

Exploiting Strong Syntactic Heuristics and Co-Training to Learn Semantic Lexicons

William Phillips and Ellen Riloff

School of Computing

University of Utah

Salt Lake City, UT 84112 USA

{phillips,riloff}@cs.utah.edu

Abstract

We present a bootstrapping method that uses strong syntactic heuristics to learn semantic lexicons. The three sources of information are appositives, compound nouns, and ISA clauses. We apply heuristics to these syntactic structures, embed them in a bootstrapping architecture, and combine them with co-training. Results on WSJ articles and a pharmaceutical corpus show that this method obtains high precision and finds a large number of terms.

1 Introduction

Syntactic structure helps us understand the semantic relationships between words. Given a text corpus, we can use knowledge about syntactic structures to obtain semantic knowledge. For example, Hearst (Hearst, 1992) learned hyponymy relationships by collecting words in lexico-syntactic expressions, such as “NP, NP, and other NPs”, and Roark and Charniak (Roark and Charniak, 1998) generated semantically related words by applying statistical measures to syntactic contexts involving appositives, lists, and conjunctions.

Exploiting syntactic structures to learn semantic knowledge holds great promise, but can run into problems. First, lexico-syntactic expressions that explicitly indicate semantic relationships (e.g., “NP, NP, and other NPs”) are reliable but a lot of semantic information occurs outside these expressions. Second, general syntactic structures (e.g., lists and conjunctions) capture a wide range of semantic relationships. For example, conjunctions frequently

join items of the same semantic class (e.g., “cats and dogs”), but they can also join different semantic classes (e.g., “fire and ice”). Some researchers (Roark and Charniak, 1998; Riloff and Shepherd, 1997) have applied statistical methods to identify the strongest semantic associations. This approach has produced reasonable results, but the accuracy of these techniques still leaves much room for improvement.

We adopt an intermediate approach that learns semantic lexicons using *strong* syntactic heuristics, which are both common and reliable. We have identified certain types of appositives, compound nouns, and identity (ISA) clauses that indicate specific semantic associations between words. We embed syntactic heuristics in a bootstrapping process and present empirical results demonstrating that this bootstrapping process produces high-quality semantic lexicons. In another set of experiments, we incorporate a co-training (Blum and Mitchell, 1998) mechanism to combine the hypotheses generated by different types of syntactic structures. Co-training produces a synergistic effect across different heuristics, substantially increasing the coverage of the lexicons while maintaining nearly the same level of accuracy.

2 Semantic Lexicon Learning

The goal of our research is to automatically generate a *semantic lexicon*. For our purposes, we define a semantic lexicon to be a list of words with semantic category labels. For example, the word “bird” might be labeled as an ANIMAL and the word “car” might be labeled as a VEHICLE. Semantic lexicons have proven to be useful for many lan-

guage processing tasks, including anaphora resolution (Aone and Bennett, 1996; McCarthy and Lehnert, 1995), prepositional phrase attachment (Brill and Resnik, 1994), information extraction (Soderland et al., 1995; Riloff and Schmelzenbach, 1998), and question answering (Harabagiu et al., 2000; Hirschman et al., 1999).

Some general-purposes semantic dictionaries already exist, such as WordNet (Miller, 1990). WordNet has been used for many applications, but it may not contain the vocabulary and jargon needed for specialized domains. For example, WordNet does not contain much of the vocabulary found in medical texts. In previous research on semantic lexicon induction, Roark and Charniak (Roark and Charniak, 1998) showed that 3 of every 5 words learned by their system were not present in WordNet. Furthermore, they used relatively unspecialized text corpora: Wall Street Journal articles and terrorism news stories. Our goal is to develop techniques for semantic lexicon induction that could be used to enhance existing resources such as WordNet, or to create dictionaries for specialized domains.

Several techniques have been developed to generate semantic knowledge using weakly supervised learning techniques. Hearst (Hearst, 1992) extracted information from lexico-syntactic expressions that explicitly indicate hyponymic relationships. Hearst’s work is similar in spirit to our work in that her system identified reliable syntactic structures that explicitly reveal semantic associations. Meta-bootstrapping (Riloff and Jones, 1999) is a semantic lexicon learning technique very different from ours which utilizes information extraction patterns to identify semantically related contexts. Named entity recognizers (e.g., (Bikel et al., 1997; Collins and Singer, 1999; Cucerzan and Yarowsky, 1999)) can be trained to recognize proper names associated with semantic categories such as PERSON or ORGANIZATION, but they typically are not aimed at learning common nouns such as “surgeon” or “drugmaker”.

Several researchers have used some of the same syntactic structures that we exploit in our research, namely appositives and compound nouns. For example, Riloff and Shepherd (Riloff and Shepherd, 1997) developed a statistical co-occurrence model for semantic lexicon induction that was designed

with these structures in mind. Roark and Charniak (Roark and Charniak, 1998) followed up on this work by using a parser to explicitly capture these structures. Caraballo (Caraballo, 1999) also exploited these syntactic structures and applied a cosine vector model to produce semantic groupings. In our view, these previous systems used “weak” syntactic models because the syntactic structures sometimes identified desirable semantic associations and sometimes did not. To compensate, statistical models were used to separate the meaningful semantic associations from the spurious ones. In contrast, our work aims to identify “strong” syntactic heuristics that can isolate instances of general structures that reliably identify the desired semantic relations.

3 A Bootstrapping Model that Exploits Strong Syntactic Heuristics

For the purposes of this research, we will define two distinct types of lexicons. One lexicon will consist of proper noun phrases, such as “Federal Aviation Administration”. We will call this the *PNP (proper noun phrase) lexicon*. The second lexicon will consist of common (non-proper) nouns, such as “airplane”. We will call this the *GN (general noun) lexicon*. The reason for creating these distinct lexicons is that our algorithm takes advantage of syntactic relationships between proper nouns and general nouns.

3.1 Syntactic Heuristics

Our goal is to build a semantic lexicon of words that belong to the same semantic class. More specifically, we aim to find words that have the same hypernym, for example “dog” and “frog” would both have the hypernym ANIMAL.¹ We will refer to words that have the same immediate hypernym as *semantic siblings*.

We hypothesize that some syntactic structures can be used to reliably identify semantic siblings. We have identified three candidates: appositives, compound nouns, and identity clauses whose main verb is a form of “to be” (we will call these ISA clauses).

¹The appropriate granularity of a set of semantic classes, or the organization of a semantic hierarchy, is always open to debate. We chose categories that seem to represent important and relatively general semantic distinctions.

While these structures often do capture semantic siblings, they frequently capture other types of semantic relationships as well. Therefore we use heuristics to isolate subsets of these syntactic structures that consistently contain semantic siblings. Our heuristics are based on the observation that many of these structures contain both a proper noun phrase and a general noun phrase which are co-referent and usually belong to the same semantic class. In the following sections, we explain the heuristics that we use for each syntactic structure, and how those structures are used to learn new lexicon entries.

3.1.1 Appositives

Appositives are commonly occurring syntactic structures that contain pairs of semantically related noun phrases. A simple appositive structure consists of a noun phrase (NP), followed by a comma, followed by another NP, where the two NPs are co-referent. However, appositives often signify hypernym relationships (e.g., “the dog, a carnivorous animal”).

To identify semantic siblings, we only use appositives that contain one proper noun phrase and one general noun phrase. For example, “George Bush, the president” or “the president, George Bush”. Theoretically, such appositives could also indicate a hypernym relationship (e.g., “George Bush, a mammal”), but we have found that this rarely happens in practice.

3.1.2 Compound Nouns

Compound nouns are extremely common but they can represent a staggering variety of semantic relationships. We have found one type of compound noun that can be reliably used to harvest semantic siblings. We loosely define these compounds as “GN⁺ PNP” noun phrases, where the compound noun ends with a proper name but is modified with one or more general nouns. Examples of such compounds are “violinist James Braum” or “software maker Microsoft”. One of the difficulties with recognizing these constructs, however, is resolving the ambiguity between adjectives and nouns among the modifiers (e.g., “violinist” is a noun). We only use constructs in which the GN modifier is unambiguously a noun.

3.1.3 ISA Clauses

Certain “to be” clauses can also be harvested to extract semantic siblings. We define an ISA clause as an NP followed by a VP that is a form of “to be”, followed by another NP. These identity clauses also exhibit a wide range of semantic relationships, but harvesting clauses which contain one proper NP and one general NP can reliably identify noun phrases of the same semantic class. We found that this structure yields semantic siblings when the subject NP is constrained to be a proper NP and the object NP is constrained to be a general NP (e.g., “Jing Lee is the president of the company”).

3.2 The Bootstrapping Model

Figure 1 illustrates the bootstrapping model for each of the three syntactic structures. Initially, the lexicons contain only a few manually defined seed words: some proper noun phrases and some general nouns. The syntactic heuristics are then applied to the text corpus to collect potentially “harvestable” structures. Each heuristic identifies structures with one proper NP and one general NP, where one of them is already present in the lexicon as a member of a desired semantic class. The other NP is then assumed to belong to the same semantic class and is added to a *prospective word list*. Finally, statistical filtering is used to divide the prospective word lists into exclusive and non-exclusive subsets. We will describe the motivation for this in Section 3.2.2. The exclusive words are added to the lexicon, and the bootstrapping process repeats. In the remainder of this section, we explain how the bootstrapping process works in more detail.

3.2.1 Bootstrapping Procedure

The input to our system is a small set of *seed words* for the semantic categories of interest. To identify good seed words, we sorted all nouns in the corpus by frequency and manually identified the most frequent nouns that belong to each targeted semantic category.

Each bootstrapping iteration alternates between using either the PNP lexicon or the GN lexicon to grow the lexicons. As a motivating example, assume that (1) appositives are the targeted syntactic structure (2) bootstrapping begins by using the PNP lexicon, and (3) PEOPLE is the semantic category of

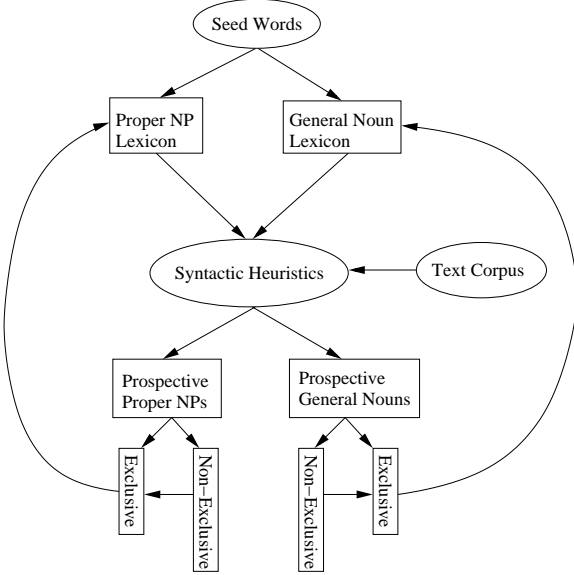


Figure 1: Bootstrapping Model

interest. The system will then collect all appositives that contain a proper noun phrase known to be a person. So if “Mary Smith” belongs to the PNP lexicon and the appositive “Mary Smith, the analyst” is encountered, the head noun “analyst” will be learned as a person.

The next bootstrapping iteration uses the GN lexicon, so the system will collect all appositives that contain a general noun phrase known to be a person. If the appositive “John Seng, the financial analyst” is encountered, then “John Seng” will be learned as a person because the word “analyst” is known to be a person from the previous iteration. The bootstrapping process will continue, alternately using the PNP lexicon and the GN lexicon, until no new words can be learned.

We treat proper noun phrases and general noun phrases differently during learning. When a proper noun phrase is learned, the full noun phrase is added to the lexicon. But when a general noun phrase is learned, only the head noun is added to the lexicon. This approach gives us generality because head nouns are usually (though not always) sufficient to associate a common noun phrase with a semantic class. Proper names, however, often do not exhibit this generality (e.g., “Saint Louis” is a location but “Louis” is not).

However, using full proper noun phrases can limit

the ability of the bootstrapping process to acquire new terms because exact matches are relatively rare. To compensate, head nouns and modifying nouns of proper NPs are used as *predictor* terms to recognize new proper NPs that belong to the same semantic class. We identify reliable predictor terms using the *evidence* and *exclusivity* measures that we will define in the next section. For example, the word “Mr.” is learned as a good predictor term for the person category. These predictor terms are only used to classify noun phrases during bootstrapping and are not themselves added to the lexicon.

3.2.2 Exclusivity Filtering

Our syntactic heuristics were designed to reliably identify words belonging to the same semantic class, but some erroneous terms still slip through for various reasons, such as parser errors and idiomatic expressions. Perhaps the biggest problem comes from ambiguous terms that can belong to several semantic classes. For instance, in the financial domain “leader” can refer to both people and corporations. If “leader” is added to the person lexicon, then it will pull corporation terms into the lexicon during subsequent bootstrapping iterations and the person lexicon will be compromised.

To address this problem, we classify all candidate words as being *exclusive* to the semantic category or *non-exclusive*. For example, the word “president” nearly always refers to a person so it is exclusive to the person category, but the word “leader” is non-exclusive. Only the *exclusive* terms are added to the semantic lexicon during bootstrapping to keep the lexicon as pure (unambiguous) as possible. The *non-exclusive* terms can be added to the final lexicon when bootstrapping is finished if polysemous terms are acceptable to have in the dictionary.

Exclusivity filtering is the only step that uses statistics. Two measures determine whether a word is exclusive to a semantic category. First, we use an *evidence measure*:

$$Evidence(w, c) = \frac{S_{w,c}}{S_w}$$

where S_w is the number of times word w was found in the syntactic structure, and $S_{w,c}$ is the number of times word w was found in the syntactic structure collocated with a member of category c . The evidence measure is the maximum likelihood estimate

that a word belongs to a semantic category given that it appears in the targeted syntactic structure (a word is assumed to belong to the category if it is collocated with another category member). Since few words are known category members initially, we use a low threshold value (.25) which simply ensures that a non-trivial proportion of instances are collocated with category members.

The second measure that we use, *exclusivity*, is the number of occurrences found in the given category’s prospective list divided by the number of occurrences found in all other categories’ prospective lists.

$$Exclusivity(w, c) = \frac{S_{w,c}}{S_{w,-c}}$$

where $S_{w,c}$ is the number of times word w was found in the syntactic structure collocated with a member of category c , and $S_{w,-c}$ is the number of times word w was found in the syntactic structure collocated with a member of a different semantic class. We apply a threshold to this ratio to ensure that the term is exclusive to the targeted semantic category.

3.3 Experimental Results

We evaluated our system on several semantic categories in two domains. In one set of experiments, we generated lexicons for PEOPLE and ORGANIZATIONS using 2500 Wall Street Journal articles from the Penn Treebank (Marcus et al., 1993). In the second set of experiments, we generated lexicons for PEOPLE, ORGANIZATIONS, and PRODUCTS using approximately 1350 press releases from pharmaceutical companies.²

Our seeding consisted of 5 proper nouns and 5 general nouns for each semantic category. We used a threshold of 25% for the evidence measure and 5 for the exclusivity ratio. We ran the bootstrapping process until no new words were learned, which ranged from 6-14 iterations depending on the category and syntactic structure.

Table 1 shows 10 examples of words learned for each semantic category in each domain. The people and organization lists illustrate (1) how dramatically the vocabulary can differ across domains, and (2) that the lexicons may include domain-specific word meanings that are not the most common meaning

²We found these texts using Yahoo’s financial industry pages at <http://biz.yahoo.com/news/medical.html>.

People (WSJ): adman, co-chairman, head, economist, shareholder, AMR Chairman Robert Crandall, Assistant Secretary David Mullins, Deng Xiaoping, Abby Joseph Cohen, C. Everett Koop

Organization (WSJ): parent, subsidiary, distiller, arm, suitor, AMR Corp., ABB ASEA Brown Boveri, J.P. Morgan, James River, Federal Reserve Board

People (Pharm): surgeon, executive, recipient, co-author, pioneer, Amgen Chief Executive Officer, Barbara Ryan, Chief Scientific Officer Norbert Riedel, Dr. Cole, Analyst Mark Augustine

Organization (Pharm): device-maker, drug-maker, licensee, organization, venture, ALR Technologies, Aventis Pharmaceuticals, Bayer AG, FDA Advisory Panel, Hadassah University Hospital

Product (Pharm): compound, stent, platform, blocker, antibiotic, Bexxar, Viratrol, MBX-102, Apothesys Decision Support System, AERx Pain Management System

Table 1: Examples of Learned Words

of a word in general. For example, the word “parent” generally refers to a person, but in a financial domain it nearly always refers to an organization. The pharmaceutical product category contains many nouns (e.g., drug names) that may not be in a general purpose lexicon such as WordNet.

Tables 2 and 3 show the results of our evaluation. We ran the bootstrapping algorithm on each type of syntactic structure independently. The Total column shows the total number of lexicon entries generated by each syntactic structure. The Correct column contains two accuracy numbers: X/Y. The first value (X) is the percentage of entries that were judged to be correct, and the second value (Y) is the accuracy after removing entries resulting from parser errors.³

The PNP lexicons were substantially larger than the GN lexicons, in part because we saved full noun

³For example, our parser frequently mistags adjectives as nouns, so many adjectives were hypothesized to be people. If the parser had tagged them correctly, they would not have been allowed in the lexicon.

Category	Appositives		Compounds		ISA		Union	
	Total	Correct	Total	Correct	Total	Correct	Total	Correct
<i>People (WSJ)</i>	1826	.97/.97	2026	.99/.99	113	.94/.94	3543	1.0/1.0
<i>Orgs (WSJ)</i>	674	.87/.94	3770	.77/.78	54	.93/.96	4191	.79/.79
<i>People (Pharm)</i>	280	.86/.87	1723	.87/.88	39	1.0/1.0	1872	.85/.91
<i>Orgs (Pharm)</i>	205	.85/.88	1128	.85/.92	248	.85/.91	1399	.78/.84
<i>Products (Pharm)</i>	64	.94/.95	223	.77/.79	64	.84/.84	330	.83/.85

Table 2: Proper Noun Phrase Lexicon Results

Category	Appositives		Compounds		ISA		Union	
	Total	Correct	Total	Correct	Total	Correct	Total	Correct
<i>People (WSJ)</i>	159	.91/.95	60	.30/.56	41	.85/.97	229	.73/.88
<i>Orgs (WSJ)</i>	84	.69/.75	54	.26/.47	6	1.0/1.0	134	.52/.66
<i>People (Pharm)</i>	34	.91/.91	32	.66/.75	18	1.0/1.0	66	.79/.84
<i>Orgs (Pharm)</i>	36	.58/.60	29	.35/.46	41	.51/.66	95	.45/.54
<i>Products (Pharm)</i>	8	.75/1.0	11	.09/.33	13	.54/1.0	32	.50/.89

Table 3: General Noun Lexicon Results

phrases in the PNP lexicon but only head nouns in the GN lexicon. Probably the main reason, however, is that there are many more proper names associated with most semantic categories than there are general nouns. Consequently, we evaluated the PNP and GN lexicons differently. For the GN lexicons, a volunteer (not one of the authors) labeled every word as correct or incorrect. Due to the large size of the PNP lexicons, we randomly sampled 100 words for each syntactic structure and semantic category and asked volunteers to label these samples. Consequently, the PNP evaluation numbers are estimates of the true accuracy.

The Union column tabulates the results obtained from unioning the lexicons produced by the three syntactic structures independently.⁴ Although there is some overlap in their lexicons, we found that many different words are being learned. This indicates that the three syntactic structures are tapping into different parts of the search space, which suggests that combining them in a co-training model could be beneficial.

⁴Since the number of words contributed by each syntactic structure varied greatly, we evaluated the Union results for the PNP lexicon by randomly sampling 100 words from the unioned lexicons regardless of which structure generated them. This maintained the same distribution in our evaluation set as exists in the lexicon as a whole. However, this sampling strategy means that the evaluation results in the Union column are not simply the sum of the results in the preceding columns.

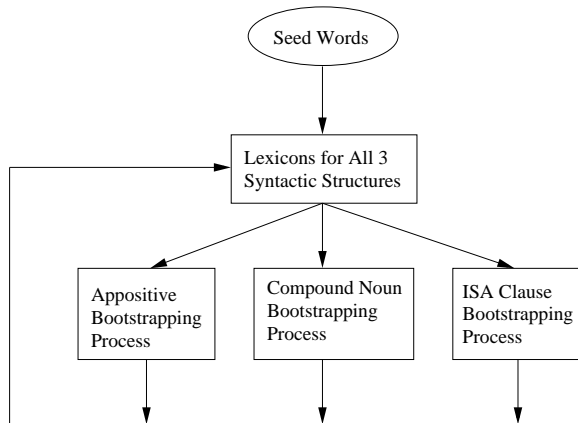


Figure 2: Co-Training Model

3.4 Co-Training

Co-training (Blum and Mitchell, 1998) is a learning technique which combines classifiers that support different views of the data in a single learning mechanism. The co-training model allows examples learned by one classifier to be used by the other classifiers, producing a synergistic effect. The three syntactic structures that we have discussed provide three different ways to harvest semantically related noun phrases.

Figure 2 shows our co-training model, with each syntactic structure serving as an independent classifier. The words hypothesized by each classifier are

put into a single PNP lexicon and a single GN lexicon, which are shared by all three classifiers. We used an aggressive form of co-training, where all terms hypothesized by a syntactic structure with frequency $\geq \theta$ are added to the shared lexicon. The threshold ensures some confidence in a term before it is allowed to be used by the other learners. We used a threshold of $\theta=3$ for the WSJ corpus and $\theta=2$ for the pharmaceutical corpus since it is substantially smaller. We ran the bootstrapping process until no new words were learned, which was 12 iterations for the WSJ corpus and 10 iterations for the pharmaceutical corpus.⁵

Category	PNP cotrn	PNP w/o	GN cotrn	GN w/o
<i>People (WSJ)</i>	5414	3543	347	229
<i>Orgs (WSJ)</i>	4227	4191	213	134
<i>People (Pharm)</i>	2217	1872	84	66
<i>Orgs (Pharm)</i>	4068	1399	196	95
<i>Products (Pharm)</i>	309	330	38	32

Table 4: Lexicon sizes with and w/o co-training

Table 4 shows the size of the learned lexicons with co-training and without co-training (i.e., running the classifiers separately). In almost all cases, many additional words were learned using the co-training model. Tables 5 and 6 show the evaluation results for the lexicons produced by co-training. The co-training model produced substantially better coverage, while achieving nearly the same accuracy. One exception was organizations in the pharmaceutical domain, which suffered a sizeable loss in precision. This is most likely due to the co-training loop being too aggressive. If one classifier produces a lot of mistakes (in this case, the compound noun classifier), then those mistakes can drag down the overall accuracy of the lexicon.

4 Conclusions

We have presented a method for learning semantic lexicons that uses strong syntactic heuristics in a bootstrapping algorithm. We exploited three types of syntactic structures (appositives, compound NPs,

⁵After co-training finished, we also added terms to the lexicon that were hypothesized by an individual classifier with frequency $< \theta$ if they had not previously been labeled.

and ISA clauses) in combination with heuristics to identify instances of these structures that contain both a proper and general noun phrase. Each syntactic structure generated many lexicon entries, in most cases with high accuracy. We also combined the three classifiers using co-training. The co-training model increased the number of learned lexicon entries, while maintaining nearly the same level of accuracy. One limitation of this work is that it can only learn semantic categories that are commonly found as proper nouns and general nouns.

This research illustrates that common syntactic structures can be combined with heuristics to identify specific semantic relationships. So far we have experimented with three structures and one type of heuristic (proper NP/general NP collocations), but we believe that this approach holds promise for other semantic learning tasks as well. In future work, we hope to investigate other types of syntactic structures that may be used to identify semantically related terms, and other types of heuristics that can reveal specific semantic relationships.

5 Acknowledgements

This research was supported by the National Science Foundation under award IRI-9704240. Thanks to Erin Davies, Brijesh Garabadu, Dominic Jones, and Henry Longmore for labeling data.

References

- C. Aone and S. W. Bennett. 1996. Applying machine learning to anaphora resolution. In Stefan Wernter, Ellen Riloff, and Gabriele Scheler, editors, *Connectionist, Statistical, and Symbolic Approaches to Learning for Natural Language Processing*, pages 302–314. Springer-Verlag, Berlin.
- Daniel M. Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel. 1997. Nymble: a high-performance learning name-finder. In *Proceedings of ANLP-97*, pages 194–201.
- A. Blum and T. Mitchell. 1998. Combining Labeled and Unlabeled Data with Co-Training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT-98)*.
- E. Brill and P. Resnik. 1994. A Transformation-based Approach to Prepositional Phrase Attachment Disambiguation. In *Proceedings of the Fifteenth International Conference on Computational Linguistics (COLING-94)*.
- S. Carballo. 1999. Automatic Acquisition of a Hypernym-Labeled Noun Hierarchy from Text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 120–126.

Category	Appositives		Compounds		ISA		Union	
	Total	Correct	Total	Correct	Total	Correct	Total	Correct
<i>People (WSJ)</i>	1890	.98/.98	4979	.99/.99	143	.90/.90	5414	.99/.99
<i>Orgs (WSJ)</i>	744	.83/.88	3791	.77/.77	115	.76/.78	4227	.78/.78
<i>People (Pharm)</i>	292	.87/.88	2132	.82/.90	56	.80/.82	2217	.81/.90
<i>Orgs (Pharm)</i>	281	.79/.80	3872	.53/.59	305	.77/.82	4068	.49/.50
<i>Products (Pharm)</i>	65	.94/.95	225	.78/.80	69	.84/.84	309	.83/.86

Table 5: Proper Noun Phrase Lexicon Results after Co-Training

Category	Appositives		Compounds		ISA		Union	
	Total	Correct	Total	Correct	Total	Correct	Total	Correct
<i>People (WSJ)</i>	200	.89/.93	160	.58/.78	73	.69/.81	347	.69/.83
<i>Orgs (WSJ)</i>	125	.66/.71	66	.26/.46	55	.46/.60	213	.47/.60
<i>People (Pharm)</i>	44	.86/.86	38	.66/.74	30	.90/.93	84	.75/.80
<i>Orgs (Pharm)</i>	73	.56/.59	90	.23/.35	70	.49/.61	196	.36/.47
<i>Products (Pharm)</i>	9	.78/1.0	17	.24/.67	17	.65/1.0	38	.50/.91

Table 6: General Noun Lexicon Results after Co-Training

- M. Collins and Y. Singer. 1999. Unsupervised Models for Named Entity Classification. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99)*.
- S. Cucerzan and D. Yarowsky. 1999. Language Independent Named Entity Recognition Combining Morphological and Contextual Evidence. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99)*.
- S. Harabagiu, D. Moldovan, M. Pasca, R. Mihalcea, Surdeanu M., R. Bunescu, R. Girju, V. Rus, and P. Morarescu. 2000. FALCON: Boosting Knowledge for Answer Engines. In *Proceedings of the Ninth Text Retrieval Conference (TREC-9)*.
- M. Hearst. 1992. Automatic Acquisition of Hyponyms from Large Text Corpora. In *Proceedings of the Fourteenth International Conference on Computational Linguistics (COLING-92)*.
- Lynette Hirschman, Marc Light, Eric Breck, and John D. Burger. 1999. Deep Read: A reading comprehension system. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*.
- M. Marcus, B. Santorini, and M. Marcinkiewicz. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.
- Joseph F. McCarthy and Wendy G. Lehnert. 1995. Using Decision Trees for Coreference Resolution. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 1050–1055.
- G. Miller. 1990. Wordnet: An On-line Lexical Database. *International Journal of Lexicography*, 3(4).
- E. Riloff and R. Jones. 1999. Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*.
- E. Riloff and M. Schmelzenbach. 1998. An Empirical Approach to Conceptual Case Frame Acquisition. In *Proceedings of the Sixth Workshop on Very Large Corpora*, pages 49–56.
- E. Riloff and J. Shepherd. 1997. A Corpus-Based Approach for Building Semantic Lexicons. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 117–124.
- B. Roark and E. Charniak. 1998. Noun-phrase Co-occurrence Statistics for Semi-automatic Semantic Lexicon Construction. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics*, pages 1110–1116.
- S. Soderland, D. Fisher, J. Aseltine, and W. Lehnert. 1995. CRYSTAL: Inducing a conceptual dictionary. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 1314–1319.