

Acquiring Knowledge from the Web to be used as Selectors for Noun Sense Disambiguation

Hansen A. Schwartz and Fernando Gomez

School of Electrical Engineering and Computer Science

University of Central Florida

{hschwartz, gomez}@cs.ucf.edu

Abstract

This paper presents a method of acquiring knowledge from the Web for noun sense disambiguation. Words, called selectors, are acquired which take the place of an instance of a target word in its local context. The selectors serve for the system to essentially learn the areas or concepts of WordNet that the sense of a target word should be a part of. The correct sense is chosen based on a combination of the strength given from similarity and relatedness measures over WordNet and the probability of a selector occurring within the local context. Our method is evaluated using the coarse-grained all-words task from SemEval 2007. Experiments reveal that path-based similarity measures perform just as well as information content similarity measures within our system. Overall, the results show our system is out-performed only by systems utilizing training data or substantially more annotated data.

1 Introduction

Recently, the Web has become the focus for many word sense disambiguation (WSD) systems. Due to the limited amount of sense tagged data available for supervised approaches, systems which are typically referred to as unsupervised, have turned to the use of unannotated corpora including the Web. The advantage of these systems is that they can disambiguate all words, and not just a set of words for which training data has been provided. In this paper we present an unsupervised system which uses the Web in a novel fashion to perform

sense disambiguation of any noun, incorporating both similarity and relatedness measures.

As explained in (Brody et al., 2006), there are generally two approaches to unsupervised WSD. The first is referred to as *token* based, which compares the relatedness of a target word to other words in its context. The second approach is *type* based, which uses or identifies the most common sense of a word over a discourse or corpus, and annotates all instances of a word with the most common sense. Although the *type* based approach is clearly bound to fail occasionally, it is commonly found to produce the strongest results, rivaling supervised systems (McCarthy et al., 2004). We identify a third approach through the use of *selectors*, first introduced by (Lin, 1997), which help to disambiguate a word by comparing it to other words that may replace it within the same local context.

We approach the problem of word sense disambiguation through a relatively straightforward method that incorporates ideas from the *token*, *type*, and *selector* approaches. In particular, we expand the use of *selectors* in several ways. First, we revise the method for acquiring selectors to be applicable to the web, a corpus that is, practically speaking, impossible to parse in whole. Second, we describe a path-based similarity measure that is more suited for a portion of our method than the relatedness measures used by *token* based systems. Finally, we expand the use of selectors to help with disambiguating nouns other than the one replaced.

2 Background

2.1 Word Sense Disambiguation

A popular approach to using the web or unannotated corpora for word sense disambiguation involves the use of monosemous relatives. Monosemous relatives are words which are similar to a

© 2008. Licensed under the *Creative Commons Attribution-Noncommercial-Share Alike 3.0 Unported* license (<http://creativecommons.org/licenses/by-nc-sa/3.0/>). Some rights reserved.

sense of the target word, but which only have one sense. By searching text for these words, one can build training data for each sense of a target word. This idea was proposed by (Leacock et al., 1998). More recently, the idea has been used to automatically create sense tagged corpora (Mihalcea, 2002; Agirre and Martinez, 2004). These methods queried large corpora with relatives rather than with the context.

With some resemblances to our approach, (Martinez et al., 2006) present the *relatives in context* method. A key similarity of this method with ours is the use of context in the web queries. They produce queries with relatives in place of the target word in a context with a window size of up to 6. Similarly, (Yuret, 2007) first chooses substitutes and determines a sense by looking at the probability of a substitute taking the place of the target word within the Web1T corpus. The number of hits each query has on the web is then used to pick the correct sense. Our approach differs from these in that we acquire words(selectors) from the web, and proceed to choose a sense based on similarity measures over WordNet (Miller et al., 1993). We also attempt to match the context of the entire sentence if possible, and we are more likely to receive results from longer queries by including the wildcard instead of pre-chosen relatives.

We adopted the term *selector* from (Lin, 1997) to refer to a word which takes the place of another in the same local context. Lin searched a local context database, created from dependency relationships over an unannotated corpora in order to find selectors. In this case, the local context was represented by the dependency relationships. Given that the task of producing a dependency parse database of the Web is beyond our abilities, we search for the surrounding local context as text in order to retrieve selectors for a given word. Another difference is that we compare the relatedness of selectors of other words in the sentence to the target word, and we also incorporate a path-based similarity measure along with a gloss-based relatedness measure.

2.2 Similarity and Relatedness Measures

Semantic similarity and relatedness measures have an extensive history. The measures reported in this work were included based on appropriateness with our approach and because of past success according to various evaluations (Patwardhan et al., 2003;

Budanitsky and Hirst, 2006).

Many similarity measures have been created which only use paths in the WordNet ontology. One approach is to simply compute the length of the shortest path between two concepts over the hypernym/hyponym relationship (Rada et al., 1989). Other methods attempt to compensate for the uniformity problem, the idea that some areas of the ontology are more dense than others, and thus all edges are not equal. (Wu and Palmer, 1994) uses the path length from the root to the lowest common subsumer(LCS) of two concepts scaled by the distance from the LCS to each concept. Another method, by (Leacock et al., 1998), normalizes path distance based on the depth of hierarchy. Our method attempts to produce a normalized depth based on the average depth of all concepts which are leaf nodes below the lowest common subsumer in a tree.

We employ several other measures in our system. These measures implement various ideas such as *information content* (Jiang and Conrath, 1997; Lin, 1997) and *gloss overlaps* (Banerjee and Pedersen, 2003). For our work the path-based and information content measures are referred to as *similarity measures*, while the gloss-based methods are referred to as *relatedness measures*. Relatedness measures can be used to compare words from different parts of speech. In past evaluations of token based WSD systems, information content and gloss-based measures perform better than path-based measures (Patwardhan et al., 2003; Budanitsky and Hirst, 2006).

3 Method

The general idea of our method is to find the sense of a target noun which is most similar to all selectors which can replace the target and most related to other words in context and their selectors. Our method requires that a test sentence has been part-of-speech tagged with noun, verb, and adjective POS, and we use the selectors from all of these parts of speech as well as noun selectors of pronouns and proper nouns. In this work, we only disambiguate nouns because *similarity* measures for target selectors are based heavily on the depth that is present in the WordNet noun ontology. However, we are still able to use verb and adjective selectors from the context through *relatedness* measures working over all parts of speech listed. The method can be broken into two steps:

1. Acquire probabilities of selectors occurring for all nouns, verbs, adjectives, pronouns and proper nouns from the Web.
2. Rank the senses of a target noun according to similarity with its own selectors and relatedness with other selectors in the context.

These steps are described in detail below. Finally, we also describe a similarity measure we employ.

3.1 Acquiring Selectors

We acquire *target selectors* and *context selectors* from the Web. Target selectors are those words which replace the current target word in the local context, while *context selectors* are words which may replace other words in the local context. There are four different types of context selectors:

noun context selectors essentially the target selectors for other nouns of the sentence.

verb context selectors verbs which are found to replace other verbs in the sentence.

adjective context selectors adjectives which replace other adjectives in the sentence.

pro context selectors nouns which replace pronouns and proper nouns.

A query must be created based on the original sentence and target word. This is fairly straightforward as the target word is removed and replaced with a * to indicate the wildcard. For example, when searching for selectors of “batter” from “She put the batter in the refrigerator.”, a query of “She put the * in the refrigerator.” is used. The queries are sent through the Yahoo! Search Web Services¹ in order to retrieve matching text on the web.

The selectors are extracted from the samples returned from the web by matching the wildcard of the query to the sample. The wildcard match is thrown out if any of the following conditions are true: *longer than 4 words, contains any punctuation, is composed only of pronouns or the original word*. Keep in mind we acquire the nouns that replace the pronouns of the original sentence, so a selector is never a pronoun. WordNet is used to determine if the phrase is a compound and the base morphological form of the head word. Results containing head words not found in WordNet are filtered out. Proper nouns are used if they are found in WordNet. Finally, the list of selectors is

adjusted so no single word takes up more than 30% of the list.

The Web is massive, but unfortunately it is not large enough to find results when querying with a whole sentence a majority of the time. Therefore, we perform truncation of the query to acquire more selectors. For this first work with selectors from the web, we chose to create a simple truncation focused just on syntax in order to run quickly. The steps below are followed and the final step is repeated until a stop condition is met.

- i Shorten to a size of 10 words.
- ii Remove end punctuation, if not preceded by *.
- iii Remove front punctuation, if not preceded by *.
- iv Remove determiners (*the, a, an, this, that*) preceding *.
- v Remove a single word.

When removing a single word, the algorithm attempts to keep the * in the center. Figure 1 demonstrates the loop that occurs until a stop condition is met: enough selectors are found or the query has reached a minimum size. Since a shorter query should return the same results as a longer query, we filter the selectors from longer query results out of the shorter results. It is important that the criteria to continue searching is based on the number of selectors and not on the number of samples, because many samples fail to produce a selector. Validation experiments were performed to verify that each step of truncation was helpful in returning more results with valid selectors, although the results are not reported as the focus is on the method in general. Selectors are tied to the queries used to acquire them in order to help emphasize results from longer queries.

The steps to acquire all types of selectors (target or any in context) are the same. The part of speech only plays a part in determining the base form or compounds when using WordNet. Note that all selectors for each noun, verb, adjective, and pronoun/proper can be acquired in one pass, so that duplicate queries are not sent to the Web. When the process is complete we have a probability value for each selector word (w_s) to occur in a local context given by the acquisition query (q). The probability of w_s appearing in q is denoted as:

$$p_{occ}(w_s, q)$$

3.2 Ranking Senses

There are essentially two assumptions made in order to rank the senses of a noun.

¹<http://developer.yahoo.com/search/>

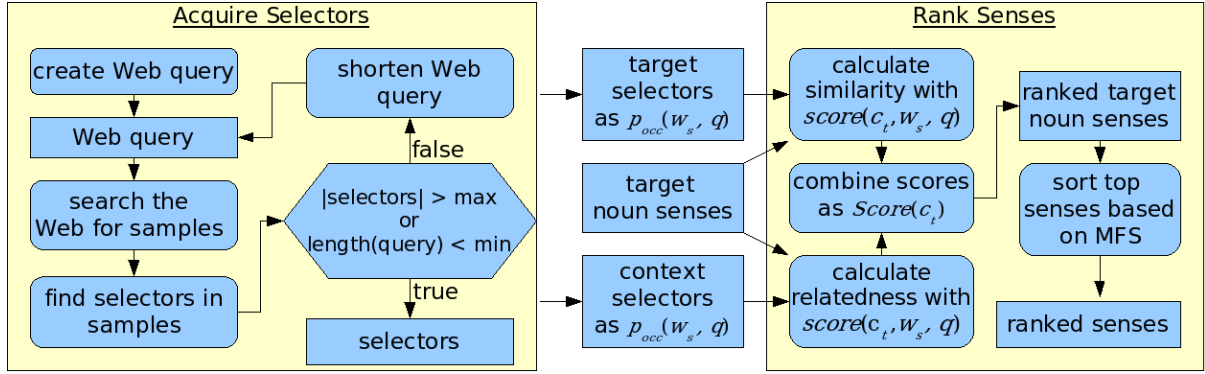


Figure 1: The overall process undertaken to disambiguate a noun. (Note that selectors only need to be acquired once for each sentence since they can be reused for each target noun.)

1. Similar concepts (or noun senses) appear in similar syntactic constructions.
2. The meaning of a word is often related to other words in its context

The first assumption implies the use of a similarity measure with *target selectors*. The meaning of the target selectors should be very similar to that of the original word, and thus we compare similarity between all target selectors with each sense of the original word.

The second assumption reflects the information provided by *context selectors*, for which we use a relatedness measure to compare with the original word. Note that because context selectors may be of a different part of speech, we should be sure this measure is able to handle multiple parts of speech.

Regardless of the similarity or relatedness measure used, the value produced is applied the same for both *target selectors* and *context selectors*. We are comparing the senses (or concepts) of the original target word with all of the selectors. To find the similarity or relatedness of two words, rather than two concepts, one can use the maximum value over all concepts of the selector word and all the senses of the target word, (Resnik, 1999, *word similarity*):

$$wsr(w_t, w_s) = \max_{c_t, c_s} [srm(c_t, c_s)]$$

where *srm* is a similarity or relatedness measure and c_t, c_s represent a sense (concept) of the target word (w_t) and selector word (w_s) respectively. We would like to get a value for each sense of a target word if possible, so we derive similarity or relatedness between one concept and one word as:

$$cwsr(c_t, w_s) = \max_{c_s} [srm(c_t, c_s)]$$

Intuitively, combining *cwsr* with p_{occ} is the basis for scoring the senses of each noun. However, we also take several other values into account, in order to learn most effectively from Web selectors. The score is scaled by the number of senses of the selector and the length of the query used to acquire it. This gives less ambiguous selectors and those selectors with a most similar local context a stronger role. These values are represented by $senses(w_s)$ and $qweight = \frac{current_length}{original_length}$:

$$\begin{aligned} score(c_t, w_s, q) \\ = p_{occ}(w_s, q) * cwsr(c_t, w_s) * \frac{qweight}{senses(w_s)} \end{aligned}$$

The scores are summed with:

$$sum_{type}(c_t) = \sum_q \sum_{w_s} score(c_t, w_s, q)$$

where q ranges over all queries for a type (*type*) of selector, and w_s ranges over all selectors acquired with query q .

Overall, the algorithm gives a score to each sense by combining the normalized sums from all types of the selectors:

$$Score(c_t) = \sum_{type} \frac{sum_{type}(c_t)}{\max_{c \in w_t} [sum_{type}(c)]} * scale_{type}$$

where *typ* ranges over a type of selector (target, noun context, verb context, adjective context, pro context), c ranges over all senses of the target word (w_t), and $scale_{type}$ is a constant for each type of selector. We experimented with different values over 60 instances of the corpus to decide on a scale value of 1 for *target selectors*, a value of 0.5 for

noun and verb *context selectors*, and a value of 0.1 for adjective and pro *context selectors*. This weights the scores that come from target selectors equal to that of noun and verb context selectors, while the adjective and pro selectors only play a small part.

Finally, the senses are sorted based on their *Score*, and we implement the most frequent sense heuristic as a backoff strategy. All those senses within 5% of the top sense’s *Score*, are re-sorted, ranking those with lower sense numbers in WordNet higher. The highest ranking sense is taken to be the predicted sense.

3.3 Similarity Measure

We use the notion that similarity is a specific type of relatedness (Rada et al., 1989; Patwardhan et al., 2003). For our purposes, a *similarity* measure is used for nouns which may take the place of a target word within its local context, while words which commonly appear in other parts of the local context are measured by *relatedness*. In particular, the *similarity* measure places emphasis strictly on the *is-a* relationship. As an example, “bottle” and “water” are *related* but not *similar*, while “cup” and “bottle” are *similar*. Because of this distinction, we would classify our path-based measure as a *similarity* measure.

A well known problem with path-based measures is the assumption that the links between concepts are all uniform (Resnik, 1999). As a response to this problem, approaches based on information content are used, such as (Resnik, 1999; Jiang and Conrath, 1997; Lin, 1997). These measures still use the *is-a* relationship in WordNet, but they do not rely directly on edges to determine the strength of a relationship between concepts. (Patwardhan et al., 2003) shows that measures based on information content or even gloss based measures generally perform best for comparing a word with other words in its context for word sense disambiguation. However, these measures may not be as suited for relating one word to other words which may replace it (*target selectors*). Therefore, our similarity measure examines the use of links in WordNet, and attempts to deal with the uniformity problem by normalizing depths based on average leaf node depth.

All types of relatedness measures return a value representing the strength of the relation between the two concepts. These values usually range be-

tween 0 and 1. Note that concepts are not the same as words, and the example above assumes one chooses the sense of “water” as a liquid and the sense of “bottle” and “cup” as a container. Our similarity measure is based on finding the normalized depth (*nd*) of a concept (*c*) in the WordNet Hierarchy:

$$nd(c) = \frac{depth(c)}{ald(c)}$$

Where *depth* is the length from the concept to the root, and *ald* returns the average depth of all descendants (hyponyms) that do not have hyponyms themselves (average leaf depth):

$$ald(c) = \frac{\sum_{L \in lnodes(c)} depth(l)}{|lnodes(c)|}$$

To be clear, *lnodes* returns a list of only those nodes without hyponyms that are themselves hyponyms of *c*. We chose to only use the leaf depth as opposed to all depths of descendants, because *ald* produces a value representing maximum depth for that branch in the tree, which is more appropriate for normalization.

Like other similarity measures, for any two concepts we compute the lowest (or deepest) common subsumer, *lcs*, which is the deepest node in the hierarchy which is a hypernym of both concepts. The similarity between two concepts is then given by the normalized depth of their *lcs*:

$$sim(c_1, c_2) = nd(lcs(c_1, c_2))$$

Thus, a concept compared to itself will have a score of 1, while the most dissimilar concepts will have a score of 0. Following (Wu and Palmer, 1994; Lin, 1997) we scale the measure by each concept’s *nd* as follows:

$$scaled_sim(c_1, c_2) = \frac{2 * sim(c_1, c_2)}{nd(c_1) + nd(c_2)}$$

where our *normalized depth* replaces the *depth* or *information content* value used by the past work.

4 Evaluation

We evaluated our algorithm using the SemEval 2007 coarse-grained all-words task. In order to achieve a coarse grained sense inventory WordNet 2.1 senses were manually mapped to the top-level of the Oxford Dictionary of English by an expert lexicographer. This task avoids the issues of a fine granular sense inventory, which provides senses

type	insts	avgSels
target	1108	68.5
noun context	1108	68.5
verb context	591	70.1
adj context	362	37.3
pro context	372	31.9

Table 1: Total word instances for which selectors were acquired (**insts**), and average number of selectors acquired for use in each instance (**avgSels**).

that are difficult even for humans to distinguish. Additionally, considering how recent the event occurred, there is a lot of up-to-date data about the performance of other disambiguation systems to compare with. (Navigli et al., 2007)

Out of 2269 noun, verb, adjective, or adverb instances we are concerned with disambiguating the 1108 noun instances from the 245 sentences in the corpus. These noun instances represent 593 different words. Since we did not use the coarse-grained senses within our algorithm, the predicted senses were correct if they mapped to the correct coarse-grained sense. The average instance had 2.5 possible coarse-grained senses. The average number of selectors acquired for each word is given in Table 1. The bottom of Table 2 shows the random baseline as well as a baseline using the most frequent sense (MFS) heuristic. As previously mentioned, many supervised systems only perform marginally better than the MFS. For the SemEval workshop, only 6 of 15 systems performed better than this baseline on the nouns (Navigli et al., 2007), all of which used MFS as a back off strategy and an external sense tagged data set. Our results are presented as precision (P), recall (R), and F1 value ($F1 = 2 * \frac{P * R}{P + R}$).

4.1 Results and Discussion

Table 2 shows the results when using various similarity for the *target selectors*. We selected gloss-based measures (Banerjee and Pedersen, 2003; Patwardhan et al., 2003) due to the need for handling multiple parts of speech for the *context selectors*. Functionality for our use of many different relatedness measurements was provided by WordNet::Similarity (Pedersen et al., 2004). Our method performs better than the MFS baseline, and clearly better than the random baseline. As one can see, the *scaled_sim* (path2) similarity measure along with the gloss based relatedness

	gloss1	gloss2
path1	78.8	78.3
path2	80.2	78.6
path3	78.7	78.6
IC1	78.6	79.3
IC2	78.5	79.2
IC3	78.0	78.1
gloss1	78.4	80.0
gloss2	78.6	78.9
MFS	baseline	77.4
random	baseline	59.1

Table 2: Performance of our method, given by F1 values (precision = recall), with various similarity measures for *target selectors*: **path1**= sim (normalized depth), **path2** = *scaled_sim*, **path3** = (Wu and Palmer, 1994), **IC1** = (Resnik, 1999), **IC2** = (Lin, 1997), **IC3** = (Jiang and Conrath, 1997), and relatedness measures for *context selectors*: **gloss1** = (Banerjee and Pedersen, 2003), **gloss2** = (Patwardhan et al., 2003). Baselines: **MFS** = most frequent sense, **random** = random choice of sense.

measure of (Banerjee and Pedersen, 2003) gave the best results. Note that the path-based and information content measures, in general, performed equally.

We experimented with using the gloss-based relatedness measures in place of similarity measures. The idea was that one measure could be used for both target selectors and context selectors. As one can gather from the bottom of table 2, for the most part, the measures performed equally. The experimental runtime of the path-based and information content measures was roughly one-fourth that of the gloss-based measures.

Table 3 presents results from experiments where we only attempted to annotate instances with over a minimum number of target selectors (tMin) and context selectors (cMin). We use steps of four for target selectors and steps of ten for context selectors, reflecting a ratio of roughly 2 target selectors for every 5 context selectors. It was more common for an instance to not have any target selectors than to not have context selectors, so we present results with only a tMin or cMin. The main goal of these experiments was simply to determine if the algorithm performed better on instances that we were able to acquire more selectors. We were able to see this was the case as the precision improved at the expense of recall from avoiding the noun instances

tMin	cMin	A	P	R	F1
0	0	1108	80.2	80.2	80.2
4	0	658	84.4	50.1	62.9
16	0	561	85.2	43.1	57.2
0	10	982	81.1	71.9	76.2
0	40	908	81.3	66.6	73.3
4	10	603	85.4	46.4	60.1
8	20	554	85.3	42.6	56.9
12	30	516	86.4	40.2	54.9
16	40	497	86.5	38.8	53.5

Table 3: Number attempted (**A**), Precision (**P**), Recall (**R**) and **F1** values of our method with restrictions on a minimum number of target selectors (**tMin**) and context selectors (**cMin**).

sel	noMFS	1SPD
80.2	79.6	79.8

Table 4: Results of a variety of experiments using *path2* and *gloss1* from the previous table. **noMFS** = no use of most frequent sense, **1SPD** = use of 1 sense per discourse.

that did not have many selectors.

Table 4 shows the results when we modify the method in a few ways. All these results use the *path2* (*scaled_sim*) and *gloss1* (Banerjee and Pedersen, 2003) measures. The results of Table 2 include first sense heuristic used as a back-off strategy for close calls, when multiple senses have a score within 0.05 of each other. Therefore, we experiment without this heuristic presented as *noMFS*, and found our method still performs strongly. We also implemented one sense per discourse, reported as *1SPD*. Our experimental corpus had five documents, and for each document we calculated the most commonly predicted sense and used that for all occurrences of the word within the document. Interestingly, this strategy does not seem to improve the results in our method.

4.2 Comparison with other systems

Table 5 shows the results of our method (sel) compared with a few systems participating in the SemEval coarse-grained all-words task. These results include the median of all participating systems, the top system not using training data (UPV-WSD) (Buscaldi and Rosso, 2007), and the top system using training data (NUS-PT) (Chan et al., 2007). The best performance reported on the

sel	med	UPV-WSD	NUS-PT	SSI
80.2	71.1	79.33	82.31	84.12

Table 5: Comparison of noun F1 values with various participants in the SemEval2007 coarse-grained all-words task.

nouns for the SemEval coarse-grained task, was actually from a system by the authors of the task (SSI) (Navigli and Velardi, 2005). All systems performing better than the MFS used the heuristic as a backoff strategy when unable to output a sense (Navigli et al., 2007). Also, the systems performing better than ours (including SSI) used more sources of sense annotated data.

5 Conclusion

We have presented a method for acquiring knowledge from the Web for noun sense disambiguation. Rather than searching the web with pre-chosen relatives, we search with a string representing the local context of a target word. This produces a list of selectors, words which may replace the target word within its local context. The selectors are then compared with the senses of the target word via similarity and relatedness measures to choose the correct sense. By searching with context instead of simply relatives, we are able to insure more relevant results from the web. Additionally, this method has an advantage over methods which use relatives and context in that it does not restrict the results to include pre-chosen words.

We also show that different types of similarity and relatedness measures are appropriate for different roles in our disambiguation algorithm. We found a path-based measure to be best with *target selectors* while a slower gloss-based method was appropriate for *context selectors* in order to handle multiple POS. For many tasks, information content based measures perform better than path-based measures. However, we found a path-based measure to be just as strong if not stronger in our approach.

Results of our evaluation using the SemEval coarse-grained all-words task showed strength in the use of selectors from the Web for disambiguation. Our system was out-performed only by systems using training data or substantially more annotated data. Future work may improve results through the use of sense tagged corpora, a grammatical parse, or other methods commonly used in

WSD. Additionally, better precision was achieved when requiring a minimum number of selectors, giving promise to improved results with more work in acquiring selectors. This paper has shown an effective and novel method of noun sense disambiguation through the use of selectors acquired from the web.

6 Acknowledgement

This research was supported in part by the NASA Engineering and Safety Center under Grant/Cooperative Agreement NNX08AJ98A.

References

- Agirre, Eneko and David Martinez. 2004. Unsupervised wsd based on automatically retrieved examples: The importance of bias. In *Proceedings of EMNLP 2004*, pages 25–32, Barcelona, Spain, July. Association for Computational Linguistics.
- Banerjee, S. and T. Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 805–810, Acapulco.
- Brody, Samuel, Roberto Navigli, and Mirella Lapata. 2006. Ensemble methods for unsupervised wsd. In *Proceedings of the 21st International Conference on Computational Linguistics*, pages 97–104, Sydney, Australia.
- Budanitsky, Alexander and Graeme Hirst. 2006. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.
- Buscaldi, Davide and Paolo Rosso. 2007. Upv-wsd : Combining different wsd methods by means of fuzzy borda voting. In *Proceedings of SemEval-2007*, pages 434–437, Prague, Czech Republic, June.
- Chan, Yee Seng, Hwee Tou Ng, and Zhi Zhong. 2007. Nus-pt: Exploiting parallel texts for word sense disambiguation in the english all-words tasks. In *Proceedings of Proceedings of SemEval-2007*, pages 253–256, Prague, Czech Republic, June.
- Jiang, Jay J. and David W. Conrath. 1997. Semantic similarity on corpus statistics and lexical taxonomy. In *Proceedings of ROCLING X*, Taiwan.
- Leacock, Claudia, Martin Chodorow, and George A. Miller. 1998. Using corpus statistics and wordnet relations for sense identification. *Computational Linguistics*, 24(1):147–165.
- Lin, Dekang. 1997. Using syntactic dependency as local context to resolve word sense ambiguity. In *Proceedings of the 35th annual meeting on Association for Computational Linguistics*, pages 64–71.
- Martinez, David, Eneko Agirre, and Xinglong Wang. 2006. Word relatives in context for word sense disambiguation. In *Proceedings of the 2006 Australasian Language Technology Workshop*, pages 42–50.
- McCarthy, Diana, Rob Koeling, Julie Weeds, and John Carroll. 2004. Finding predominant word senses in untagged text. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics*, pages 279–286, Barcelona, Spain, July.
- Mihalcea, Rada. 2002. Bootstrapping large sense tagged corpora. In *Proceedings of the 3rd International Conference on Languages Resources and Evaluations LREC 2002*, Las Palmas, Spain, May.
- Miller, George, R. Beckwith, Christiane Fellbaum, D. Gross, and K. Miller. 1993. Five papers on wordnet. Technical report, Princeton University.
- Navigli, Roberto and Paola Velardi. 2005. Structural semantic interconnections: A knowledge-based approach to word sense disambiguation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(7):1075–1086.
- Navigli, Roberto, Kenneth C. Litkowski, and Orin Hargraves. 2007. Semeval-2007 task 07: Coarse-grained english all-words task. In *Proceedings of SemEval-2007*, pages 30–35, Prague, Czech Republic, June.
- Patwardhan, S., S. Banerjee, and T. Pedersen. 2003. Using Measures of Semantic Relatedness for Word Sense Disambiguation. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, pages 241–257, Mexico City, Mexico, February.
- Pedersen, T., S. Patwardhan, and J. Michelizzi. 2004. WordNet::Similarity - Measuring the Relatedness of Concepts. In *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics Demonstrations*, pages 38–41, Boston, MA, May.
- Rada, R., H. Mili, E. Bicknell, and M. Blettner. 1989. Development and application of a metric on semantic nets. In *IEEE Transactions on Systems, Man and Cybernetics*, volume 19, pages 17–30.
- Resnik, Philip. 1999. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130.
- Wu, Zhibiao and Martha Palmer. 1994. Verb semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pages 133–138, New Mexico State University, Las Cruces, New Mexico.
- Yuret, Deniz. 2007. Ku: Word sense disambiguation by substitution. In *Proceedings of SemEval-2007*, pages 207–214, Prague, Czech Republic, June.