

# Learning by Reading: An Experiment in Text Analysis

Eduard Hovy

(with collaborators Hans Chalupsky, Jerry Hobbs, Chin-Yew Lin, Patrick Pantel,  
Andrew Philpot, and students Rutu Mulkar and Marco Pennacchiotti)

Information Sciences Institute, University of Southern California  
4676 Admiralty Way  
Marina del Rey, CA, USA  
hovy@isi.edu

**Abstract.** It has long been a dream to build computer systems that learn automatically by reading text. This dream is generally considered infeasible, but some surprising developments in the US over the past three years have led to the funding of several short-term investigations into whether and how much the best current practices in Natural Language Processing and Knowledge Representation and Reasoning, when combined, actually enable this dream. This paper very briefly describes one of these efforts, the Learning by Reading project at ISI, which has converted a high school textbook of Chemistry into very shallow logical form and is investigating which semantic features can plausibly be added to support the kinds of inference required for answering standard high school text questions.

## 1 Context and Background

From almost the beginnings of Artificial Intelligence, it was clear that automated systems require knowledge to reason intelligently, and that for multi-purpose, wide-domain, robust reasoning, the amount required is nontrivial. Experience, especially with expert systems during the 1970s, illustrated just how hard it is to acquire enough of the right knowledge, and how difficult it is to formalize that knowledge in ways suitable for supporting reasoning. Naturally, then, the dream arose to enable systems to read text and learn by themselves. But this dream has never been realized. In fact, as research in Knowledge Representation and Reasoning (KR&R) and Natural Language Processing (NLP) progressed, the two areas diverged, to the point where today they are more or less entirely separate, with unrelated conferences, journals, and research paradigms.

A few years ago, three research groups, funded by Vulcan Inc., participated in an audacious experiment called Project Halo: to (manually) convert the information contained in one chapter of a high school textbook on Chemistry into knowledge representation statements, and then to have the knowledge representation system take a standard high school Advanced Placement (AP) exam. Surprisingly, two of the three systems passed, albeit at a relatively low level of performance. The project engendered wide interest; see (Friedland et al., 2003).

Over the past year, DARPA has funded five groups in the US to conduct pilot studies that investigate the feasibility of building fully Learning by Reading (LbR) systems. The largest, Project Möbius, is a consortium of some 20 researchers from numerous institutions. Its goal

is to design a general framework for LbR systems in the future, and to advise DARPA on the wisdom of funding a new program in this area. Typical questions include: How feasible is fully automated LbR? What are the different phases/components/steps of LbR? What are the current levels of capability of the component technologies, and where are the major bottlenecks and failure points? What kind of research would best further the dream of LbR? What sorts of results could one expect after five years?

The remaining four projects proceed independently but report back to Möbius. All are smaller, 9-month efforts, and each focuses on one or more specific aspects of the general LbR problem:

- A project jointly at Boeing and SRI, led by Peter Clark of Boeing, focuses on the mismatch between English sentences and their equivalent knowledge representations propositions, with the methodology of building manually the representations for a carefully selected extract of 5 pages from the Chemistry textbook.
- A project at Northwestern University, led by Ken Forbus, concentrates on the processes of self-guided inference that occurs after new information is read. Called *introspection* or *rumination*, these processes work in parallel with the reading, and serve as a source of expectations, questions, and background checking. This project focuses on a few selected sentences from the Chemistry textbook, and the thoughts that may arise from them.
- A project at CYC Corp., led by Michael Witbrock, addresses the problem of learning the meaning of new, unknown, words in context. Starting with the knowledge already inside their very large ontology and reasoning system Cyc, researchers develop methods to apply inferences in order to build up likely interpretations of a sentence and from these hypothesize the meaning of the unknown word.
- The fourth project is the subject of this paper. Located at ISI, we are investigating how much can be done by combining traditional and statistical NLP methods, and what kinds of KR&R are absolutely required, at which points in the process. We have parsed the whole Chemistry textbook, have developed methods to convert the parses into shallow logical form, and are investigating the what types of -semantics should be added to support the reasoning required for question answering.

In this paper we briefly outline the architecture and general aspects of ISI's LbR project, which will finish in August, namely about three months after the time of writing this paper.

## 2 The Text

The text is *Chemistry: The Central Science* (9th ed.), by Brown, LeMay, Bursten, and Burdge, which is intended for senior high-school students. The publishers have kindly made an electronic version available to the projects, to be used for research purposes only.

The book contains 313590 words (12722 different words). Of this, nouns comprise 146825 / 9279 tokens/types, verbs 28627 / 1387 tokens/types, and adjectives and adverbs together 35775 / 2795. The most frequent 50 nouns cover 67.2% of all noun tokens in the book; the most frequent 50 verbs cover 92.4% of the verb tokens, and the most frequent 50 adjectives/adverbs cover 83.7%. This is relatively good news: should one have to hand-construct core meanings of (most of) these words before the systems starts to read, it will suffice to treat only the most frequent 200 nouns (covering 96.1% of all noun tokens), 50 verbs, and 50 adjectives/adverbs.

### 3 The Learning by Reading Architecture and Modules

#### 3.1 System Architecture

Given the very short time available—9 months from start to finish—the project is necessarily very incomplete; we decided to focus on just selected areas of interest and to simplify or ignore many of the interesting questions we encountered. Since our strength lies in language processing technology, we focus on the NLP areas, with the intent to discover how little KR&R we needed, and how shallow semantics is possible, to still produce something interesting. We therefore consciously sidestep the many complex phenomena involved in semantic understanding, as opposed to syntactic parsing, including in particular wordsense disambiguation, coreference and anaphora resolution, quantification, negation, complex NP structure and modification, discourse structure, and so on.

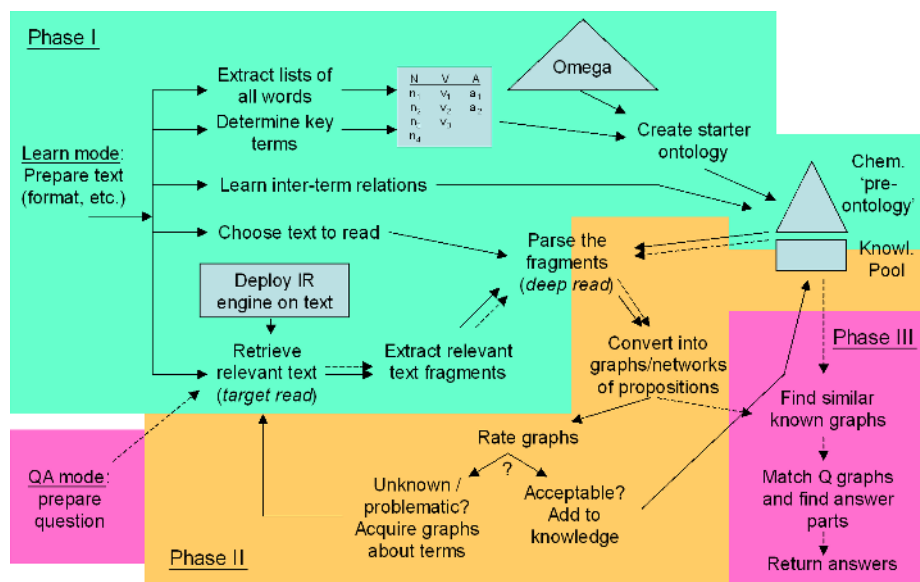
One might ask: if you ignore all this, what's the point? And indeed, designing some kind of evaluation, or at least some criteria by which one can measure and/or describe the validity and utility of the outcome, is a significant challenge, as we discuss in Section 4. Nonetheless, it is extremely interesting and quite instructive to proceed under these very strong simplifications, and to then discover what phenomena of semantics are in fact required for a practical application of LbR (as opposed to what phenomena have received a lot of attention from researchers), as well as to investigate whether there are simple and robust methods to derive these phenomena using modern techniques.

The overall system architecture is shown in Figure 1. The system was built in three 3-month stages: stage 1 covered textbook analysis, background knowledge structures, and syntactic parsing; stage 2 covered conversion of parse structures into shallow logical form, creation of notations for selected semantic phenomena, and building of inferences / transformation rules to create these representations; stage 3 covers instantiation of the final output for into a KR&R system and experiments with question answering.

#### 3.2 The Skeletal Ontology

No human would dream of reading a book like this without any knowledge whatsoever of English words. Since we had only 9 months for the project, we decided to extract from our large general-purpose term taxonomy/ontology Omega (Philpot et al., 2005) just the terms used in the book, with their inheritance structure, verb case frame information, and selected other relationships such as paronyms and synonyms. (The largest part of Omega, the Middle Model, was derived by combining WordNet 2.1 (Fellbaum et al., 1998) with Mikrokosmos (Mahesh, 1996) and adding to it a variety of additional information, notably the verb frame structures of Framenet (Baker et al., 1998), Lexical-Conceptual Structures (Dorr and Habash, 1991), and the PropBank frame (Palmer et al. 2005). The LbR 'ontology' we derived consists of 10371 terms, of which 6117 were already in Omega (the remaining terms were multi-word phrases, unknown proper names, etc.).

In a technical book such as this, complex NPs pose a serious problem for understanding. No current NLP engines do an adequate job of analyzing their internal structure. To avoid this problem, and to fill in the rather large gap in the ontology due to these terms, Pantel applied his Mutual Information-based method (see for example (Pantel, 2005; Pantel and Ravichandran, 2004) for similar word on noun clusters and concept formation) to identify



**Fig. 1.** ISI's Learning by Reading system architecture. The top region was completed in stage 1; the middle region in stage 2, and the bottom (left and right) region in stage 3.

bigrams, trigrams, and longer strings of nouns that occur unusually frequently in this corpus compared to general English. We obtained several thousand multi-word terms, including (starting with highest mutual information) *practice exercise*, *Solution Analysis*, *equilibrium constant*, *periodic table*, *boiling point*, *equilibrium constant*, *partial pressures*, and so on, which we added to the ontology through simple automated matching of the final noun, which we assumed was the head in all cases.

Although each term in the book is represented in the skeletal ontology, this does not mean much is actually known about the concept. The system knows in all cases the English forms of a word (plural, past tense, etc.), its purported taxonomic location (according to Omega, which is not specialized for Chemistry), and the frame structure of some of the verbs. It does not however have the words in the textbook disambiguated by sense: to the system, for example, the noun *water* might mean the normal fluid we drink, a sea or lake, or any of several other meanings. (Our initial assumption that words would tend to be monosemous because we are dealing only with a single domain turned out not to hold; even in Chemistry, many words have different senses. For example, “water” might refer to the substance in general, to its formula in an equation, or to a specific instance of it as being discussed in an experiment,

### 3.3 Information Retrieval

We identified the need to support three modes of reading: *skimming*; general sentence-by-sentence processing, as one normally reads a book (which we called *deep reading*); and *targeted reading*, in which one focuses on a specific term and reads fragments about it from all over the text. As described in Section 3.6, we used skimming to extract relations from

the book using statistical methods. In contrast, as discussed in Sections 3.4 and 3.5, targeted reading is required during more typical reading, both to find details about some term that may be lacking crucial information and to answer questions (Section 3.7). To enable targeted reading, Lin deployed over the textbook an instance of the Lucene information retrieval package (Lucene 2005), and built a small web-based interface for it, by which one can explore the book manually as well. This IR system and its interface we called Scout.

### 3.4 Parsing and Conversion to Hobbs Normal Form

We decomposed the conversion of natural language text into logical form into a series of steps. In this process we used our Basic Element package, built at ISI earlier this year for automated summarization evaluation (Hovy et al., 2006), which can be downloaded from <http://www.isi.edu/~cyl/BE>. The purpose of the package is to convert sentences of text into a list of relation-head-modifier triples and then to compare the lists obtained from automated text summarizers to the lists produced from gold-standard human summaries.

For LbR, we required just the triples. Using the package, we applied both Charniak's parser (Charniak, 2000) and MINIPAR (Lin, 1994) to the whole book, and for technical reasons chose the former. Using the BE package tree-decomposition rules, Lin converted the entire book's contents into so-called flat form. Next using a set of conversion rules written manually by Lin, each flat form statement was converted into what we call Hobbs Normal Form (HNF), which is a list of clauses, one per triplet, combined using logical AND, that expresses the dependency structure of the sentence in quasi-logical format. During this process, Lin also replaced the syntactic relations Subject and Object with the appropriate case frame roles obtained from Omega, using a process trained on the PropBank corpus. A brief example of a heading is shown in Figure 2.

This process has not been fully optimized, and still contains quite a few easily fixed errors (as well as some unfortunately not so easily fixed). Of the 12394 sentences in the book, 51.7% contain some kind of known error, including 2219 verbs without arguments (broken parse trees), 2621 cases in which some argument (case frame role) has been wrongly duplicated within a sentence, and 752 auxiliary verbs with main verb. Some of these errors can be ignored for later purposes; many of them can be fixed by inserting additional tree-cutting rules. Parsing the whole book took 39 minutes on a standard PC; creating the basic element triples took 340 minutes; and converting into HNF took an additional 15 minutes.

```
0201-1.fr.txt:# 2.1/1 The/2 Atomic/3 Theory/4 of/5 Matter/6
0201-1.fr.txt:
0201-1.fr.txt: [ X4 :type Theory/NN/theory
0201-1.fr.txt:      :#NN-CD 2.1/CD/2.1<* ( X1 )
0201-1.fr.txt:      :#NN-NNP The/NNP/the<* ( X2 )
0201-1.fr.txt:      :#NN-JJ Atomic/JJ/atomic<* ( X3 )
0201-1.fr.txt:      :#NN-NN Matter/NN/matter<OF ( X6 ) ]
0201-1.fr.txt:
0201-1.fr.txt: @@@LF 2.1' (e1,x1) & the' (e2,x2) & nn' (e3,x2,x1)
                   & atomic' (e4,x1) & theory' (e5,x1) & of' (e6,x1,x3)
                   & matter' (e7,x3)
```

Fig. 2. Input sentence, Basic Element triples, and HNF

### 3.5 From Hobbs Normal Form to Limited Semantics

HNF is our starting point en route to semantics. This notation is a simplified variant of the logical form developed over almost a decade by Jerry Hobbs (Hobbs, forthcoming). Our implementation avoids numerous complex problems, including wordsense disambiguation, quantifier and negation scoping, complex NP structure, and other issues. Nonetheless, it provides a starting point into which we could systematically add representations of the kinds of logical phenomena required to answer questions in the Chemistry domain. Guided by analysis of the text and questions, Hobbs and Mulkar implemented a set of abductive inference rules that takes as input HNF and returns a set of possible ‘deeper’ interpretations of them, in which some particular semantic phenomenon has been addressed each time. At the time of writing, the phenomena for which rules were created include:

- Determiner interpretation: insertion of  $\forall$  and  $\exists$  quantifiers (without scope)
- Plural handling: insertion of sets to represent aggregations, plus skolem variables to represent their individual item(s). These sets are denoted by  $S$  variables in the HNF

An example will be helpful. Consider the sentence

*All atoms of a given element are identical.*

After parsing and basic element conversion, the flat logical form is

$all(s,e1) \& atom'(e1,x) \& plural(x,s) \& of'(e2,x,y) \& given(y)$   
 $\& element'(e3,y) \& identical1'(e4,s) \& MainAssertion(e4)$

Here each  $e$  variable expresses an eventuality, each  $s$  variable a set (as introduced by a plural), and each  $x$  or  $y$  a specific instance. After application of the determiner inference rule and canonicalization of symmetric predicates, we obtain

$FORALL(x,e1,e4) \& atom'(e1,x) \& of'(e2,x,y) \& element'(e3,y) \& identical1'(e4,s)$

and after recursively collecting the properties of  $x$ , the final result is an axiom in chemical theory:

$(\forall x)[atom(x1) \& atom(x2) \& of(x1,y) \& of(x2,y) \& element(y) \rightarrow identical2(x1,x2)]$

The engine performing this transformation is a simplified version of the abductive theorem prover TACITUS (Hobbs, 1986). It combines weight scores, originally assigned to inferences, in order to obtain numerical goodness scores for each derived result, and then ranks them accordingly.

Since each rule handles a different semantic phenomenon, the rules have to be written by hand. How many such rules will eventually be required, and will the rule set eventually grow unmanageable? This is impossible to estimate, but Hobbs is optimistic. In a passage of 7 clauses selected for its high content, for example, all clauses are of form  $(\forall x)[Subject(x) \rightarrow (\exists y) VP(x, y)]$

The density of occurrence of the principal phenomena we have so far encountered in the book (which has a relatively homogeneous style of exposition) suggests that a relatively small number of rules/phenomena will do a lot of the work in translating sentences into content theory axioms.

### 3.6 Skimming to Obtain Relations

In a completely separate stream of work, Pantel and Pennacchiotti investigated the extraction of axioms from the text using the statistical text harvesting paradigm. In (Pennacchiotti and Pantel, 2006; Pantel and Pennacchiotti, 2006) they report the results. Their harvester, called Espresso, uses weak supervision to learn lexical patterns that typically signal relations of interest, and then applies these relations to extract all likely instances of the relation from the book. These extractions are then reformulated as HNF axioms and added to the system's knowledge.

Espresso follows in the pattern-based extraction paradigm of Hearst (1992), Berland and Charniak (1999), Ravichandran and Hovy (2002), Fleischmann et al. (2003), and Girju et al. (2003). The patterns themselves are not manually built, but are learned from the text after a small set of seed instances is given by the user, following the general design of Ravichandran and Hovy. The most reliable patterns are identified using pointwise mutual information, and the overly general ones are discarded. The resulting patterns are applied throughout the textbook, to deliver a set of instances of the relation. Once instances have been collected, pointwise mutual information is again used to prefer the ones best associated with the most (reliable) patterns.

Espresso was applied to the Chemistry textbook, in which the relations of particular interest (because they provide useful axioms in the domain) are *is-a*, *part-of*, *react-with*, and *produces*, as in *HCl is-a strong acid*, *atom part-of molecule*, *hydrogen-gas reacts-with oxygen-gas*, and *ammonia produces nitrous oxide* respectively. After quality filtering, Espresso produced 200 *is-a* instances (with average precision of 85.0%), 111 *part-of* (precision 60.0%), 40 *reacts-with* (precision 85.0%), and 196 *produces* (precision 72.5%).

The resulting instance expressions were then reformulated as axioms as follows:

*R is-a S* becomes  $R(x) \Diamond S(x)$

*R part-of S* becomes  $(\forall x)R(x) \Diamond (\exists y)[S(y) \ \& \ \textit{part-of}(x,y)]$

### 3.7 Answering Questions: Inferences in Powerloom<sup>1</sup>

Given the modules described above, not much more is required to answer questions. The parser's grammar has to be extended to handle question syntax, and a rudimentary question typology is required, along the lines of the QA typology described in (Hovy et al., 2002).

Most important, however, is the inclusion of a reasoning system to apply learned inferences to newly-acquired propositions or questions in order to derive appropriate connections with the existing knowledge. For this purpose we use Powerloom (Chalupsky and Russ, 2002), a knowledge representation and reasoning system that has been widely distributed and used in numerous projects over the past decade. All axioms and semantic propositions derived from HNF are asserted into Powerloom, which is responsible for their interconnection and maintenance.

The process of asserting semantic propositions into Powerloom, performed by Chalupsky, makes very clear how much is taken for granted in the textbook as background knowledge, either of Chemistry or of English in general. For example, for the sentence *each element*

<sup>1</sup> This work is still underway at the time of writing and hence this section does not include many details.

*composed of extremely small particles called atoms* the shallow semantic representation, derived from HNF and asserted to Powerloom, is

```
(ASSERT
  (FORALL (?E34 ?E35 ?x)
    (=> (AND
      (subject ?E34 ?E35)
      (element' ?E35 ?x))
      (EXISTS (?E62 ?E63 ?s1 ?e10 ?y ?e4 ?e9 ?e3 ?e5 ?a ?z ?e11 ?s2 ?e6)
        (AND
          (asserted ?E62 ?E63)
          (compose' ?E63 ?x ?s1)
          (plural ?e10 ?y ?s1)
          (small' ?e4 ?y)
          (extremely ?e9 ?e4)
          (particle' ?e3 ?y)
          (call' ?e5 ?a ?y ?z)
          (plural ?e11 ?z ?s2)
          (atom' ?e6 ?z))))))
```

With this knowledge, Powerloom is able to answer question 1 but not question 2:

1. *Is each element composed of particles?*
2. *Is each element composed of atoms?*

because it does not know that the relation *call'* denotes object identity—that it, it does not know the meaning of the word “called”.

Probably the most problematic aspect of this project is the fact there exists no set of axioms that define the meanings of general English words, even to the rudimentary level required for the simplest questions. This lack places a serious limitation on the amount of text a short-term project such as this can handle. Fortunately, as mentioned in Section 2, one need not define more than a few hundreds nouns, verbs, adjectives, and adverbs in order to cover most of the book.

## 4 Evaluation

It remains unclear exactly how one should evaluate a Learning by Reading system such as this. The general QA paradigm used in Halo and planned for Möbius—obtain a set of AP exam questions, apply them to the system before reading and then again to the system after reading, and compare the results—is somewhat inappropriate because in a 9-month period one simply cannot build enough of the system, and include enough background knowledge in the form of axioms, to be able to understand enough text for a real-world exam.

We therefore plan two approaches to evaluation. First, we are investigating the kinds of questions the system can be expected to handle, and are building a typology of the question types, arranged by complexity of reasoning. This typology will range from the simplest questions (being able to answer *Is it true that X?* after having been given proposition *X*), to formula-oriented computations (such as balancing equations), to the most complex ones



(involving deeper inference with several axioms, obtained from several places in the text). If possible, we will try to estimate the frequencies of these types in typical AP exams, in order to estimate how much of each type of background knowledge (definitions of English; formula reasoning; simple and complex inference; etc.) will be required of a full-blown LbR system.

Second, we are assembling a set of questions about material about which the system will have been given the necessary background knowledge. We will apply these questions three times, at levels of increasing complexity:

1. Baseline: just in English, using the Scout IR engine on the untreated English text;
2. No inference: after the system has read the text and asserted its knowledge and axioms in Powerloom, but without inference allowed;
3. With inference: after the system has read and ingested the text, with Powerloom inference enabled.

We will apply the same two relatively transparent evaluation metrics at each level, the first counting just how much relevant material/propositions have been found, and the second counting how many exact and correct answers have been found. (We are curious to learn the results, and hope that they show that inference is useful, in other words, that pure text-level IR is not enough for Chemistry.)

These two approaches to evaluation do not constitute a real evaluation, but will be informative in helping to understand just how far one can expect a Learning by Reading system of this (language-oriented) type to go, and in just what kinds of areas it falls short.

## References

1. Baker, C.F., C.J. Fillmore, and J.B. Lowe. 1998. The Berkeley FrameNet Project. *Proceedings of the COLING/ACL conference*, 86–90.
2. Berland, M. and E. Charniak. 1999. Finding Parts in Very Large Corpora. *Proceedings of the conference of the Association for Computational Linguistics (ACL-99)*, 57–64.
3. Chalupsky, H. and T.A. Russ. 2002. WhyNot: Debugging Failed Queries in Large Knowledge Bases. *Proceedings of the Fourteenth Innovative Applications of Artificial Intelligence conference (IAAI-02)*, 870–877.
4. Charniak, E. 2000. A Maximum-Entropy-Inspired Parser. *Proc. of NAACL '00 conference*.
5. Dorr, B.J. and N. Habash. 2001. Lexical Conceptual Structure Lexicons. In Calzolari et al. (eds), ISLE-IST-1999-10647-WP2-WP3, *Survey of Major Approaches Towards Bilingual/Multilingual Lexicons*.
6. Fellbaum, C. (ed.). 1998. *WordNet: An On-line Lexical Database and Some of its Applications*. MIT Press.
7. Fleischmann, M., E.H. Hovy, and A. Echihiabi. 2003. Offline Strategies for Online Question Answering: Answering Questions before They are Asked. *Proceedings of the conference of the Association for Computational Linguistics (ACL-03)*, 1–7.
8. Friedland, N. et al., 2003. <http://www.projecthalo.com/halotempl.asp?cid=30>.
9. Girju, R., A. Baldulescu, and D. Moldovan. 2003. Learning Semantic Constraints for the Automatic Discovery of Part-hole Relations. *Proceedings of HLT-NAACL-03 conference*, 80–87.
10. Hearst, M. 1992. Automatic Acquisition of Hyponyms from Large Text Corpora. *Proceedings of the COLING-92 conference*, 539–545.
11. Hobbs, J.R. 1986. Overview of the TACITUS Project. *Computational Linguistics* 12(3).
12. Hobbs, J.R. (forthcoming) *Discourse and Inference*.

13. Hovy, E.H., U. Hermjakob, C.-Y. Lin, and D. Ravichandran. 2002. Using Knowledge to Facilitate Pinpointing of Factoid Answers. *Proceedings of the COLING-2002 conference*. Available at [http://www.isi.edu/natural-language/projects/webclopedia/Taxonomy/taxonomy\\_toplevel.html](http://www.isi.edu/natural-language/projects/webclopedia/Taxonomy/taxonomy_toplevel.html).
14. Hovy, E.H., C.-Y. Lin, L. Zhou, and J. Fukumoto. 2006. Automated Summarization Evaluation with Basic Elements. 2006. *Proceedings of the LREC conference*.
15. Lin, D. 1994. Principar — An Efficient, Broad-Coverage, Principle-Based Parser. *Proceedings of the COLING/ACL conference*, 42–48.
16. Lucene. 2005. Open Source software <http://lucene.apache.org/java/docs/>.
17. Mahesh, K. 1996. Ontology Development for Machine Translation: Ideology and Methodology. New Mexico State University CRL report MCCS-96-292.
18. Palmer, M., D. Gildea, and P. Kingsbury. 2005. The Proposition Bank: A Corpus Annotated with Semantic Roles, *Computational Linguistics*, 31(1).
19. Pantel, P. and D. Ravichandran. 2004. Automatically Labeling Semantic Classes. *Proceedings of the HLT-NAACL-04 conference*.
20. Pantel, P. 2005. Inducing Ontological Co-occurrence Vectors. *Proceedings of the conference of the Association for Computational Linguistics (ACL-05)*.
21. Pantel, P. and Pennacchiotti, 2006. Espresso: Leveraging Generic Patterns for Automatically Harvesting Semantic Relations. *Proceedings of the conference of the Association for Computational Linguistics (COLING/ACL-06)*.
22. Pennacchiotti, M. and P. Pantel. 2006. A Bootstrapping Algorithm for Automatically Harvesting Semantic Relations. *Proceedings of the ICOS-06 conference*.
23. Philpot, A., E.H. Hovy, and P. Pantel. 2005. The Omega Ontology. *Proceedings of the ONTOLEX Workshop at the International Joint Conference on NLP*. Jeju Island, Korea.
24. Ravichandran, D. and E.H. Hovy. 2002. Learning Surface Text Patterns for a Question Answering System. *Proceedings of the conference of the Association for Computational Linguistics (ACL-02)*, 41–47.