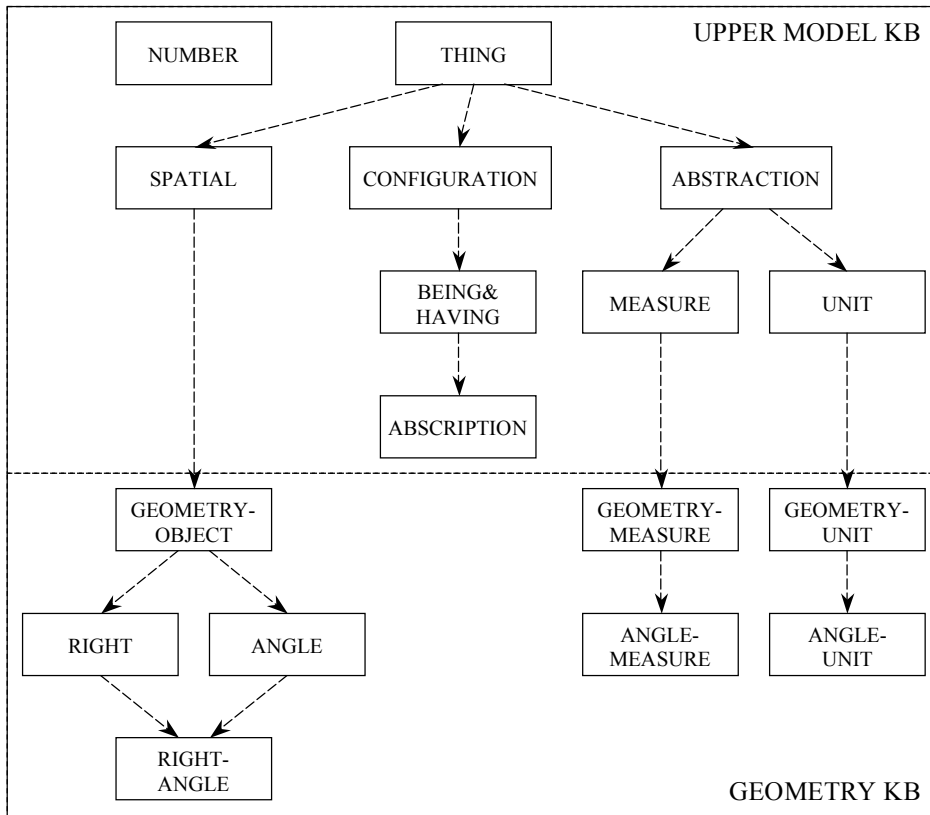

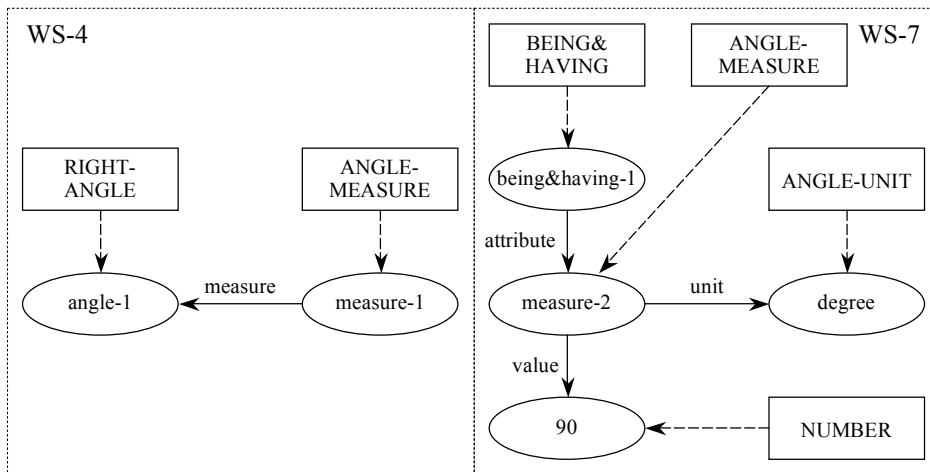Figure 2. Example of partial upper model and geometry ontology

Figure 3. Example of Semantic Representation

In the last step, the parser applies the grammar rule for clauses, given below in simplified form. The grammar rule has a context-free part and a set of unification equations. The parser first uses the context-free part to identify adjacent arcs in the chart that can be combined together in a larger phrase structure. Specifically in this case it finds the two arcs for the subject and the verb phrase of the sentence labeled `<NP>` and `<VP>` respectively, and creates a new arc for the entire sentence, labeled `<Cl>`.

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

Then the parser calls the features structure unifier to process the unification equations and build a feature structure for `<Cl>`, labeled `x0`. The first two equations create the FS `x0` from the `<VP>`'s FS `x2`, and attach to it the FS for `<NP>` on a feature called `subject`. The third equation then creates a new `semantics` feature for `<Cl>`, by calling `connect-semantics` on the semantic representations of the two components. The call to `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. The `connect-semantics` function is one of the four generation methods in the linguistic inference module.

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 shown in Figure 4.
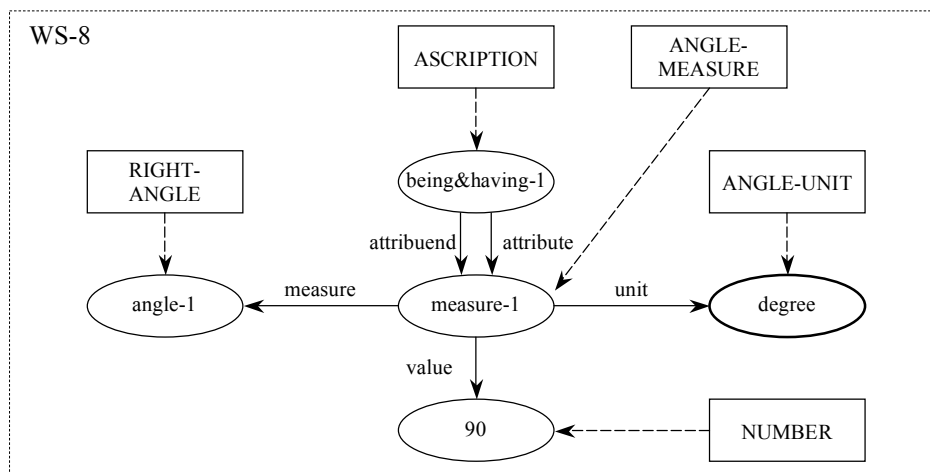

Figure 4. Resulting Semantic Representation

This way the system interleaves the syntactic parsing with the semantic interpretation and the domain-specific reasoning, in an attempt to use all possible constraints as soon as they are available.

This structure is then classified against a hierarchy of concept definitions representing classes of possible explanations, defined in Loom's terminological language. A few of

them related to the discussed example are shown in Figure 5, together with an example sentence that falls under each concept.
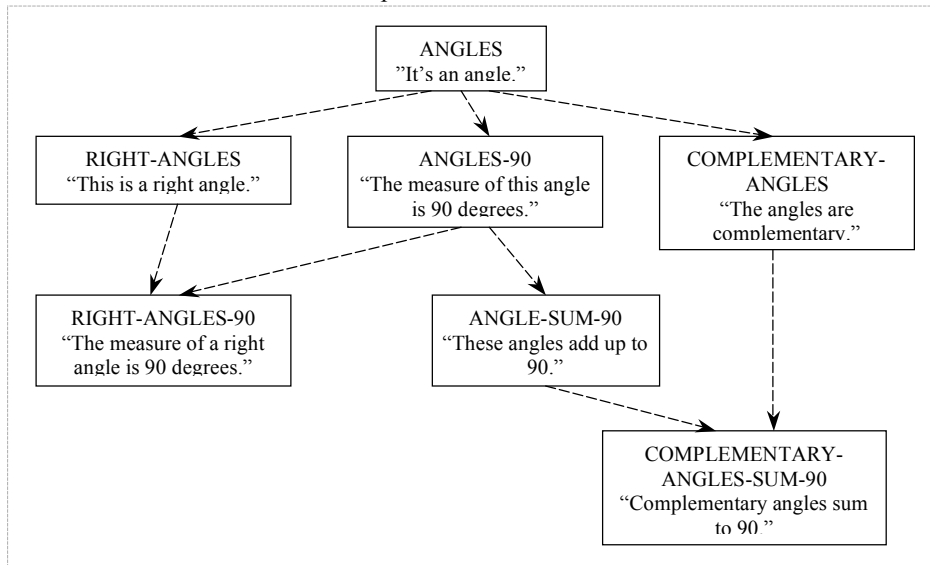

Figure 5. Partial Classification Hierarchy

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.

3.1   *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 and an inference rule that identify the container relation at the assertion level, and percolate it down to involved objects.

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

```
(: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 set of concepts and rules that will identify cases of "equal in some measurable quantity" and will generate a structure with the meaning of "equal quantity".

```
(defrelation equal-to
  :is-primitive (:and relation (:domain Thing) (:range Thing)
  :characteristics :symmetric)
(defconcept Equal-in
  :is (:and (:at-least 1 equal-to) (:some location Measure)))
(implies Equal-in
         (:same-as (:compose equal-to equal-to) location))
```

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`.

3.2   *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 Finite-Object)
                     (:range Finite-Object)))
```

Moreover, we can add an inference 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)))
(implies Equal-Measure
         (:relates congruent-to measure-of
                   (:compose equal-to measure-of)))
```

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))
```

3.3   Structural Ambiguity

Structural 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)
```

3.4   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 Evaluation of NLU Performance

We evaluated the NLU performance by measuring the accuracy of the system's classification of student explanations with respect to its set of explanation categories. As in a previous study (Aleven et al, 2002), we used the κ chance-corrected statistic (Cohen, 1960) as an agreement measure among the system and two of the authors. For two raters, κ is defined by the formula:

$$K = \frac{p_0 - p_c}{1 - p_c} = 1 - \frac{1 - p_0}{1 - p_c} = 1 - \frac{q_0}{q_c}$$

Where: p_0 = the proportion of units in which the raters agreed
p_c = the proportion of units for which agreement is expected by chance.
q_0 = the proportion of units in which the raters disagreed
q_c = the proportion of units for which disagreement is expected by chance.

The original κ measure assumes each example in the test set is assigned a single category label. However, in our system it is often the case that a sentence can be labeled with a set of categories that complement each other to cover the entire sentence meaning. Thus we need a way to measure how much two such sets of labels agree. Therefore we used "weighted κ" (Cohen, 1968), a version of the κ statistic that takes into account the degree of difference between labels (or in our case, label sets). Under such a measure a weight is assigned to each pair of sets of labels in the n x n matrix of joint label assignment frequencies for each pair of raters.

In case of weighted κ, the disagreement proportions above are computed based on the joint matrix of label assignments for each pair of raters as:

$$q_0 = \frac{\sum_{i,j} v_{ij} p_{0ij}}{v_{max}} \qquad q_c = \frac{\sum_{i,j} v_{ij} p_{cij}}{v_{max}}$$

Where: v_{ij} = the disagreement weight of cell ij in the n x n table
v_{max} = maximum disagreement weight (taken 1 in our case)
p_{0ij} = proportion of joint labels observed in cell ij
p_{cij} = proportion of joint labels in cell ij expected by chance

We used three different measures for disagreement between two different sets of labels given by two raters to the same sentence. First we used a "set equality" measure. Under this measure two sets of labels were considered to agree only if they are identical. Thus, we took $v_{ij} = 0$ if the two sets are identical and $v_{ij} = 1$ if they differ in any element.

However, this measure puts too high a penalty on small differences between two sets of labels, and thus does not accurately reflect the difference in meaning between the two sets. Then, as the second measure we used an "overlap" measure, where the degree of disagreement between two sets of labels was computed as the ratio of the number of unshared labels versus the total number of labels, using the formula:

$$v_{ij} = \frac{1}{2}\left(\frac{diff\,(S_i, S_j)}{card(S_i)} + \frac{diff\,(S_j, S_i)}{card(S_j)} \right)$$

Where: S_i and S_j are the two sets of labels
$diff(S_i, S_j)$ = the number of labels of S_i which is not shared with S_j
$card(S_i)$ = the total number of labels of S_i

While this measure is able to take into account the degree to which two sets have labels in common, it still has a problem. It considers all labels to be disjoint and equally different from one another. However in our case this is not true. Since we have a taxonomy of categories with various conceptual content, some pairs of categories are more similar than others. And most of them are not totally disjoint, they have part of the meaning content in common. Then as a third measure we used a "weighted overlap" measure that takes into account the semantic dissimilarity between individual labels. In this case the weights become:

$$v_{ij} = \frac{1}{2}\left(\frac{\displaystyle\sum_{l_k \in S_i} dist(l_k, S_j)}{card(S_i)} + \frac{\displaystyle\sum_{l_k \in S_j} dist(l_k, S_i)}{card(S_j)} \right)$$

Where: S_i and S_j are the two sets of labels
l_k = each label of each set respectively
$dist(l_k, S_i)$ = the minimum distance between label l_k and some label in S_i

The dissimilarity between two labels is approximated by the distance measure between the corresponding categories in the class taxonomy. As seen in the formula above, the dissimilarity of one set of labels with respect to the other is taken as the average of the minimum distance between each label in the first set and some label in the second set:

$$dist(l_i, S_j) = \min_{l_k \in S_j} dist(l_i, l_k)$$

The semantic distance between two categories in the taxonomy approximates the amount of information in the categories' definitions by their depth in the taxonomy. The formula we used is:

$$dist(l_i, l_j) = \min_{a_k} \frac{bdist(a_k, l_i) + bdist(a_k, l_j)}{bdist(top, a_k) + bdist(a_k, l_i) + bdist(a_k, l_j)}$$

Where: l_i, l_j = the two categories
a_k = most specific common ancestors of the two categories
top = the top category in the taxonomy
$bdist(a,l)$ = branch distance between ancestor a and category l

The branch distance between an ancestor category and a subconcept category is computed as the maximum distance between them on all possible branches that connect them.

In the evaluation we used a subset of 700 explanations randomly chosen from a corpus of 2700 explanations collected during a pilot study conducted in the Spring of 2003. During the study, the Geometry Cognitive Tutor was used for about a week in a suburban junior high school in the Pittsburgh area, as part of a 9th-grade Integrated Mathematics II course. The category taxonomy consists of 198 categories. For each of the three agreement measures, we computed the κs between two human raters (the first two authors), as well as the average of the κs between the system and each of the two human raters.

Table 1. Average pair-wise inter-rater agreement between human raters and average pair-wise agreement between the system and each human rater.

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

The results can be seen in Table 1 above. Compared to a previous evaluation (Aleven et al, 2002), the results for the weighted overlap method show a decrease of the human-human agreement by about .02, and an increase in the average system-human agreement by about .03. So the gap between humans and the system decreased from .16 to .11.

The lower human-human agreement could mean that the classification task was harder this time around. That could happen either because the corpus was more difficult or because of the increased number of classes the raters had to choose from (about 200, versus about 150 in the previous evaluation). The higher system-human agreement probably reflects the improved performance of the system.

A closer look at how agreement was distributed among the raters and the system reveals the picture in Figure 6. There are 436 cases (about 62%) where all raters agreed, 202 cases (154+48, about 29%) where two of the raters agreed and the third one did not, and 62 other cases (about 9%) where none of the raters agreed.

We performed a case by case examination of the 154 cases where the human raters agreed and the system did not, trying to reveal the cause for the system failure to generate the same set of labels as the humans. The examination showed there is no single big problem with the system, but rather a large collection of small problems, each of them affecting a couple of cases. We could group these problems into several categories, listed below with examples:

- Robustness in the face of ungrammatical input:

```
The measure of an angle formed by two adjacent angles is equal
to the sum measures of the of the adjacent angles.
2 angles that are next to each other have are the same.
```

- Syntactic or semantic coverage of new ways of expression:

```
The angles sum up to because they are on a line and adjacent.
When a ray is drawn off part of a line, the angles on one side
add up to 180.
Angle blu and angle ule are complementary angles so when added
together their sum should equal 90 degrees.
```

- Problems with implicit references or ellipses:

```
Subtract the measure of the vertical angle by the measure of
the interior angle formed.
Three adjacent angles make up the larger angle so you must add
them to find the measure.
```

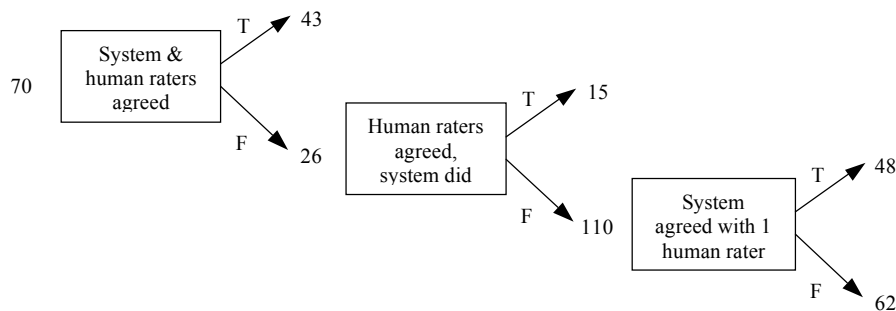- Bugs in various system components.

Figure 6. Agreement distribution among raters

The study confirmed two things. First, that the system does an adequate job on capturing the meaning of students' explanations in an application that requires high accuracy. And second, that there are still an important number of improvements that can be made to the system to increase its performance. One especially promising solution that we plan to develop soon involves connecting semantic fragments of a partial parse structure through paths found in a limited directed search through the semantic space of the concepts involved.

5 Conclusions

We presented an approach to NLU that relies on a combination of unification-based syntax and logic-based semantics. The approach requires a significant development effort, which raises the question of whether it is worth it. There are a few arguments for it. First, our approach uses domain-based knowledge similarly to how that people use it in process of language understanding. The result is that it is able to deal with the problem of semantic equivalence in a more general and more principled way. The use of domain knowledge also results in a more natural behavior. Even if there is still a significant gap between its performance and that of a human, it fails mostly on cases that are perceived as incorrect/ vague/elliptic/difficult by humans too.

Second, most of the development effort is domain-independent, and thus can be reused in other domains. We performed a study recently, where we implemented the same technology in algebra, to process students' explanations about the equation solving process. We were able to reuse all the basic mechanisms, plus the entire grammar, with some adjustments, and the upper model of the ontology. We only needed to implement the domain-specific components: a new lexicon, a new domain-dependent ontology, and a new classification hierarchy. The overall development time for the new application was about one month.

References

Aleven, V.; Popescu, O.; and Koedinger. K.R. (2002) Pilot-testing a tutorial dialogue system that supports self-explanation. In A. Cerri, G. Gouarderes, and F. Paraguacu (eds.) Proceedings of Sixth International Conference on Intelligent Tutoring Systems, ITS2002. Berlin, Germany: Springer Verlag, 246-255.

Aleven, V; Popescu, O.; Ogan, A.; and Koedinger, K.R. (2003) A formative classroom evaluation of a tutorial dialog system that supports self-explanation. Proceedings of AI in Education, Sydney, Australia.

Bloom, B.S. (1984) The 2 sigma problem: the search for methods of group instruction as effective as one on one tutoring. *Educational Researcher* 3:4-16.

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

Cohen, J. (1968) Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit. In *Psychological Bulletin* 70;213-220.

Glass, M. (2000) Processing Language Input in the CIRCSIM-Tutor Intelligent Tutoring System. In Proceedings of AAAI 2000 Fall Symposium on Building Dialogue Systems for Tutorial Applications, Menlo Park, CA: AAAI Press, 74-79.

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

Rosé, C.P.; Jordan, P.W.; Ringenberg, M.; Siler, S.; VanLehn, K.; and Weinstein, A. (2001) Interactive Conceptual Tutoring in Atlas-Andes, Proceedings of AI in Education.

Rosé, C.P.; and Lavie, A. (2001) Balancing robustness and efficiency in unification-augmented context-free parsers for large practical applications, robustness in language and speech technology. In G. van Noord and J.C. Junqua (eds.), *Robustness in Language and Speech Technology*, Kluwer Academic Press.

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

Pollard, C. and Sag, I.A. (1994) *Head-Driven Phrase Structure Grammar.* The University of Chicago Press.

Wiemer-Hastings, P.; Wiemer-Hastings, K.; and Graesser, A.C. (1999) Improving an intelligent tutor's comprehension of students with latent semantic analysis. In S.P. Lajoie and M. Vivet (eds.), *Artificial Intelligence in Education, Open Learning Environments: New Computational Technologies to Support Learning, Exploration, and Collaboration*. Proceedings of AIED-99. Amsterdam, Netherlands: IOS Press, 535-542.