From Syntax to Meaning in Natural Language Processing

Alexander G. Hauptmann
School of Computer Science, Carnegie Mellon University
Pittsburgh, PA 15213-3890
alex@cs.cmu.edu

Abstract

The development of larger scale natural language systems has been hampered by the need to manually create mappings from syntactic structures into meaning representations. A new approach to semantic interpretation is proposed, which uses partial syntactic structures as the main unit of abstraction for interpretation rules. This approach can work for a variety of syntactic representations corresponding to directed acyclic graphs. It is designed to map into meaning representations based on frame hierarchies with inheritance. We define semantic interpretation rules in a compact format. The format is suitable for automatic rule extension or rule generalization, when existing hand-coded rules do not cover the current input. Furthermore, automatic discovery of semantic interpretation rules from input/output examples is made possible by this new rule format. The principles of the approach are validated in a comparison to other methods on a separately developed domain.

Instead of relying purely on painstaking human effort, this paper combines human expertise with computer learning strategies to successfully overcome the bottleneck of semantic interpretation.

Semantic Interpretation

An important step in the language understanding process is constructing a representation of the meaning of a sentence, given the syntactic structure. Mapping from syntactic structures into a meaning representation is referred to as **semantic interpretation** or semantic mapping. To do this, we need a set of interpretation rules, which tell us how to create a meaning representation from the syntax representation.

Creating semantic interpretations can be difficult for many reasons. Consider, for example, a machine translation system with N languages and M different domains. Each domain describes a distinct world of conversational topics and concepts. While we only need to write one syntactic grammar to understand each language and only one frame representation for each domain, we must write $N * M$ different sets of semantic interpretation rules to interpret and map from each syntactic representation into

```
((FRAME *MOVE)
 (FORM QUES)
 (AGENT   ((FRAME *HUMAN)
           (PRO +)
           (NUMBER SING)
           (PERSON 2)))
 (OBJECT  ((FRAME *BODY-PART)
           (NAME *THUMB)
           (POSSESSIVE ((FRAME *HUMAN)
                        (NUMBER SG)
                        (PERSON 2)
                        (PRO +))))))
```

Figure 1: The semantic meaning representation for the sentence *"Can you move your thumb"*

each domain representation. In other natural language systems, it may often be possible to incorporate an existing syntactic grammar and a frame representation developed by others for the domain, but the semantic interpretation information must always be constructed anew. Compositional semantics approaches [Montague, 1974, Pollack and Pereira, 1988] rely on a direct functional conversion of syntactic elements to semantic representation. Charniak [Charniak, 1981] discusses the "case-slot identity theory" and its shortcomings. Only in trivial and artificially constructed domains does the syntactic representation and the meaning representation coincide isomorphically. E.g. in multi-lingual machine translation, it is desirable to represent the meaning of the sentence *My birthday is June 12, 1959* identical to the sentence *I was born on June 12, 1959*, violating the case-slot identity theory.

So-called semantic grammars combine both the syntactic knowledge as well as the meaning representation [Brown and Burton, 1975, Hendrix, 1977]. This is a difficult task since the syntactic grammar and semantic interpretation rules have to be written all in one step.

As different syntactic formalisms are proposed, new semantic mappings must be created for each domain. In addition, the process of continually constructing new semantic interpretation rule sets requires an expert who is both familiar with the intricacies of the syntactic grammar as well

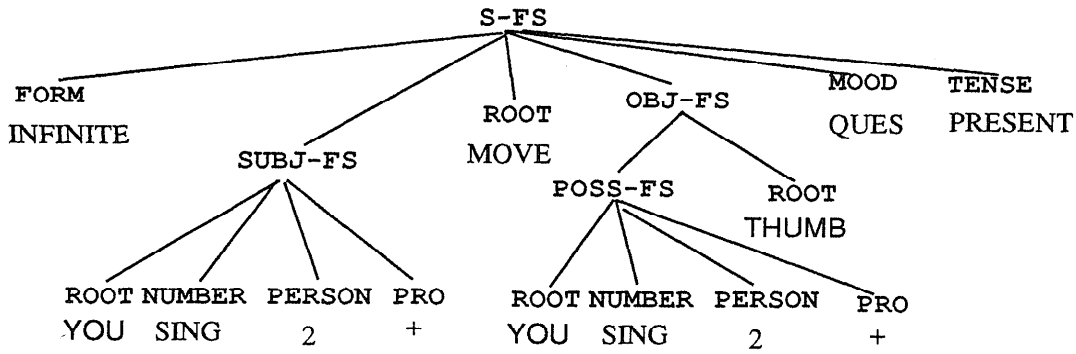

Figure 2: The LFG-style syntactic structure for the sentence *"Can you move your thumb"*

as with the frame-based knowledge representation for the domain.

Related Research

Making semantic interpretation knowledge explicit. For a good solution to the creation of meaning representations, it seems reasonable to extend the ideas of the RUS approach [Bobrow, 1978]. The RUS system is built on an ATN syntactic parser and the KL-ONE [Brachman, 1979] semantic representation formalism. Each time a main verb or complete noun phrase is parsed syntactically, the resulting structure is handed over to the semantic interpretation module to check for semantic wellformedness. In the original RUS system, this merely implied activating the concept nodes in the KL-ONE network for individual words and any links between activated concepts.

An extension of the original RUS system for the mapping from syntactic structures into semantic representations is discussed in [Bobrow and Webber, 1980, Sontheimer *et al.*, 1984]. Here the KL-ONE concepts have associated head words. This allows a word to instantiate a concept. In addition, translation rules are attached to each role in the concept, which determine how role-filler concepts are connected to the parent concept, based on syntactic structure evaluations. Rewrite rules for transforming surface structures into a deeper meaning representation were also discussed by [Palmer, 1983] and [Bayer *et al.*, 1985]. Mapping rules for these approaches usually consist of arbitrary lisp code and are difficult to write and debug, since the approach provides only minimal structure, as demonstrated in the critique of EUFID [Templeton and Burger, 1983].

The Center for Machine Translation (CMT) at Carnegie Mellon has developed another system where the semantic interpretation information is explicitly represented [Center for Machine Translation, 1989, Tomita and Carbonell, 1987a, Carbonell and Tomita, 1988]. In that system, mappings into meaning can be arbitrarily complex. The mapping information is represented in the same notation as LFG-style grammar rules. However, the representation of this semantic mapping information requires a skilled linguistic knowledge engineer who is familiar with both the domain representation as well as the grammatical mechanisms used.

This paper borrows heavily from these two approaches in that the mapping knowledge is declared explicitly, but in a more rigid notation than arbitrary lisp code.

Learning of networks. Siklossy [Siklossy, 1968, Siklossy, 1972] tried to learn syntactic rules using a semantics directed approach. His system was fed with pairs of sentence strings and desired output schemata. The output schema is the semantic representation the grammar should produce upon parsing the input string. The system needs to learn the association of the two to produce a semantic grammar. Siklossy's program learned to parse Russian phrases, starting with extremely simple sentences and progressing successively to slightly more complex sentences. It relied on direct associations to do the mappings.

Anderson [Anderson, 1977, Anderson, 1981] describes language learning systems based on the association of word strings with meaning representations in the form of conceptual graphs. His LAS system tried to learn the shape of the underlying recursive phrase structure networks, which include both syntactic and semantic information. A "graph deformation constraint" essentially forces the semantic representation to be mapped linearly onto the sentence string. LAS learns the form of syntactic/semantic rules based on a direct mapping of words to concepts.

Besides the assumption of an isomorphism between sentence string and semantics, one particular problem that plagued Anderson as well as Siklossy was the large number of carefully constructed examples needed by their systems. In contrast to those systems, the automatic rule discovery method that we propose does not try to learn more than the semantic interpretation rules. The syntactic structures are already assumed to be parsed. The system presented here is able to generalize what it has learned from a small set of examples, which increases the effectiveness of the learning approach. The learning itself is similar, in that input and output pairs are used as examples to drive the acquisition of semantic mapping rules and the complete rule knowledge is built up from the set of training example pairs.

Partial Syntactic Structures

Partial syntactic trees (or similar structures in the form of directed acyclic graphs) provide the appropriate abstraction and building blocks for semantic interpretation rules. After syntactic analysis of the input sentence, partial syntactic trees trigger the application of specific semantic interpretation rules.

Partial syntactic trees are defined through operations on the full syntactic tree representation of the input sentence. An example of such a syntactic parse structure is given in Figure 2 for the sentence *"Can you move your thumb"*. Given a single unambiguous syntactic analysis of an input, a partial syntactic tree is defined through the following operations on a syntactic tree representation.

1. Any leaf of the tree may be deleted.
2. The root node of the tree may be deleted and all but one of the subtrees from the root node. The remaining child of the deleted root node then becomes the new partial tree.

These two operations may be performed as often as necessary to produce the desired partial syntactic tree. An example of such a partial syntactic structure is shaded in Figure 3.

Semantic Interpretation Rules as Tree Transformations

If one examines the notion of semantic interpretation as mapping from a syntactic tree structure into a tree-structured meaning representation, then the process of taking a syntax tree to produce a semantic frame tree is merely a tree-to-tree transformation. Each rule used in the semantic interpretation specifies the transformation of a partial (syntactic) subtree into a different partial (semantic/pragmatic) subtree. The components of the rules are therefore simple, exploiting the compositionality of the syntax as well as the semantic representation.

To produce a semantic representation of the input, the general procedure is to take a set of transformation rules, and apply them to the given syntactic structure. The interpretation rules provide the transformations from the syntactic parse tree into the semantic representation tree.

One can divide semantic interpretation rules into two distinct sets of rule actions:

1. Creating new individual semantic units (i.e. frames)
2. Combining semantic frames using specific relations.

Let us call rules which have the first kind of action to create new semantic frames "Lexical Mapping Rules", while rules which have the second type of action to combine existing semantic frames will be called "Structural Mapping Rules" after [Mitamura, 1990].

The lexical mapping rules are not of interest here. In general, they define mappings from words (or partial syntactic tree) to frames, which is most easily done as part of the lexicon. The result of these lexical mapping rules is an augmented syntactic structure which includes individual frames as nodes in the original syntactic parse structure.

When the lexical mappings have been applied, the original syntactic parse is augmented with individual frames, as in Figure 3. The structural mapping now takes place to combine these frames into one coherent relationship. The structural mapping rules are defined as 6-tuples of the form

```
<LHS, HF HF_LOCATION, EF EF_LOCATION, S>
```

where the elements contain the following information:

- *The left-hand-side LHS* – The left-hand-side of a structural mapping rule, gives the "syntactic context" in which the rule action takes place. The partial tree specified in the LHS provides the trigger for the rule. In Figure 3, the the partial tree that defines the LHS is `((ROOT MOVE)) (SUBJ))`
- *The Head Semantic Frame HF* – The structural mapping rule must identify a frame which is present in the new augmented parse structure containing the original syntactic parse together with the added frame branches. This frame must have been created by the lexical rules earlier. We will use a slot within this frame to link in another concept. In our example, the head concept is the case frame called `*MOVE`.
- *The Location of the Head Frame HF_LOCATION* – In addition to identifying a head frame by name, the structural mapping rule must also identify a location within the parse structure, where this frame should be located. We must know where in the current tree to look for the node which contains the head frame. The *MOVE frame is located in the FRAME branch at the top level of the f-structure in Figure 3, which can simply be written as `((FRAME))`.
- *The Slot S* – A slot must be identified from the head frame which will be filled by the action of the rule. The slot defines the relationship of the head frame to another frame. In the example, the slot in the *MOVE frame which we want to fill is the `AGENT` slot.
- *The Embedded Frame EF* – This part of the structural mapping rule identifies the other frame which fills the slot of the head frame. Clearly the embedded frame may not violate any slot filler restrictions present in the slot of the head frame. As shown in Figure 1, we want the `*HUMAN` frame to fill the slot `AGENT`.
- *The Location of the Embedded Frame EF_LOCATION* – Just as before, we need to specify in what part of the parse tree we are allowed to find the embedded frame that we want to be using. As it turns out, in the example in Figure 3, there are two *HUMAN frames, one is part of the possessive for *thumb*, the other is the subject. We want to specify the one that is the frame branch of the subject of the sentence from the top level with `((SUBJ ((FRAME))))`.

Now all the rule parts are defined for a rule which can create the frame fragment
`((FRAME *MOVE) (AGENT ((FRAME *HUMAN))))`
from the syntactic structure in Figure 3. Analogous rules

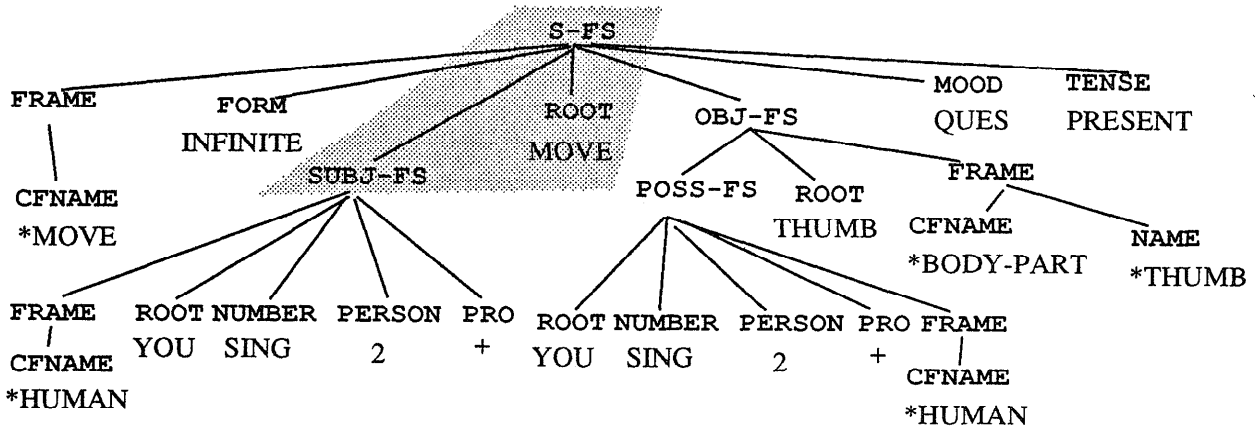

Figure 3: The syntactic structure of the sentence *"Can you move your thumb"* augmented with individual semantic frames. A partial syntactic structure is shaded at the top.

are easily constructed to create the remainder of the target meaning of Figure 1.

Generalization from Specific Mapping Rules

To facilitate the process of creating the "grammar" for semantic interpretation, it is desirable to hand-code only a minimal amount of information. In that case, it is advantageous to judiciously generalize based on a small core information about semantic interpretation. Thus we strive to create an environment where a human "teacher" merely maps a few examples and the system learns the principles of semantic interpretation from these examples with the proper guidance.

The simple composition of the semantic interpretation rules as defined above makes rule generalizations easy, so rules created in one context can be automatically modified and applied in a different situation.

For example, if instead of the sentence *"Can you move your right thumb"*, we now have the sentence *"Can you bend your arm"*, we can adapt the structural mapping rule from above by substituting for the concept *MOVE the new concept *BEND in the head frame concept (*HF*) part of the rule as well as in the context (*LHS*) of the rule. We allow this substitution because all other critical parts of the rule are identical and the embedded concept that was substituted is sufficiently similar to the original one, with similarity defined as proximity in the frame hierarchy.

In the same way, we can find other cases where we can generalize from

```
<LHS HF HF_Location EF EF_Location S>
```

to

```
<LHS HF HF_Location EF' EF'_Location S>
```

or to

```
<LHS' HF HF_Location EF EF_Location S>
```

if *EF'* and *LHS'* are similar to the original *EF* and *LHS* in the rules which we already have.

Automatic Mapping Rule Discovery

Additionally, the structure of these rules makes it possible for semantic interpretation knowledge to be learned from example pairs of syntactic structures and their corresponding semantic representations. Even though the semantic interpretation information that is learned is based on specific examples, the rule generalization process allows the specific rules to be applied to a variety of other related instances for semantic interpretation.

Initially, the system is given all syntactic parses for a particular training sentence. Each sentence is also associated with a single target meaning representation, which defines the correct semantic interpretation for that sentence. The system is also given the lexical mappings, which augment the syntactic structure with individual frames. From this it is possible to infer the necessary structural mapping rules in the format mentioned above, based on the existing example only. There may be examples for which ambiguity prevents the correct learning of mapping rules, but this has turned out to be irrelevant when given a sufficiently representative set of examples. The inferred rules, of course, are specific to the particular example sentence and its semantic representation, but the extension of specific rules through the generalization procedure described above makes it possible to extend the rules to new sentences.

This automatic mapping rules discovery method was implemented as the SIM-ARD system and tested, with the results reported below. The SIM-ARD system used several heuristics such as selecting only the minimal syntactic context for the LHS of a rule and retaining only unambiguous rules to avoid incorrect inferences. The heuristic, as implemented, also assumes a somewhat parallel hierarchy in syntax and semantics. This assumption postulates that a frame A which is below another frame B in the augmented syntactic parse structure (after lexical mapping) will not become the head frame for a rule using frame B as the embedded frame. This assumption is reminiscent of the graph deformation condition in Anderson's LAS [Anderson, 1977],

but much less restrictive, and characterizes only the relationship between syntax and meaning representation, not between sentence string and meaning representation. The heuristic is also able to deal with multiple syntactic parses, choosing the one which best fits the semantic representation of the sentence.

An Experimental Validation

To verify these claims, an experimental comparison of three different approaches to semantic interpretation was performed.

The base line system was the unmodified parsing and semantic mapping module developed by the Center for Machine Translation (CMT) at Carnegie Mellon.

The second approach, called the **SIM** system (Syntax for Interpretation of Meaning) used a set of mapping rules created by a person in the format of the semantic interpretation rules outlined above, within an integrated grammar writing environment. The rules were based on specific mappings from sentence examples in the training corpus, syntactically parsed by the CMT parser. The rules were automatically generalized during the test run.

The final approach also used the semantic interpretation rule format as outlined above. However, only a set of lexical mappings were defined for the system and all structural mapping rules were discovered automatically by the SIM-ARD system using the rule discovery/learning procedures described earlier.

The Domain

The domain that was used for these experiments was the doctor-patient domain developed at the Center for Machine Translation at Carnegie Mellon. It provided the basis for a series of knowledge-based machine translation system prototypes between 1985 and 1988. The translation system translates conversations between a doctor and a patient from English into Japanese and vice versa [Tomita and Carbonell, 1987b, Carbonell and Tomita, 1988]. Typical sentences in the domain include complaints about pain symptoms and ailments by the patient and questions as well as prescriptions for treatments by the doctor (e.g. "My left foot hurts more when I step on it.", "Apply liniment to your foot twice a day for two weeks", etc.)

The CMT system's LFG-style grammar[Bresnan and Kaplan, 1982] is described as follows[Tomita and Carbonell, 1987a, page 70]:

> "We have written a fairly comprehensive English syntactic grammar and Japanese syntactic grammar. The English grammar handles declaratives, imperatives, yes-no questions, wh-questions and other gapped constructions, auxiliary complexes and related phenomena. Additionally we built grammar rules [for] specialized constructions such as times and dates. The Japanese grammar corresponds roughly in coverage to the English grammar ..."

The domain representation is frame-based and consists of frame concepts such as *PATIENT, *HUMAN, *HAVE-SYMPTOM, and *BODY-PART. Each frame is positioned in a hierarchy. For example, the *PATIENT concept is a specific kind of *HUMAN concept. Each concept in the hierarchy inherits features from all of its more general parent concepts.

From this domain, 80 training sentences and 227 test sentences were used. Each sentence had at least one syntactic representation in addition to a well defined meaning in the domain.

A Comparison of Semantic Interpretation Approaches

Table 1 shows the results of the experimental comparison in the doctor-patient domain. The table shows that the new SIM approach, which allowed the semantic interpretation rules to be created more rapidly, was also more accurate. The SIM-ARD automatic mapping rule discovery module produced performance comparable to that of a linguist, without any human intervention. Similar results were obtained from a comparison using Japanese sentences in the same domain. The approach has also been successfully simulated on 3 further different domains with different syntactic representations.

Significance

Summing up, the achievement described in this paper is a step towards the creation of larger natural language processing system by requiring less human linguistic effort. The problem of semantic interpretation has been attacked through cooperative human-computer problem solving. The human task is reduced to a minimal amount, while taking advantage of the machine's ability to generalize from examples and to automatically discover rules. Linguistic knowledge is acquired, in the form of semantic interpretation rules from a human, through generalization of specific rules and through automatic rule discovery.

The key to success of the current evaluations for this approach are:

- The use of partial syntactic structures as the LHS of a rule.
- Compact rules to perform tree transformations of syntactic structures into frame structures.
- The ability to generalize from semantic interpretation rules which are based on specific sentences and structures.
- The automatic discovery of semantic interpretation rules from input/output specifications.

Future research will explore using the SIM approach to map into logical knowledge representations instead of frame based representations. Currently the system ignores issues of quantification and scoping. It also remains to be investigated how the system could recover from rules that were entered, learned or generalized incorrectly.

While the basic strategy of computer directed knowledge acquisition holds much promise for other aspects of natural language processing, the approach could also be applied to other systems beyond natural language processing.

English Test Corpus Results						
	CMT		SIM		SIM-ARD	
	Sentences	%	Sentences	%	Sentences	%
Unmapped/wrong meanings	77	33.9	49	21.6	64	28.2
Correct mappings	150	66.1	178	78.4	163	71.8
Total	227	100.0	227	100.0	227	100.0

Table 1: A comparison of the number of correctly mapped sentences for different semantic interpretation approaches on the 227 sentence test set in the doctor-patient domain.

The methods are general enough to apply in any circumstances where a rule-based system translates from or into well defined hierarchical structures for which rule creation is difficult.

References

Anderson, J.R. 1977. Simulation of language acquisition. In LaBerge, D. and Samuels, S.J., editors 1977, *Heuristic Use of Semantics*. L. Erlbaum Assoc., Hillsdale, NJ. 305 – 360.

Anderson, J.R. 1981. A theory of language acquisition based on general learning principles. In *Proc. Seventh Int. Jt. Conf. on Artificial Intelligence*. 97 – 103.

Bayer, S.; Joseph, L.; and Kalish, C. 1985. Grammatical relations as the basis for natural language processing and text understanding. In *Proc. Ninth Int. Jt. Conf. on Artificial Intelligence*. 788 – 790.

Bobrow, R. J. and Webber, B. 1980. Knowledge representation for syntactic/semantic processing. In *Proc. National Conference of the American Association for Artificial Intelligence*, Stanford University. 316 – 323.

Bobrow, R. J. 1978. The RUS system. BBN Report 3878, Bolt, Beranek, and Newman, Inc., Cambridge, Mass.

Brachman, R.J. 1979. Taxonomy, descriptions and individuals in natural language understanding. In *COLING-79*, University of California, San Diego. 33 – 37.

Bresnan, J. and Kaplan, R. 1982. Lexical functional grammar: A formal system for grammatical representation. In *The mental representation of grammatical relations*. MIT Press, Cambridge, MA. 173 – 281.

Brown, J. S. and Burton, R. R. 1975. Multiple representations of knowledge for tutorial reasoning. In Bobrow, D. G. and Collins, A., editors 1975, *Representation and Understanding*. Academic Press, New York. 311–349.

Carbonell, J. G. and Tomita, M. 1988. Multilingual machine translation, a knowledge-based approach. In *Defense Applications of Artificial Intelligence*. Lexington Books, D.C. Heath and Co.

Center for Machine Translation, 1989. KBMT-89 project report. Technical report, Center for Machine Translation, Carnegie Mellon University.

Charniak, E. 1981. The case-slot identity theory. *Cognitive Science* 5:285 – 292.

Hendrix, G. G. 1977. Human engineering for applied natural language processing. In *Proc. Fifth Int. Jt. Conf. on Artificial Intelligence*. MIT. 183 – 191.

Mitamura, T. 1990. The hierarchical organization of predicate frames for interpretive mapping in natural language processing. Technical Report CMU-CMT-90-117, Center for Machine Translation, Carnegie Mellon University. Ph.D. Dissertation, Linguistics Department, University of Pittsburgh.

Montague, R. 1974. *Formal Philosophy*. Yale University Press, New Haven, CT.

Palmer, M.S. 1983. Inference-driven semantic analysis. In *Proc. National Conference of the American Association for Artificial Intelligence*. 310 – 313.

Pollack, M.E. and Pereira, F.C.N. 1988. An integrated framework for semantic and pragmatic interpretation. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics*. Also available as SRI International, Technical Note 455, January 1989.

Siklossy, L. 1968. *Natural Language Learning by Computer*. Ph.D. Dissertation, Carnegie Mellon University, Department of Computer Science.

Siklossy, L. 1972. *Natural Language Learning by Computer*. Prentice-Hall, Englewood Cliffs, NJ. 288 – 330.

Sontheimer, N.K.; Weischedel, R.M.; and Bobrow, R.J. 1984. Semantic interpretation using KL-ONE. In *Proceedings of COLING-84*, Stanford University.

Templeton, M. and Burger, J. 1983. Problems in natural language interface to DBMS with examples from EUFID. In *Proceedings of the Conference on Applied Natural Language Processing*, Santa Monica, CA. Association for Computational Linguistics. 3 – 16.

Tomita, M. and Carbonell, J. G. 1987a. The universal parser architecture for knowledge-based machine translation. In Habermann, A. N., editor 1987a, *Computer Science Research Review*. Carnegie Mellon University, Pittsburgh, PA. 63 – 73.

Tomita, M. and Carbonell, J. G. 1987b. The universal parser architecture for knowledge-based machine translation. In *Proceedings of IJCAI-87*, Milan, Italy.