Applying a Semantic Interpreter to a Knowledge Extraction Task

Fernando Gomez [1], Carlos Segami [2]

[1] Dept. of Computer Science, School of EECS
University of Central Florida, Orlando, Fl 32816

[2] Dept. of Mathematics and Computer Science
Barry University, Miami Shores, FL 33161

Abstract. A system that extracts knowledge from encyclopedic texts is presented. The knowledge extraction component is based on a semantic interpreter of English based on an enhanced WordNet. The input to the knowledge extraction component is the output of the semantic interpreter. The extraction task was chosen in order to test the semantic interpreter. The following aspects are described: the definition of verb predicates and semantic roles, the organization of the inferences, an evaluation of the system, and a session with the system.

1 Introduction

There could be little doubt that a knowledge extraction component (KE) should be based on the output of a semantic interpreter. The more general the semantic interpreter the easier it should be to build different knowledge extraction tasks for different domains. This paper describes a KE that is fully based on the output of a semantic interpreter. It is also shown that the inferences of the KE are organized on the verb predicates used by the semantic interpreter to assign meaning to the grammatical relations of the sentence. Moreover, the KE uses the same ontology as that of the semantic interpreter. Because the KE component is grounded on the semantic interpretation algorithm and shares the same ontology, the construction of different KE extraction tasks reduces to building some inferences in the predicates used by the semantic interpreter. Incompatibilities between ontologies used by diverse components of the system do not exist. Furthermore, the KE designer does not have to be concerned with defining ontological categories, because these have been built for him/her in the semantic interpreter. This paper is organized as follows. Section 2) explains the semantic interpreter briefly; Section 3) describes the knowledge extraction task from the *The World Book Encyclopedia* (World Book, Inc., Chicago.1987). Sections 4 and 5 explain the organization of the inferences in the KE. And sections 6, 7, 8 and 9 provide the testing, a sample session, related work, and conclusions, respectively.

Gomez F. and Segami C. (2005).
Applying a Semantic Interpreter to a Knowledge Extraction Task.
In *Proceedings of the 2nd International Workshop on Natural Language Understanding and Cognitive Science*, pages 100-109
DOI: 10.5220/0002569201000109

2 The Semantic Interpreter

We have defined verb predicates for WordNet verb classes [2], which have undergone considerable reorganization and redefinition following the criteria imposed by the interpretation algorithm. The WordNet upper-level ontology for nouns [11] has also undergone reorganization and redefinition [5] based on the feedback that we have obtained from the semantic interpreter.

The selectional restrictions in the predicates are linked to the WordNet ontology for nouns. The predicates form a hierarchy in which semantic roles and inferences are inherited by subpredicates from their superpredicates. For instance, the predicate *graduate-from* has the following hierarchy:

```
GRADUATE-FROM → RECEIVE-AN-ACADEMIC-DEGREE → GET-AWARD → GET →
        TRANSFER-OF-POSSESSION → ACTION
```

where the arrow represents the is-a relation.

The syntax for the semantic roles in the predicates is:

```
(role (<slr>)(<grs>) (<slr>)(<grs>) ... (<slr>)(<grs>))
```

Where <*slr*> stands for any number of selectional restrictions, and <*grs*> for any number of grammatical relations. The grammatical relations for PPs are represented by writing "prep" followed by the prepositions that realize the semantic role, e.g. (prep about on ...). The list of selectional restrictions is a preference list [15]. Entry *slri* is preferred over entry *slri+1*. Thus, if entry *slri* subsumes the ontological category of the head noun of the grammatical relation in the sentence, the entry *slri+1* is not tried [3]. However, the list of grammatical relations is an unordered list. The entry for the predicate *graduate-from* is:

```
[GRADUATE-FROM (IS-A (RECEIVE-AN-ACADEMIC-DEGREE)) (WN-MAP (GRADUATE1))
       (AGENT (HUMAN) (SUBJ)) (THEME (ACADEMIC-DEGREE) ((PREP WITH)))
       (FROM-POSS (EDUCATIONAL-INSTITUTION ORGANIZATION)((PREP FROM)))]
```

The entry *wn-map* means that all the synsets of graduate1 and all the verbs that fall under the class of graduate1 are mapped into the predicate *graduate-from*. The entry for the *theme* is intended to interpret sentences such as "X graduated with a degree in Physics from Y," in which the *theme* is realized by [with NP] if the head noun of the NP is an *academic-degree*. The PP [from NP] matches the *from-poss* if the head noun of the NP is an *educational-institution*. This is the category preferred. However, if the head noun of the NP is not an *educational-institution*, but it is an *organization*, the *from-poss* role will also match. The default category, *organization*, is needed because some educational institutions are not part of the WN noun ontology.

The semantic interpretation algorithm reported in [3] is activated by the parser after parsing a clause. The parser does not resolve structural ambiguity, which is delayed until semantic interpretation. The goals of the algorithm are to select one predicate from the list of predicates for a verb form, attach PPs and identify semantic roles and adjuncts. For each grammatical relation (GR) in the clause and for every predicate in the list of predicates, the algorithm verifies if the predicate explains the GR. A predicate *explains* an GR if there is a semantic role in the predicate realized by the GR and the selectional restrictions of the semantic role subsume the ontological category of the head noun of the grammatical relation. This process is repeated for each GR in the clause and each predicate in the list of predicates. Then, the predicate that explains the most GRs is selected as the meaning of the verb. The semantic roles of the predicate have been identified as a result of this process. In case of ties, the predicate that has the greatest number of semantic roles realized is preferred. Every grammatical relation that has not been mapped into a semantic role must be an adjunct or an NP modifier. The entries for adjuncts are stored in the root node *action* and are inherited by all predicates. Adjuncts are recognized *after the meaning of the verb has been determined* because they are not part of the argument structure of the predicate.

3 Description of the Task

Assume that one wants to extract knowledge from the Encyclopedia about the schools attended by people as students. The system should build a template for each school attended by the person as a student. Each template built by the system must contain the following information if known: name of the school, school type, date of entrance, date of graduation, location of the school, subject/degree of study, age of the student at entrance, and age of student at graduation. When an extraction task begins, the fillers of the template are all initialized to nil. Hence the entire template is constructed from scratch by the semantic interpreter. The main relation to recognize is that of *attend school as a student*, which can be expressed in many ways. For instance, the text may say that person X entered school Y, that X was transferred to Y, that X graduated from Y, that X was educated at Y, that X received/got/obtained a degree from Y, that X studied at Y, that X was/became a student at Y, that X was an alumnus of Y, that X's parents sent X to Y, that X withdrew from Y, that Y accepted/admitted X, etc. Besides recognizing that all these verbs may imply *attend-a-school*, the algorithm must identify all semantic roles of the sentence and map them into the entries in the template. For instance, if the sentence says that “X graduated from Y in 1943,” the algorithm must recognize that “from Y” is the school attended by X and that “in 1943” is a temporal adjunct expressing the date in which X graduated. But, this mapping should not be from syntactic relations for the verb “graduate” to entries in the template, but from semantic roles for the predicate *graduate-from* to semantic roles for the predicate *attend-a-school*. There are several reasons for it, the most important being that other verbs besides “graduate” may express the relation “graduate-from,” such as “X received/obtained/got a degree from Y.” Another related reason is that the same semantic role may be expressed by different grammatical relations. Because the template is constructed from the semantic roles and from temporal and locative adjuncts, correct identification of semantic roles and adjuncts becomes critical for the precision and recall of the overall system.

4 Using the Hierarchical Organization of Predicates to Establish the Inferences

The hierarchical organization of predicates provided by the semantic interpreter does permit already to establish the inference *attend-a-school* for many verbs, because these are mapped into subpredicates of *attend-a-school*. For instance, the verb “enter” followed by a post-verbal NP whose head noun is an *educational-institution* is recognized by the interpreter as the predicate *enter-a-school* whose superconcepts are given by:

```
ENTER-A-SCHOOL → ATTEND-A-SCHOOL → ATTEND-AN-EVENT-ORGANIZATION →
      INTERACT → ACTION
```

The verb “transfer” followed by [to NP], where the head noun of the NP is an *educational-institution* is recognized as *transfer-to-school* which is also a subpredicate of *attend-a-school*. In these cases, the designer of the KE has to do nothing because the predicate *attend-a-school* already exists, and the integrator is going to integrate this predicate and its semantic roles onto the template. The hierarchies of predicates in the interpreter have been designed to maximize the inferences that can be established by inheritance and to anchor the inferences into a generic predicate rather than on individual senses of verb forms, which would lead to a proliferation of inference rules. However, inference rules connecting generic predicates will be needed as explained in the next section. These observations apply to every class of predicates constructed by the interpreter. For instance, if one wants to extract knowledge about the things people value/respect/appreciate, etc. the interpreter has already the predicate *value-something* whose hierarchy is:

```
VALUE-SOMETHING → RESPECT-VALUE-SOMETHING → JUDGE → ACTION
```

This predicate does not only include one of the senses of “value,” but all WordNet verbs under the class respect1 (see below), and *treasure*, *appreciate*, and one of the senses of “recognize.”

```
respect, esteem, value, prize, prise
       => think the world of
       => reverence, fear, revere, venerate
           => enshrine, saint
           => worship
       => admire, look up to
```

If one wants to build a template for each of the jobs somebody had, their location, time, and duration, the interpreter already provides a hierarchy of subpredicates of *work-be-employed*. For instance, the predicate do-service, encompassing such usages as “serve as ambassador/teacher/etc,” has the hierarchy:

```
DO-SERVICE → WORK-BE-EMPLOYEED → TRANSFER-OF-POSSESSION → ACTION
```

Some inference rules connecting a few other predicates to *work-be-employed* would be the only things that the KE designer would need to do.

5 Lateral Inferences

However, not all inferences can be established from the hierarchical organization of the predicates. Besides linking the predicates strictly up the hierarchy, predicates need to be connected laterally. This is done by defining inference rules. These rules infer predicates and map semantic roles from the inferring predicate into roles of the inferred predicate. For instance, the predicates *graduate-from*, *study-a-subject*, and others are not classified as subpredicates of *attend-a-school*. However, that relation needs to be inferred if the sentence is “X graduated from Y,” or “X studied at Y,” where Y is an *educational-institution*. The hierarchy for *receive-an-academic-degree* is:

```
RECEIVE-AN-ACADEMIC-DEGREE → GET-AWARD → GET →
      TRANSFER-OF-POSSESSION → ACTION
```

The output of the semantic interpreter for a sentence of the form "X received a Ph.D from Y" is: "X" is the agent, "a Ph.D" is the *theme*, and "from Y" is the *from-poss*. In this case, a rule needs to be defined in the predicate *receive-an-academic-degree*, which infers *attend-a-school* and maps semantic roles from *receive-an-academic-degree* to *attend-a-school*. This is the rule:

```
((if% x-is-a $from-poss educational-institution)
(add-inference
 (((pr (attend-a-school)) (agent ($agent)) (to-loc ($from-poss))
   (end-time ($at-time)) (degree-of-study ($theme))
   (graduate-at-age ($at-the-age))))))
```

The rule says that if the *from-poss* role of the predicate *receive-an-academic-degree* is a subconcept of *educational-institution*, then infer the predicate *attend-a-school* with *agent* the *agent* of the predicate *receive-an-academic-degree*, with *to-loc* the *from-poss* of *receive-an-academic-degree*, etc. In general, the syntax for role mapping in the inference rules is:

```
(<role>(<$role>)
```

where <role> is the role in the predicate being inferred, and <$role> is the role of the predicate in which the inference rule is anchored. If a <role> does not exist in the output of the interpreter, then the <$role> in the inferred predicate is nil.

Here again, the hierarchies of predicates minimize the need of inference rules since inference rules are inherited by subpredicates from superpredicates. For instance, *graduate-from* inherits the rule from *receive-an-academic-degree*. The inference rules can be viewed as a semantic network of predicates connected by conditional links. Besides connecting the predicates, the network maps semantic roles from predicate X into roles of predicate Y. Predicate Y may connect to other predicates, or infer other predicates as you prefer to express it, by means of these conditional links. Predicate X may infer Y, and Y may infer X. That is to say the link connecting X to Y may be bidirectional. For instance, one may want to infer that "X attended Y" from "X studied at Y," and that "X studied at Y" from "X attended Y." In fact, there is an inference rule on *attend-a-school* that infers *study-a-subject* and vice versa. The algorithm that fires the rules is not caught in a circularity because it keeps track of all predicates that have been inferred. The algorithm is:

- Let A be the interpretation structure built by the interpreter. Initialize the list Exclude to nil. Initialize the list Inferences to nil. After applying this algorithm, this list will contain the inferences obtained from the predicate in A.
  1. Let *pr* be the predicate in A. Add *pr* to the list Exclude.
  2. Let SI be the list of structures obtained from firing the inference rules associated with *pr*.
  3. For each structure *a1* in SI
     a) If the predicate of a1 is not in the list Exclude,
     b) Add a1 to the list Inferences.
     c) Apply steps 1, 2, and 3 with A replaced by a1.

These rules are easy to understand and can be easily written by someone with very little knowledge of natural language processing. For this application, we defined 35 rules anchored on 18 predicates. Space limitations impede us from illustrating this algorithm with some examples.

6 Testing

In order to test the system, we selected 50 articles at random from over 5,000 biographical articles in The World Book Encyclopedia. For each article selected, the template built by a human was compared with the template built by the system. We counted the number of slots in the template that were filled correctly by the system, the number that were filled incorrectly, and the number that were missed. We let *C* be the total number of correct slots for all articles, *I* the total number of incorrect slots, and *M* the total number of missed slots. Then, the measure of recall is given by C/(C+I+M), and the measure of precision is C/(C+I). The results obtained were 87% recall and 97% precision.

Many of the articles selected contained only two or three sentences relevant to the task. This is just the nature of the biographical articles in the World Book Encyclopedia. A lower number of articles contained between four and 10 relevant sentences, and a few others more than ten. The system gets very few incorrect slots, and therefore very high precision, because of the accuracy of the semantic interpreter. The system fails to interpret some adverbial clauses with an elliptical verb, e.g., “After a few months at Oxford University, Brummell was left ....” This is a problem that has recurred several times, and which we plan to solve in a general way. In the Carter and Eisenhower articles, the system fails to infer that “to receive an appointment to the US Naval Academy” means to be admitted to the US Naval Academy as student, e.g., “In 1942, ... Carter received an appointment to the US Naval Academy.” Other failures are due to some discourse problems, which in general are not acute in the Encyclopedia. We use a centering model [6] with specific knowledge based on the rhetorical structure of the encyclopedic articles. In the sentence, “When Dutch was 9 years old, he and ... settled in Dixon, Ill, where the boy finished high school,” the system does not resolve the definite reference “the boy.”

7 Sample Session

In this example we illustrate the performance of the system when reading the John F. Kennedy biographical article from the World Book Encyclopedia, and extracting knowledge concerning the educational institutions he attended. For each institution attended, the system fills a template consisting of the following slots:

```
[ (ATTENDED ()) (LOCATION ()) (TIME ()) (FROM-TIME ()) (END-TIME ())
  (AT-THE-AGE ()) (ENTER-AT-AGE ()) (GRADUATE-AT-AGE ()) (SUBJECT ()) ]
```

The "attended" slot indicates the institution attended, "location" is the location of the institution, "time" indicates when the institution was attended ("he attended Harvard in 1950"), "from-time" and "end-time" are the starting time and end time of attendance, "at-the-age," "enter-at-age" and "graduate-at-age" indicate attendance in terms of the age of the individual ("he entered Harvard at the age of 20"), and "subject" refers to the field of study.

The Kennedy article is fairly long, containing over 300 sentences, most of which are not related to the educational institutions attended by Kennedy. A "skimmer" module first selects the sentences deemed relevant to the knowledge extraction problem at hand, and only these sentences are interpreted. The sentences selected from the Kennedy article include the following:

John Kennedy attended elementary schools in Brookline and Riverdale. In 1930, when he was 13 years old, his father sent him to the Canterbury School in New Milford, Conn. The next year, he transferred to Choate Academy in Wallingford, Conn. Kennedy was graduated from Choate in 1935 at the age of 18. He enrolled at Princeton University that fall, but he developed jaundice and left school after Christmas. He entered Harvard University in 1936. There he majored in government and international relations. Kennedy was graduated cum laude in 1940. He then enrolled in the Stanford University graduate business school, but dropped out six months later.

The system parses, interprets and builds representation structures for each sentence in the input. Here we show the parser output for the first sentence:

```
G0 (SUBJ ((PN JOHN KENNEDY))
    VERB ((MAIN-VERB ATTEND ATTENDED) (TENSE SP))
    OBJ ((ADJ ELEMENTARY) (NOUN SCHOOLS))
    PREP (AND ((IN ((PN BROOKLINE)))) ((IN ((PN RIVERDALE))))))
```

The parser output becomes the input to the interpreter, which produces the following interpretation:

```
Clause CL1
   SUBJ  : (PERSON JOHN_FITZGERALD_KENNEDY)   <(AGENT)>
   VERB  : ATTENDED  <ATTEND-A-SCHOOL-I:NIL supported by 2 GRs>
   OBJ   : (GRADE_SCHOOL1 ELEMENTARY_SCHOOL1)   <(TO-LOC)>
   PREP  : AND
      IN PREP-NP: (LOCATION BROOKLINE)   <(AT-LOC)>
            Attach: Verb  Confidence: WEAK
      IN PREP-NP: (LOCATION RIVERDALE)   <(AT-LOC)>
            Attach: Verb  Confidence: WEAK
```

The verbal concept is identified as "attend-a-school." The "agent" of the action is Kennedy, a subconcept of "person." The "to-loc" role is "elementary school," a subconcept of "grade-school." Brookline and Riverdale are subconcepts of "location" and fill the "at-loc" semantic role.

The interpreter output is then transformed into the following set of knowledge representation structures:

```
RIVERDALE
    (instance-of (location))  (related-to (@a9) (@a10) (@a11))
BROOKLINE
    (instance-of (location))  (related-to (@a9) (@a10) (@a11))
ELEMENTARY_SCHOOL1
    (is-a (grade_school1))  (related-to (@a9) (@a10) (@a11))
JOHN_FITZGERALD_KENNEDY
    (is-a (person))  (attend-a-school ($null ($more (@a9) (@a11))))
    (study-a-subject ($null ($more (@a10))))
```

```
 @A9
    (args (john_fitzgerald_kennedy) (elementary_school1) (brookline)
(riverdale))
    (pr (attend-a-school))
    (agent (john_fitzgerald_kennedy (q (constant))))
    (to-loc (elementary_school1 (q (constant))))
    (at-loc (brookline (q (constant))) (riverdale (q (constant))))
    (instance-of (action))  (time (past))
 ATTEND-A-SCHOOL
    (related-to (@a9) (@a11))
 @A10
    (instance-of (inference (@a9)))  (pr (study-a-subject))
    (agent (john_fitzgerald_kennedy (q (constant))))
    (args (john_fitzgerald_kennedy) (elementary_school1) (brookline)
(riverdale))
    (at-educational-institution (elementary_school1 (q (constant))))
    (at-loc (brookline (q (constant))) (riverdale (q (constant))))
    (time (past))
 STUDY-A-SUBJECT
    (related-to (@a10))
```

The knowledge in these structures is used to fill the predefined knowledge extraction template, yielding the entry:

```
JOHN_FITZGERALD_KENNEDY
   (ATTEND-A-SCHOOL (ELEMENTARY_SCHOOL1))
   (LOCATION (BROOKLINE) (RIVERDALE))
```

After reading all the relevant sentences, the output produced is:

```
JOHN_FITZGERALD_KENNEDY
   (ATTEND-A-SCHOOL (ELEMENTARY_SCHOOL1))
      (LOCATION (BROOKLINE) (RIVERDALE))
   (ATTEND-A-SCHOOL (CANTERBURY_SCHOOL))
      (LOCATION (NEW_MILFORD))  (TIME (1930))
   (TRANSFER-TO-SCHOOL (CHOATE_ACADEMY))
      (LOCATION (WALLINGFORD))  (TIME (NEXT_YEAR))
      (FROM-TIME (NEXT_YEAR))  (GRADUATE-AT-AGE (18))  (END-TIME (1935))
   (ATTEND-A-SCHOOL (PRINCETON_UNIVERSITY1))
      (FROM-TIME (THAT_FALL))
   (ENTER-A-SCHOOL (HARVARD_UNIVERSITY1))
      (TIME (1936))  (FROM-TIME (1936))
      (SUBJECT (GOVERNMENT1) (INTERNATIONAL_RELATION))
   (ATTEND-A-SCHOOL (STANFORD_UNIVERSITY_GRADUATE_BUSINESS_SCHOOL))
      (FROM-TIME (THEN1))
```

Some time references still need to be solved, such as “next-year,” “that-fall,” and “then,” which can be done by accessing the other entries in the frame. However, these temporal references have not been implemented. These results show the system's high degree of precision. Recall is also high, having missed Kennedy's age when entering Canterbury School, the end time for Princeton, the graduation date for Harvard, and the end time for Stanford.

8 Related Work

This work is related to that described in [7] in which the acquisition of knowledge is closely connected to the semantic interpretation process. A paper that deals with the issue of inferences using WN is [8]. The authors implement a marker propagation algorithm that uses the verb entailment, the glosses and the concept hierarchy in WN. As the authors observe, the lack of semantic relations for the verbs and the few number of entailments that WN provides are some of the serious limitations with their approach.

There have been several systems in relation to the MUC project [12] that extract patterns from texts. These systems rely on the user to identify the relevant patterns, or on annotated corpora. None of these systems approach the semantic interpretation of complete sentences. In some of these systems, the user identifies the patterns of interest and the system uses WN for the generalization process. Riloff [13] generates extraction patterns from annotated texts. Other systems require pre-constructed templates [1]. However, a semi-automated system that does not require annotated texts is [14] that constructs a domain lexicon by using a bootstrapping algorithm that starts with a set of *seed* words, and adds new words belonging to a semantic category. The enhanced list of seed words is then reviewed by a human who selects the words that should be added to the domain lexicon from those proposed by the algorithm. This system may be very useful for building lexicons for specialized domains, but not for acquiring knowledge from encyclopedic texts which deal with general domain knowledge. Moreover, because the system does not address the issues of semantic interpretation in a general context, its scope of applications will be limited to the extraction of some well-defined patterns. Similar remarks apply to the work on acquiring hyponyms from patterns that originated in [9]. This work does not assign meaning to the constituents of the sentence.

This work also differs from work reported in [10] in that the knowledge acquisition designer does not have to be concerned with defining ontological categories, or semantic interpretation rules because they are already part of the semantic interpreter. Moreover, the ontological categories, namely those of WordNet, are of a general nature and have received a wide acceptance in the natural language processing community.

A critique that can be leveled against our approach could be that it needs the hand-crafted construction of verb predicates, which is a rather difficult and time-consuming job. The reply to this is that once the verb predicates are defined, they are defined for every natural language application. This is so because their definitions are not tied to any given application, and their selectional restrictions are based on a general ontology of English. In [4], the reader may find a progress to date of the goal of building predicates for English verbs regardless of domain. In this paper, we have shown that the predicates can be applied to a knowledge extraction task from an encyclopedia of intermediate complexity.

9 Conclusions

We have described a knowledge extraction system that acquires knowledge from encyclopedic texts. The system is based on a general semantic interpreter of English that uses the WordNet ontology for nouns and verb predicates constructed for WordNet verb classes. Because the knowledge extraction system and the semantic interpreter share the same ontology and because the inferences of the KE are based on the structure and organization of the predicates used by the semantic interpreter, the definition of new extraction tasks is relatively easy. The system has been tested in the The World Book Encyclopedia producing very solid results.

References

1. M. E. Califf and R.J. Mooney. Relational learning of pattern match rules for information extraction. In *Proc. of the ACL Workshop on Natural Language Learning*, pages 9-15, Stanford, California, 1997.
2. C. Fellbaum. A semantic network of english verbs. In C. Fellbaum, editor, WordNet: *An electronic Lexical Database and some of its applications*, page chapter 3. MIT Press, 1998, Cambridge, Mass, 1998.
3. F. Gomez. An algorithm for aspects of semantic interpretation using an enhanced wordnet. In Proccedings of the 2nd North American Meeting of the North American Association for Computational Linguistics, NAACL-2001, pages 87-94, 2001.
4. F. Gomez. Building verb predicates: A computational view. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, ACL-04*, pages 351- 358, Barcelona, 2004.
5. F. Gomez. Grounding the ontology on the semantic interpretation algorithm. In *Proceedings of the Second International WordNet Conference*, pages 124-129, Masaryk University, Brno, 2004.
6. B. J. Grosz, A. K. Joshi, and S. Weinstein. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):201-225, 1995.
7. U. Hahn and K. Schnattinger. A text understander that learns. In *COLING-ACL*, pages 476-482, Montreal, Quebec, 1998.
8. S. Harabagiu and D. Moldovan. Knowledge processing on extended wordnet. In C. Fellbaum, editor, *WordNet: An electronic Lexical Database and some of its applications*, pages 379-405. MIT Press, 1998, Cambridge, Mass, 1998.
9. M.A. Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proc. of COLING-92*, pages 530-545, Nantes, France, 1992.
10. R.D. Hull and F. Gomez. Automatic acquisition of biographic knowledge from encyclopedic texts. *Expert Systems with Applications*, 16:261-270, 1999.
11. George Miller. Nouns in wordnet. In C. Fellbaum, editor, *WordNet: An electronic Lexical Database and some of its applications*, page chapter 1. MIT Press, 1998, Cambridge, Mass, 1998.
12. MUC-4. *Proc. of the Fourth Message Understanding Conference (MUC-4)*. Morgan Kaufmann, San Mateo, California, 1992.
13. E. Riloff. An empirical study of automated dictionary construction for information extraction in three domains. *Artificial Intelligence*, 85:101-134, 1996.
14. E. Riloff and M. Schmelzenbach. An empirical approach to conceptual case frame acquisition. In *Proc. of the Sixth Workshop on Very Large Corpora*, pages 0-0, 1998.
15. Y.A. Wilks. Preference semantics. In E.L. Keenan, editor, *Formal Semantics of Natural Language.* Cambridge University Press, Cambridge, UK, 1975.