ACQUISITION OF SEMANTIC LEXICONS

Using Word Sense Disambiguation to Improve Precision

BONNIE J. DORR
Department of Computer Science and
Institute for Advanced Computer Studies
University of Maryland
bonnie@umiacs.umd.edu

AND

DOUG JONES
Natural Language Processing
Research Branch
U.S. Department of Defense
jones@afterlife.ncsc.mil

Abstract

This paper addresses the problem of large-scale acquisition of computational-semantic lexicons from machine-readable resources. We describe semantic filters designed to reduce the number of incorrect assignments (i.e., improve precision) made by a purely syntactic technique. We demonstrate that it is possible to use these filters to build broad-coverage lexicons with minimal effort, at a depth of knowledge that lies at the syntax-semantics interface. We report on our results of disambiguating the verbs in the semantic filters by adding WordNet sense annotations. We then show the results of our classification on unknown words and we evaluate these results.

1. Introduction

This paper addresses the problem of large-scale acquisition of computational-semantic lexicons from machine-readable resources. As online dictionaries, thesauri, and other knowledge sources become readily available to NLP researchers, automated acquisition has become increasingly more attractive. Several researchers have noted that the average time needed to construct a lexical entry by hand can be as much as 30 minutes (Neff and McCord, 1990;

Copestake *et al.*, 1995; Walker and Amsler, 1986). Given that most large-scale NLP applications require lexicons of 20-60,000 words, automation of the acquisition process has become a necessity.

Previous research in automatic acquisition focuses primarily on the use of statistical techniques for bilingual alignment (Church and Hanks, 1990; Klavans and Tzoukermann, 1995; Wu and Xia, 1995), extraction of syntactic constructions from online dictionaries and corpora (Alshawi, 1989; Copestake *et al.*, 1995; Farwell *et al.*, 1993; Boguraev and Briscoe, 1989; Brent, 1993; Dorr *et al.*, 1995), or derivation of semantic structures from definition analyses (Lonsdale *et al.*, 1995; Wilks *et al.*, 1989; Wilks *et al.*, 1990). These earlier investigations do not take full advantage of the systematic relation between syntax and semantics during the lexical acquisition process.

Several researchers have argued that the task of simplifying lexical entries on the basis of broad semantic class membership is complex and, perhaps, infeasible (see, e.g., Boguraev and Briscoe (1989)). However, a number of researchers (Fillmore, 1968; Grimshaw, 1990; Gruber, 1965; Guthrie *et al.*, 1991; Hearst, 1991; Jackendoff, 1983; Jackendoff, 1990; Levin, 1993; Pinker, 1989; Yarowsky, 1992; Sanfilippo and Poznanski, 1992) have demonstrated conclusively that there is a clear relationship between syntactic context and word senses; it is our aim to exploit this relationship for the acquisition of semantic lexicons. The work of Sanfilippo and Poznanski (1992) is the most closely related to our approach in that it attempts to recover a syntactic-semantic relation from machine-readable dictionaries. However, they claim that the semantic classification of verbs based on standard machine-readable dictionaries (e.g., the LDOCE) is "a hopeless pursuit [since] standard dictionaries are simply not equipped to offer this kind of information with consistency and exhaustiveness."

We adopt the central thesis of Levin (1993), i.e., that the semantic class of a verb and its syntactic behavior are predictably related. We exploit this syntax-semantics relation in a large-scale experiment for classification of novel verbs, i.e., those not occurring in Levin's classes. We start by coupling a syntactic-filter approach (Dorr *et al.*, 1995) with a semantic-filter approach, using the synonymy relation defined in WordNet (Miller, 1986; Miller, 1990; Miller and Fellbaum, 1991).[1] The semantic filter is designed to reduce the number of incorrect assignments produced by the syntactic-filter approach. We classify all of the verbs in Longman's Dictionary of Contemporary English (LDOCE) (Procter, 1978) using the coupled filter approach. The results reveal a substantial improvement in *precision*—62.5%

[1]We used Version 1.5 of WordNet, available at http://www.cogsci.princeton.edu/~wn.

from a baseline of 6.5%—over the syntactic-filter approach alone on *known* verbs, i.e., just those LDOCE verbs that are classified in (Levin, 1993).

Next, we simulate the classification process on *unknown* verbs by randomly leaving out verbs from Levin's classes. Automatic evaluation of the technique on these unknown verbs reveals 42–49% precision, still much higher than the precision of the syntactic-filter approach on known verbs.

Finally, we show how the semantic filter can be further enhanced with automatic techniques for word-sense disambiguation based on a pre-defined association between WordNet word senses and Levin's verbs. The success of the approach is evaluated both by an automatic method—examining the classification of verbs that are randomly left out of (Levin, 1993) (64% precision)—and by a non-automatic method—hand-verifying the classification of verbs entirely outside of (Levin, 1993) (56-66% precision).

We have used these techniques to build a database of 7767 English verb entries containing semantic information. We are currently porting this information into languages such as Arabic, Spanish, and Korean for multilingual NLP tasks such as foreign language tutoring and machine translation. Our results clearly indicate that word sense disambiguation is a key component to developing effective techniques for automatic lexical acquisition.

2. Verb Classification Based on Syntactic Behavior

In (Dorr and Jones, 1996) we demonstrated that there is a strong link between the syntax and the semantics of verbs if we are able to disambiguate verbs. Although it is important from a theoretical point of view that this link exists, it is difficult to exploit it in constructing large scale lexicons. The biggest obstacle, of course, is that the verbs in online dictionaries are not disambiguated according to the semantic classes that Levin has described. For this reason, we will build on existing work which establishes links from the syntactic encoding of verbs in LDOCE to Levin's semantic classes. The work of Dorr et al. (1995) provides a baseline from which to begin this investigation. This section gives a brief overview of the syntactic-filter approach of Dorr et al. (1995). We will show how we combine these results with a technique for word-sense disambiguation in order to provide a method for accurate semantic classification of both known and unknown words.

The syntactic filter approach of (Dorr *et al.*, 1995) produced a ranked assignment of verbs indicating the most likely semantic classes for each verb in (Levin, 1993) based on syntactic tests (e.g., whether a verb occurs in a dative construction such as *Mary gave John the book*).[2] For example,

[2]The 191 classes from (Levin, 1993) cover 2813 verbs that occur in the LDOCE. Since verbs may occur in multiple classes, the number of possible assignments of LDOCE verbs

the verb *enrich* was correctly classified in semantic class 9.8 *Fill Verbs* by virtue of its LDOCE codes—T1 (as in *The discovery of oil will enrich the nation*), T1-BY (as in *Their relationship was enriched by his heightened awareness*), and T1-WITH (as in *The dish is enriched with cream*).[3]

The syntactic approach alone was demonstrated to classify Levin verbs with 47% accuracy (i.e., 1812 correct verb classifications out of 3851 possible assignments). However, this measure of success is flawed in that the "accuracy" factor was based on the number of correct assignments in the five top-ranked assignments produced by their algorithm.[4] In fact, if we examine the 15 top-ranked assignments per verb, the result would have been 67.7% (2607 correct verb classifications out of 3851 possible correct assignments). Clearly, we could continue *ad infinitum*, generating assignments (both correct and incorrect), until all possible correct assignments were covered.

A better measure of the efficacy of the algorithm would be to examine the ratio of correct assignments to the total number of assignments. The algorithm in (Dorr *et al.*, 1995) is correct only 13% of the time (1812 correct assignments out of 13761 total assignments) if given up to 5 assignments per verb.[5] If given up to 15 assignments, the situation would deteriorate further: even though 2607 out of 3851 possible assignments would be correct, these correct assignments constitute only 6.5% of the total number of assignments made by the algorithm.

We borrow terminology from Information Retrieval to characterize these results. Table 1 illustrates the general case of evaluating a set of binary decisions.[6] Here, a system is required to make n binary decisions, and each decision has one correct answer (either *yes* or *no*). The entries in the table show the number of decisions with each possible result. The cell a refers to the number of times the system decided *yes*, when *yes* was the correct answer (i.e., 40248 in the case where up to 15 assignments are made per verb). For our specific case, the *Decides Yes* row refers to the times that the filter assigns a verb to a given Levin class. The *Yes is Correct* column refers to the times that the filter assigns a verb to a given Levin class, and the verb appears in that class in Levin's book. Using this terminology, we now turn to an evaluation of the syntactic filter.

into classes is 3851.

[3]It also incorrectly classified *enrich* in 4 additional semantic classes (47.8, 31.1, 33, and 45.4). We will see shortly that the syntactic filter approach is flawed in that the incorrect class assignments vastly outnumber the correct class assignments.

[4]Similar remarks are given by Yarowsky (1992) regarding the contrast between qualitative measures and finer-grained distinctions such as precision and recall.

[5]There would be 14,065 (2813 x 5) assignments, but in some cases, there were fewer than 5 assignments that were statistically correlated with a verb.

[6]This table is taken from (Lewis, 1992), page 75.

	Yes is Correct	No is Correct	
Decides Yes	a	b	$a+b$
Decides No	c	d	$c+d$
	$a+c$	$b+d$	$a+b+c+d=n$

TABLE 1. Contingency Table for a Set of Binary Decisions

From the definitions in Table 1, we use the following terms:

1. recall $= a/(a+c)$
2. precision $= a/(a+b)$
3. fallout $= b/(b+d)$

Table 2 shows the settings of a, b, c, d, and n for the experiment described above; from this we see that the algorithm in (Dorr *et al.*, 1995) achieves a recall of 67.7%, but a precision of 6.5%.

	Yes is Correct	No is Correct	
Decides Yes	2607	37641	40248
Decides No	1244	495791	497035
	3851	533432	537283

TABLE 2. Results of Experiment by Dorr et al. (1995)

In addition to the high number of *incorrect* assignments (i.e., low precision) produced by the syntactic filter, the algorithm described above was applied only to the LDOCE verbs that are in (Levin, 1993), i.e., 2813 verbs (3851 potential class assignments). Moreover, the experiment did not take into account the problem of polysemous word senses. Although the

LDOCE divides its syntactic codes into groups corresponding to separate verb senses, the algorithm assumed a single set of LDOCE codes for each English verb, even in cases where a subset of the codes in a pattern corresponds to more than one verb meaning. This was a problem for cases such as the verb *sleep*, which appears in Levin's class 40.4 ("Snooze Verbs") (as in appears in *Gloria slept*) as well as class 54.3 ("Fit Verbs") (as in *The cabin sleeps 5 people*). Both of these senses of *sleep* appeared in the LDOCE (the first with the code 'I' and the second with the code 'T1'), yet the algorithm collapsed these two into the same code pattern 'I T1.' Moreover, the algorithm was unable to determine whether there was any connection between such verbs and other verbs that had this code pattern as a subpattern (e.g., *nap*, which also occurs in class 40.4 but has the pattern 'I T1 N'). This was noted as a shortcoming in (Dorr *et al.*, 1995); however, our own techniques described below largely eliminate this discrepancy. In fact, if the LDOCE codes had previously been separated into groups for independent word meanings, we would still be faced with the difficulty of cross-mapping the system distinctions of LDOCE with those of WordNet.

The remainder of this paper describes the formulation and refinement of a semantic filter that increases the precision of this earlier experiment—raising the ratio of correct assignments with respect to the total number of remaining assignments—and that also extends the coverage of the algorithm to novel verbs (i.e., ones not occurring in (Levin, 1993)).

3. Semantic Filter: Increasing Precision

We take as our starting point 7767 LDOCE verbs, of which 2813 do not occur in Levin's classes. As we saw above, the syntactic filter discovers 2607 of the 3851 assignments of these verbs to Levin's semantic classes. Each of the 7767 verbs was assigned up to 15 possible semantic classes ranked by the degree of likelihood that the verb belongs to that class, giving a total of 113,106 ranked assignments.[7] Our goal is to build a semantic filter that we apply to the 113,106 ranked assignments; of the resulting assignments, we will focus specifically on the 2607 we know to be correct in order to verify the accuracy of our algorithm.

To create a semantic filter, we take a semantic class from Levin and extend it with related verbs from WordNet. We call this extended list a *semantic field*. Verbs that do not occur in the semantic field of a particular class fail to pass through the semantic filter for that class, by definition. We first examined different semantic relations provided by WordNet (synonymy, hyponymy, and synonyms of synonyms) in order to determine which

[7]Several verbs were assigned to fewer than 15 classes, thus yielding only 113,106 (not 116,505) total assignments.

one would be most appropriate for constructing semantic fields for each of Levin's 191 verb classes.[8] The WordNet word senses are quite fine-grained; we felt they were by and large adequate for assigning senses to the Levin verbs. We evaluated the performance of these different relations by examining the degree of class coverage of the relation using a prototypical verb from each class.[9]

For example, the Change of State verbs of the *break* subclass (Class 45.1) contains the verbs *break, chip, crack, crash, crush, fracture, rip, shatter, smash, snap, splinter, split, tear*. The full semantic field contains the union of the related verbs for every verb in the original Levin class. Thus, if we build our semantic field on the basis of the synonymy relation, all synonyms of verbs in a particular class would be legal candidates for membership in that class. For Class 45.1, using the synonymy relation would result in a field size of 185 (i.e., there are 185 WordNet synonyms for the 13 verbs in the class); by contrast, the hyponymy relation would yield a field size of 245.

To choose a relation to use for the semantic field, we looked at verbs semantically related to the prototypical verb in each class, and checked how many of the verbs in each class would be included in the filter. We picked the relation that matched the greatest number of Levin verbs without creating sets of verbs that were *too large*, i.e., significantly larger than the mean class size, 21.9. Larger sets of verbs would accidentally match more verbs that do not belong in that class. As can be seen in Table 3, synonyms of the prototype verb match an average of 20% of the Levin verbs, while having an average size of 10.98 verbs. If synonyms of synonyms are included, then we get an average of 25% of the verbs, but the average size of these sets jumps to 66.76. The diminishing returns continue if we include synonyms of these verbs, yielding 33% of the original verbs but having an average size of 355.06. With the hyponymy relation, the mean ratio of class coverage is lower than any of the synonymy cases. Since the synonymy relation retrieves the greatest percentage of Levin's verbs with the smallest field size, we constructed the semantic filter on the basis of the synonymy relation.

Let us now look at the behavior of the synonymy-based semantic filter. Of the 113,106 assignments of LDOCE verbs to Levin classes given by the syntactic filter, 6029 (19%) pass through the semantic filter. Clearly, the semantic filter constrains the possible assignments, but the question to ask is whether the constraint improves the accuracy of the assignments. To

[8] Levin's semantic classes are labeled with numbers ranging from 9 to 57; the actual number of semantic classes is 191 (not 46) due to many class subdivisions under each major class.

[9] A verb is considered to be *prototypical* with respect to a class if it conforms to all of Levin's membership tests for that class. These tests are based on grammaticality of usage in certain well-defined contexts (e.g., the dative construction).

Mean Value	*syn1*	*syn2*	*syn3*	*hyp*
Ratio of Coverage	0.20	0.25	0.33	.17
Number Related Verbs	10.98	66.76	355.06	13.47

syn1 = synonymy; *syn2* = synonyms of synonyms;
syn3 = synonyms of synonyms of synonyms; *hyp* = hyponymy

TABLE 3. Comparison of WordNet Relations for Semantic Fields

answer this, we first examined the 2813 verbs in LDOCE that also appear in Levin to see if they matched Levin's categorization.

Table 4 shows the settings of a, b, c, d, and n (from Table 1) in the semantic-filter approach.

	Yes is Correct	No is Correct	
Decides Yes	2607	1561	4168
Decides No	1244	36397	37641
	3851	37958	41809

TABLE 4. Results of Syntactic-Semantic Filter

Without the semantic filter, the syntactic filter provides up to 15 semantic-class assignments for each of the 2813 verbs, giving 40,248 assignments, as shown in Table 5. 2607 of these assignments (6.5%) are correct. When we add the semantic filter, the number of assignments drops to 4168, 10% of the unfiltered assignments. 2607 of these (62.5%) are correct, a ten-fold improvement over the unfiltered assignments.[10]

[10] Note that the semantic filter is primarily designed to remove incorrect assignments. Thus, in the current experiment—where the Levin verbs themselves are included in the semantic field—the filter does not affect the number of correct assignments. As we will see in later experiments on novel verbs, the semantic filter may erroneously remove some correct assignments, but will still induce a gain in precision by removing a much larger percentage of incorrect assignments.

	All	Filtered
Total Assignments	40,248	4,168
Right Assignments	2,607	2,607
Wrong Assignments	37,641	1,561
Precision (Right/Total)	6.5%	62.5%
Recall (Right/Possible)	68%	68%

TABLE 5. Increasing Precision with the Semantic Filter

It is important to point out that even though the semantic filter is, in a sense, "seeded" by words in Levin, it still sometimes categorized the Levin verb incorrectly. Since the filter is based on synonyms of Levin verbs, in some cases, a synonym of a verb from some other class will appear in the set that does not belong there. In this case, there are 1561 assignments known to be wrong, out of a total of 4168 assignments, which is 37%. For example, the verb *scatter* is a synonym of *break* in WordNet. Because the verb *break* occurs in each of these classes, the semantic filter based on synonyms assigns *scatter* to classes 10.6 (*Cheat* Verbs), 23.2 (*Split* Verbs), 40.8.3 (*Hurt* Verbs), 45.1 (*Break* Verbs), 48.1.1 (*Appear* Verbs). But the correct class for *scatter* is 9.7 (*Spray/Load* Verbs). This illustrates the difficulty of using an approach that does not account for polysemous word senses. We will address this point further in section 5.

Setting aside the polysemy problem, we see that this semantic filter is very useful for reducing the number of incorrect assignments.

4. Performance on Novel Words

We now examine how well our approach performs on unknown words by constructing a semantic filter based on three different proportions of the 2607 class assignments: (a) 50%, (b) 70%, and (c) 90%, chosen randomly.[11] We then checked whether the "unknown" verbs (those not used to construct the semantic filter) were assigned to their correct classes.

Table 6, shown on page 11, summarizes the recall and precision results for semantic filtering on these three different proportions of Levin verbs.

[11] For this experiment, we chose randomly selected subsets: First we selected a random 90% of the Levin verbs, then we chose 77.7% of those to give 70% of the Levin verbs for the 70% study, then we chose 71.4% of those to give the verbs for the 50% study.

Precision reflects the number of correct assignments divided by the total number of assignments, using the semantic filter. Recall reflects the number of correct assignments made by the semantic filter divided by the number of correct assignments made by the original syntactic assignment.[12]

Consider the rows of Table 6 that show the behavior of the experiment which uses 50% of Levin's verbs, and tries to guess the remaining verbs using synonymy. Recall that there are 2607 verbs all together. In this case, 1282 verbs were chosen at random to use in constructing the filter. We call these the "known" verbs. This leaves 1325 for use in evaluating the semantic filter—we call these the "novel" verbs. For the 1282 known verbs, the filter made 1752 assignments to semantic classes. There were 470 wrong assignments and 1282 right ones, giving a precision rate of 73.2% and recall rate of 100.0%. The results based on the other percentages are all shown in Table 6.

There are two notable points about this table: (1) the precision for known words decreases as the number of known word assignments increases; and (2) the recall for novel words increases as the number of novel word assignments decreases. The reason for (1) is that, as the number of known word assignments increases, the size of the semantic filter increases, i.e., there is a greater probability that incorrect assignments will pass through the filter. Similarly, the reason for (2) is that, as the number of novel word assignments decreases, the size of the semantic field increases, i.e., there is a greater probability that a novel word will be related to a verb in the field and hence pass through the semantic filter.

As a preliminary experiment, we ran the semantic filter alone, i.e., decoupled from the syntactic filter, to see how well our semantic filter performed without aid from syntax. For the semantic filter based on disambiguated synonyms, the recall was about 48% higher, but the precision was only 69% as high as with the syntactic filter. For disambiguated cohyponyms (i.e., hyponyms of hypernyms—going up one level in abstraction, and then one level back down), recall was about 43% higher but precision only a third as high.[13] These preliminary results showed that the semantic filter alone produces markedly worse results. Thus, it appears that it is the

[12]In contrast to the preceding section, we take the number of possible correct assignments (i.e., the value corresponding to "Yes is correct" in Table 1) to be 2607. We use this "normalized" value rather than the total number of correct assignments (3851) because our goal is to examine the relative degree of improvement in precision of the semantic filter *alone*—i.e., *after* syntactic assignment has taken place—for different percentages of "novel" class assignments. The side effect of using this normalized figure is that the recall for "known" class assignments remains fixed at 100.0% since the number of correct guesses is the same as the number of original assignments in each case.

[13]An example of a WordNet hypernym for the verb *chide* is *criticize*. This verb has, in turn, the hyponym *berate*. Thus, *berate* is a cohyponym of the verb *chide*.

combination of the two filters, not either of the two alone, that produces the best classification results, given our main goal of increasing precision.

Semantic-Filter Assignments to Levin Classes

% Levin	*Original Assignments*		*Number of Guesses*			*Ratios*	
			Total	Wrong	Right	Precision	Recall
50%	known	1282	1752	470	1282	73.2%	100.0%
	novel	1325	841	429	412	49.0%	31.1%
70%	known	1798	2628	830	1798	68.4%	100.0%
	novel	809	663	360	303	45.7%	37.5%
90%	known	2341	3632	1291	2341	64.5%	100.0%
	novel	266	271	158	113	41.7%	42.5%
100%	all known	2607	4168	1561	2607	62.5%	100.0%

Original Syntactic-Filter Assignments to Levin Classes

% Levin	*Original Assignments*		*Number of Assignments*			*Ratios*	
			Total	Wrong	Right	Precision	Recall
100%	Known	2607	40248	37641	2607	6.5%	100%

TABLE 6. Undisambiguated Synonyms

5. Improved Semantic Filter: Accounting for Polysemy

As mentioned previously, the problem with the semantic filter we have defined is that it is not sensitive to polysemous word senses of the particular verbs in the semantic classes. For example, there are 23 senses of the verb *break* in WordNet. This includes senses which correspond to the Change of State verbs, such as Sense 9, "break, bust, cause to break", the synonyms of which are *destroy, ruin, bust up, wreck, wrack*. But it also includes irrelevant senses, such as Sense 7, "break dance", the synonyms of which are *dance, do a dance, perform a dance*. Clearly, the semantic filter would behave better if we used word senses in creating the fields.

As an attempt to address the polysemy problem, we conducted an exploratory study in which the verbs in Levin's semantic classes were disambiguated by hand: each verb received as many WordNet senses as were applicable. Appendix A provides the details behind this hand disambiguation process, which took a total of 1 month. We then re-generated WordNet-based semantic fields for each class, using only the word senses that were deemed relevant (through hand-disambiguation) to that class. We re-applied the semantic filter (coupled with the syntactic filter, as before) using these new semantic fields.

The performance of the various filters is shown in Table 7. To see the effect of disambiguation, compare the difference between undisambiguated and disambiguated synonyms. Precision has increased from 62.5% to 85.3%. For novel verbs, in the experiment which uses 50% of the verbs and tries to guess the rest, the precision increases from 49.0% to 70.8%. But note also that the recall decreases: with disambiguation (in the 50% study), recall drops from 31.1% for undisambiguated verbs to 21.6% for disambiguated verbs.

Table 7 also shows the performance of two other semantic filters based on hyponyms. We found that using cohyponyms gave much better recall than plain synonymy, although the precision is lower.[14] We also built a filter based on the union of synonyms with cohyponyms. The effect of the synonyms on this filter was negligible, presumably since synonyms are often cohyponyms. The results for both of these filters are shown in Table 7. The overall result is that the synonymy relation gives higher precision, whereas the cohyponym relation gives higher recall.

We applied our approach to 5000 LDOCE verbs that are entirely outside of (Levin, 1993). An example of a semantic class that we augmented with verbs from this set using the semantic-filter approach is class 33 *Judgment Verbs*. Table 8 shows newly-classified verbs for this class, based on the (broad) filter using undisambiguated synonyms as well as the (improved) filter using disambiguated synonyms.

It is interesting to note that less than 2000 assignments were made for the 5000 LDOCE verbs not included in Levin. The primary factor in this low number of assignments is that the syntactic information associated

[14] This is because synonyms are in a 'sibling' relationship to the synset of the initial verb, whereas cohyponyms are in a 'cousin' relationship to the synset of the initial verb, i.e., a much broader-ranging set of related words that typically subsumes the verb's synonyms. For example, the verb *weaken* has 11 synonyms, but a total of 896 cohyponyms (including *weaken*). Many of these verbs are so highly specific that they are significantly distinct from what one might think of as the meaning of the verb *weaken*: *Westernize*, *become unfashionable*, *change flavor*, *complicate*, *obtain*, *redeem*, *scorn*, etc. Using this compound relation as a semantic filter would induce at least as many correct assignments as the synonymy relation (maintaining or increasing recall), while also inducing many more incorrect assignments (lowering precision).

Undisambiguated Synonyms				
%	*Known*		*Novel*	
Levin	Recall	Precision	Recall	Precision
100%	100.0%	62.5%	n.a.	n.a.
90%	100.0%	64.5%	42.5%	41.7%
70%	100.0%	68.4%	37.5%	45.7%
50%	100.0%	73.2%	31.1%	49.0%

Disambiguated Synonyms				
%	*Known*		*Novel*	
Levin	Recall	Precision	Recall	Precision
100%	100.0%	85.3%	n.a.	n.a.
90%	100.0%	86.2%	29.3%	63.9%
70%	100.0%	88.3%	26.1%	68.5%
50%	100.0%	91.7%	21.6%	70.8%

Disambiguated Cohyponyms				
%	*Known*		*Novel*	
Levin	Recall	Precision	Recall	Precision
100%	100.0%	37.7%	n.a.	n.a.
90%	100.0%	39.0%	68.8%	29.5%
70%	100.0%	41.5%	63.0%	31.1%
50%	100.0%	45.8%	58.6%	34.6%

Union of Disambiguated Synonyms with Cohyponyms				
%	*Known*		*Novel*	
Levin	Recall	Precision	Recall	Precision
100%	100.0%	37.6%	n.a.	n.a.
90%	100.0%	38.9%	69.5%	29.7%
70%	100.0%	41.4%	64.4%	31.5%
50%	100.0%	45.8%	59.6%	34.9%

TABLE 7. Comparison of Filters

with verbs outside of Levin was frequently not correlated with any of the semantic classes—primarily because Levin's classes were not exhaustive. The syntactic filter failed to find an appropriate class in such cases, and

33. Judgment Verbs: abuse, acclaim, applaud, backbite, bless, calumniate, castigate, celebrate, censure, chasten, chastise, chide, commend, compensate, compliment, condemn, congratulate, criticize, decry, defame, denigrate, denounce, deprecate, deride, disparage, eulogize, excuse, extol, fault, felicitate, ...
New Words Classified in 33 Using Broad Filter (30 new verbs): abide-by, answer-back, bawl-out, belittle, berate, besmirch, call-for, chew-out, condone, derogate, drink-to, excoriate, ignore, justify, laugh-at, lionize, look-down-on, maltreat, minimize, oppress, pay-for, pick-at, reinforce, sanctify, spurn, subdue, sully, tell-on, traduce, turn-down
New Words Classified in 33 Using Improved Filter (7 new verbs): berate, besmirch, evoke, maltreat, spurn, sully, turn-down

TABLE 8. Classification of Unknown Verbs into Judgment Class: Comparison of Two Kinds of Semantic Filter

the semantic filter subsequently had no classes from which to choose. For example, the non-Levin verb *absorb* has the syntactic pattern 'T1 WV5 WV5-BY WV5-IN WV5-INTO', which was not associated with any of the existing semantic classes in the experiment. The main point here is that, in order to classify a broader range of verbs, we would need to investigate the possibility that additional classes exist *outside* of those in Levin. An investigation along these lines is described in (Dorr, 1997), where syntactic codes such as the ones above are associated with 26 new semantic classes. We expect that the incorporation of these results into future experiments will prove fruitful.

6. Conclusions

Our goal throughout the acquisition task is to eliminate as many incorrect assignments as possible while preserving the correct assignments, and in this respect we are encouraged by the behavior of the semantic filter on "unknown" verbs. Recall that to assess this behavior, we excluded randomly selected Levin verbs from the semantic filter, and saw how the filter behaved on these verbs. Our main result is that the semantic field substantially reduces the number of incorrect assignments given by the syntactic filter.

We used our approach to assign new verbs, i.e., all of the verbs in LDOCE, to the semantic classes of Levin. Since there are 7767 verbs in LDOCE, and there are 191 semantic classes in Levin, there are 1,483,497 potential assignments of verbs to these semantic classes. The syntactic filter reduces the number of assignments under consideration to 113,106 (7.6% of the number of potential assignments) while preserving 67% of the assignments we know to be correct. The various semantic filters in turn reduce

the number of assignments further. For example, the broad semantic filter reduced the 113,106 verbs that passed through the syntactic filter down to 6029 assignments, 19% of the number of assignments based on syntax and 0.4% of the potential assignments.

Acknowledgements

This research was conducted in the Computational Linguistics and Information Processing (CLIP) Laboratory at the University of Maryland (UM), the Linguistics Department at the Universidad de Barcelona (UB), and the Computer Science Department at the Universidad Politécnica de Cataluña (UPC). The work was supported, in part, by National Science Foundation Presidential Faculty Fellowship (PFF/PECASE) Award IRI-9629108, Alfred P. Sloan Research Fellow Award BR3336, Department of Defense contract MDA90496C1250, DARPA/ITO Contract N66001-97-C-8540, Army Research Laboratory contract LETTER11097 through United Research Corporation, and Army Research Laboratory contract DAAL03-91-C-0034 through Battelle. We would like to thank Julie Dahmer, Charles Lin, and David Woodard for their help in annotating the verbs. We would also like to thank Karen Kohl for permission to use her WordNet annotations for Part One of Levin's book as hints for WordNet senses for Part Two.

A. Details of Levin Disambiguation

The 191 semantic classes of Levin were broken into 17 segments. One person annotated every segment in the suite, and the two other people each annotated half, so every segment was annotated by two people.[15] They were instructed to mark the best WordNet sense with a `1`. If there was no "best" match, then they were to mark the next best ones with a `2`.[16] Table 9 shows how many senses were judged relevant for each class. For example, the first row shows how person **a** disambiguated the verbs. For the first segment, there were 3 verb classes which had a total of 182 WordNet senses. 13 of them were marked as the "Best" match. 2 senses were only ranked as "Good". These, added with the 13 best matches, give the 8% of the senses we used for the filters.

As might be predicted, the combination of the 3 coders' results involved resolution of inter-coder variability. Our approach to combining the conflicting results was biased toward including senses rather than excluding

[15] There are 16 major segments—`seg00` was used as a pilot case for the remaining work.

[16] If no senses were applicable, they gave a `0` mark. There were very few of these, and we did not use these marks for our filters.

them. Even with this relatively permissive methodology, the majority of the WordNet senses were eliminated, which we took as an indication that further fine-tuning of the filter was not necessary.

Specifically, we handled the inter-coder variability by simply combining the results of all positive votes across all three coders and then eliminating any conflicting negative votes. That is, if coder A voted for sense 1 for a given verb and coder B voted for sense 2 for that verb, then the result was that both sense 1 and sense 2 were taken to be valid senses. Also, if one coder voted for a particular verb sense (i.e., marked with `1`) that another coder voted against (i.e., marked with `0`), we eliminated the case marked with `0`. For example, one coder marked verb sense 1 for *anoint* (in Levin's verb class 29.3) with a `1` and another marked it with a `0`. The final result includes this sense—*oil, anele, ambrocate*—for the verb *anoint*. Finally, we conflated the positive votes (i.e., the `1`'s) and the questionable votes (i.e., the `2`'s), so that any hint of a match was counted as a full match (i.e., only the `1` mark was counted).

Segment	Classes	Senses	Best	% Best	# Good	% ≥ Good	Person
seg00	3	182	13	7%	2	8%	a
seg00	3	182	8	4%	12	11%	b
seg00	3	182	13	7%	4	9%	c
seg01	7	679	67	10%	6	11%	a
seg01	7	679	124	18%	12	20%	c
seg02	10	641	113	18%	36	23%	b
seg02	10	641	101	16%	7	17%	c
seg03	11	743	112	15%	11	17%	b
seg03	11	743	124	17%	5	17%	c
seg04	17	982	134	14%	3	14%	a
seg04	17	982	206	21%	2	21%	c
seg05	12	706	71	10%	3	10%	a
seg05	12	706	149	21%	2	21%	c
seg06	11	853	119	14%	20	16%	b
seg06	11	853	135	16%	8	17%	c
seg07	10	759	179	24%	17	26%	b
seg07	10	759	148	19%	12	21%	c
seg08	5	687	121	18%	7	19%	a
seg08	5	687	153	22%	10	24%	c
seg09	17	722	129	18%	3	18%	a
seg09	16	722	164	23%	9	24%	c
seg10	21	654	241	37%	7	38%	b
seg10	21	654	16	2%	1	3%	c
seg11	20	829	179	22%	10	23%	b
seg11	20	829	142	17%	6	18%	c
seg12	7	461	84	18%	3	19%	a
seg12	7	461	195	42%	6	44%	c
seg13	6	967	153	16%	5	16%	a
seg13	6	967	360	37%	10	38%	c
seg14	9	896	156	17%	29	21%	b
seg14	9	896	171	19%	17	21%	c
seg15	10	799	139	17%	13	19%	b
seg15	10	799	156	20%	4	20%	c
seg16	16	699	122	17%	6	18%	a
seg16	16	699	171	24%	2	25%	c

TABLE 9. Survey of the WordNet Annotations

References

H. Alshawi. Analysing the Dictionary Definitions. In B. Boguraev and T. Briscoe, editors, *Computational Lexicography for Natural Language Processing*, pages 153–169. Longman, London, 1989.

Branimir Boguraev and Ted Briscoe. Utilising the LDOCE Grammar Codes. In Branimir Boguraev and Ted Briscoe, editors, *Computational Lexicography for Natural Language Processing*, pages 85–116. Longman, London, 1989.

Michael Brent. Unsupervised Learning of Lexical Syntax. *Computational Linguistics*, 19:243–262, 1993.

Kenneth Church and P. Hanks. Word Association Norms, Mutual Information and Lexicography. *Computational Linguistics*, 16:22–29, 1990.

Ann Copestake, Ted Briscoe, P. Vossen, A. Ageno, I. Castellon, F. Ribas, G. Rigau, H. Rodríguez, and A. Samiotou. Acquisition of Lexical Translation Relations from MRDS. *Machine Translation*, 9:183–219, 1995.

Bonnie J. Dorr and Douglas Jones. Role of Word Sense Disambiguation in Lexical Acquisition: Predicting Semantics from Syntactic Cues. In *Proceedings of the International Conference on Computational Linguistics*, pages 322–333, Copenhagen, Denmark, 1996.

Bonnie J. Dorr, Joseph Garman, and Amy Weinberg. From Syntactic Encodings to Thematic Roles: Building Lexical Entries for Interlingual MT. *Machine Translation*, 9:71–100, 1995.

Bonnie J. Dorr. Large-Scale Acquisition of LCS-Based Lexicons for Foreign Language Tutoring. In *Proceedings of the ACL Fifth Conference on Applied Natural Language Processing (ANLP)*, pages 139–146, Washington, DC, 1997.

David Farwell, Louise Guthrie, and Yorick Wilks. Automatically Creating Lexical Entries for ULTRA, a Multilingual MT System. *Machine Translation*, 8(3):127–145, 1993.

Charles Fillmore. The Case for Case. In E. Bach and R. Harms, editors, *Universals in Linguistic Theory*, pages 1–88. Holt, Rinehart, and Winston, 1968.

Jane Grimshaw. *Argument Structure*. The MIT Press, Cambridge, MA, 1990.

Jeffrey S. Gruber. *Studies in Lexical Relations*. PhD thesis, MIT, Cambridge, MA, 1965.

J. Guthrie, L. Guthrie, Y. Wilks, and H. Aidinejad. Subject-Dependent Co-occurrence and Word Sense Disambiguation. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pages 146–152, University of California, Berkeley, CA, 1991.

M. Hearst. Noun Homograph Disambiguation Using Local Context in Large Text Corpora. In *Using Corpora*, University of Waterloo, Waterloo, Ontario, 1991.

Ray Jackendoff. *Semantics and Cognition*. The MIT Press, Cambridge, MA, 1983.

Ray Jackendoff. *Semantic Structures*. The MIT Press, Cambridge, MA, 1990.

Judith L. Klavans and Evelynne Tzoukermann. Dictionaries and Corpora: Combining Corpus and Machine-Readable Dictionary Data for Building Bilingual Lexicons. *Machine Translation*, 10:185–218, 1995.

Beth Levin. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago, IL, 1993.

David Dolan Lewis. *Representation and Learning in Information Retrieval*. PhD thesis, University of Massachusetts, Amherst, 1992.

Deryle Lonsdale, Teruko Mitamura, and Eric Nyberg. Acquisition of Large Lexicons for Practical Knowledge-Based MT. *Machine Translation*, 9:251–283, 1995.

George A. Miller and Christiane Fellbaum. Semantic Networks of English. In Beth Levin and Steven Pinker, editors, *Lexical and Conceptual Semantics, Cognition Special Issue*, pages 197–229. Elsevier Science Publishers, B.V., Amsterdam, The Netherlands, 1991.

George A. Miller. Dictionaries in the Mind. *Language and Cognitive Processes*, 1:171–185, 1986.

George A. Miller. WordNet: An On-Line Lexical Database. *International Journal of*

Lexicography, 3:235–312, 1990.

Mary Neff and Michael McCord. Acquiring Lexical Data from Machine-Readable Dictionary Resources for Machine Translation. In *Third International Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages (TMI-90)*, pages 85–90, Austin, Texas, 1990.

Steven Pinker. *Learnability and Cognition: The Acquisition of Argument Structure*. The MIT Press, Cambridge, MA, 1989.

P. Procter. *Longman Dictionary of Contemporary English*. Longman, London, 1978.

Antonio Sanfilippo and V. Poznanski. The Acquisition of Lexical Knowledge from Combined Machine-Readable Dictionary Resources. In *Proceedings of the Applied Natural Language Processing Conference*, pages 80–87, Trento, Italy, 1992.

Donald Walker and Robert Amsler. The Use of Machine-readable Dictionaries in Sublanguage Analysis. In R. Grishman and R. Kittredge, editors, *Analyzing Language in Restricted Domains*, pages 69–83. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1986.

Yorick Wilks, Dan Fass, Cheng-Ming Guo, James E. McDonald, Tony Plate, and Brian M. Slator. A Tractable Machine Dictionary as a Resource for Computational Semantics. In Branimir Boguraev and Ted Briscoe, editors, *Computational Lexicography for Natural Language Processing*, pages 193–228. Longman, London, 1989.

Yorick Wilks, Dan Fass, Cheng-Ming Guo, James E. McDonald, and Tony Plate. Providing Machine Tractable Dictionary Tools. *Machine Translation*, 5(2):99–154, 1990.

D. Wu and X. Xia. Large-Scale Automatic Extraction of an English-Chinese Translation Lexicon. *Machine Translation*, 9:285–313, 1995.

D. Yarowsky. Word-Sense Disambiguation: Using Statistical Models of Roget's Categories Trained on Large Corpora. In *Proceedings of the Fourteenth International Conference on Computational Linguistics*, pages 454–460, Nantes, France, 1992.