A Method for Word Sense Disambiguation of Unrestricted Text

Rada Mihalcea and **Dan I. Moldovan**
Department of Computer Science and Engineering
Southern Methodist University
Dallas, Texas, 75275-0122
{rada,moldovan}@seas.smu.edu

Abstract

Selecting the most appropriate sense for an ambiguous word in a sentence is a central problem in Natural Language Processing. In this paper, we present a method that attempts to disambiguate all the nouns, verbs, adverbs and adjectives in a text, using the senses provided in WordNet. The senses are ranked using two sources of information: (1) the Internet for gathering statistics for word-word cooccurrences and (2) WordNet for measuring the semantic density for a pair of words. We report an average accuracy of 80% for the first ranked sense, and 91% for the first two ranked senses. Extensions of this method for larger windows of more than two words are considered.

1 Introduction

Word Sense Disambiguation (WSD) is an open problem in Natural Language Processing. Its solution impacts other tasks such as discourse, reference resolution, coherence, inference and others. WSD methods can be broadly classified into three types:

1. WSD that make use of the information provided by machine readable dictionaries (Cowie et al., 1992), (Miller et al., 1994), (Agirre and Rigau, 1995), (Li et al., 1995), (McRoy, 1992);

2. WSD that use information gathered from training on a corpus that has already been semantically disambiguated (supervised training methods) (Gale et al., 1992), (Ng and Lee, 1996);

3. WSD that use information gathered from raw corpora (unsupervised training methods) (Yarowsky, 1995) (Resnik, 1997).

There are also hybrid methods that combine several sources of knowledge such as lexicon information, heuristics, collocations and others (McRoy, 1992) (Bruce and Wiebe, 1994) (Ng and Lee, 1996) (Rigau et al., 1997).

Statistical methods produce high accuracy results for small number of preselected words. A lack of widely available semantically tagged corpora almost excludes supervised learning methods. A possible solution for automatic acquisition of sense tagged corpora has been presented in (Mihalcea and Moldovan, 1999), but the corpora acquired with this method has not been yet tested for statistical disambiguation of words. On the other hand, the disambiguation using unsupervised methods has the disadvantage that the senses are not well defined. None of the statistical methods disambiguate adjectives or adverbs so far.

In this paper, we introduce a method that attempts to disambiguate all the nouns, verbs, adjectives and adverbs in a text, using the senses provided in WordNet (Fellbaum, 1998). To our knowledge, there is only one other method, recently reported, that disambiguates unrestricted words in texts (Stetina et al., 1998).

2 A word-word dependency approach

The method presented here takes advantage of the sentence context. The words are paired and an attempt is made to disambiguate one word within the context of the other word. This is done by searching on Internet with queries formed using different senses of one word, while keeping the other word fixed. The senses are ranked simply by the order provided by the number of hits. A good accuracy is obtained, perhaps because the number of texts on the Internet is so large. In this way, all the words are processed and the senses are ranked. We use the ranking of senses to curb the computational complexity in the step that follows. Only the most promising senses are kept.

The next step is to refine the ordering of senses by using a completely different method, namely the *semantic density.* This is measured by the number of common words that are within a semantic distance of two or more words. The closer the semantic relationship between two words the higher the semantic density between them. We introduce the semantic density because it is relatively easy to measure it on a MRD like WordNet. A metric is introduced in this sense which when applied to all possible combinations of the senses of two or more words it ranks them.

An essential aspect of the WSD method presented here is that it provides a raking of possible associations between words instead of a binary yes/no decision for each possible sense combination. This allows for a controllable precision as other modules may be able to distinguish later the correct sense association from such a small pool.

3 Contextual ranking of word senses

Since the Internet contains the largest collection of texts electronically stored, we use the Internet as a source of corpora for ranking the senses of the words.

3.1 Algorithm 1

For a better explanation of this algorithm, we provide the steps below with an example. We considered the verb-noun pair *"investigate report"*; in order to make easier the understanding of these examples, we took into consideration only the first two senses of the noun *report.* These two senses, as defined in WordNet, appear in the synsets: {*report#1, study*} and {*report#2, news report, story, account, write up*}.

INPUT: semantically untagged word$_1$ - word$_2$ pair $(W_1 - W_2)$
OUTPUT: ranking the senses of one word
PROCEDURE:

STEP 1. *Form a similarity list for each sense of one of the words.* Pick one of the words, say W_2, and using WordNet, form a similarity list for each sense of that word. For this, use the words from the synset of each sense and the words from the hypernym synsets. Consider, for example, that W_2 has m senses, thus W_2 appears in m similarity lists:
$(W_2^1, W_2^{1(1)}, W_2^{1(2)}, ..., W_2^{1(k_1)})$
$(W_2^2, W_2^{2(1)}, W_2^{2(2)}, ..., W_2^{2(k_2)})$
...
$(W_2^m, W_2^{m(1)}, W_2^{m(2)}, ..., W_2^{m(k_m)})$
where W_2^1, W_2^2, ..., W_2^m are the senses of W_2, and $W_2^{i(s)}$ represents the synonym number s of the sense W_2^i as defined in WordNet.
Example The similarity lists for the first two senses of the noun *report* are:
(report, study)
(report, news report, story, account, write up)

STEP 2. *Form* W_1 - $W_2^{i(s)}$ *pairs.* The pairs that may be formed are:
$(W_1 - W_2^1, W_1 - W_2^{1(1)}, W_1 - W_2^{1(2)}, ..., W_1 - W_2^{1(k_1)})$
$(W_1 - W_2^2, W_1 - W_2^{2(1)}, W_1 - W_2^{2(2)}, ..., W_1 - W_2^{2(k_2)})$
...
$(W_1 - W_2^m, W_1 - W_2^{m(1)}, W_1 - W_2^{m(2)}, ..., W_1 - W_2^{m(k_m)})$
Example The pairs formed with the verb *investigate* and the words in the similarity lists of the noun *report* are:
(investigate-report, investigate-study)
(investigate-report, investigate-news report, investigate-story, investigate-account, investigate-write up)

STEP 3. *Search the Internet and rank the senses* $W_2^{i(s)}$. A search performed on the Internet for each set of pairs as defined above, results in a value indicating the frequency of occurrences for W_1 and the sense of W_2. In our experiments we used (Altavista, 1996) since it is one of the most powerful search engines currently available. Using the operators provided by AltaVista, query-forms are defined for each $W_1 - W_2^{i(s)}$ set above:
(a) ("W_1 W_2^i" OR "W_1 $W_2^{i(1)}$" OR "W_1 $W_2^{i(2)}$" OR ... OR "W_1 $W_2^{i(k_i)}$")
(b) ((W_1 NEAR W_2^i) OR (W_1 NEAR $W_2^{i(1)}$) OR (W_1 NEAR $W_2^{i(2)}$) OR ... OR (W_1 NEAR $W_2^{i(k_i)}$))
for all $1 \leq i \leq m$. Using one of these queries, we get the number of hits for each sense i of W_2 and this provides a ranking of the m senses of W_2 as they relate with W_1.
Example The types of query that can be formed using the verb *investigate* and the similarity lists of the noun *report*, are shown below. After each query, we indicate the number of hits obtained

by a search on the Internet, using AltaVista.
(a) ("investigate report" OR "investigate study") *(478)*
("investigate report" OR "investigate news report" OR "investigate story" OR "investigate account" OR "investigate write up") *(281)*
(b) ((investigate NEAR report) OR (investigate NEAR study)) *(34880)*
((investigate NEAR report) OR (investigate NEAR news report) OR (investigate NEAR story) OR (investigate NEAR account) OR (investigate NEAR write up)) *(15884)*

A similar algorithm is used to rank the senses of W_1 while keeping W_2 constant (un-disambiguated). Since these two procedures are done over a large corpora (the Internet), and with the help of similarity lists, there is little correlation between the results produced by the two procedures.

3.1.1 Procedure Evaluation

This method was tested on 384 pairs: 200 verb-noun (file br-a01, br-a02), 127 adjective-noun (file br-a01), and 57 adverb-verb (file br-a01), extracted from SemCor 1.6 of the Brown corpus. Using query form (a) on AltaVista, we obtained the results shown in Table 1. The table indicates the percentages of correct senses (as given by SemCor) ranked by us in top 1, top 2, top 3, and top 4 of our list. We concluded that by keeping the top four choices for verbs and nouns and the top two choices for adjectives and adverbs, we cover with high percentage (mid and upper 90's) all relevant senses. Looking from a different point of view, the meaning of the procedure so far is that it *excludes* the senses that do not apply, and this can save a considerable amount of computation time as many words are highly polysemous.

	top 1	top 2	top 3	top 4
noun	76%	83%	86%	98%
verb	60%	68%	86%	87%
adjective	79.8%	93%		
adverb	87%	97%		

Table 1: Statistics gather from the Internet for 384 word pairs.

We also used the query form (b), but the results obtained were similar; using the operator $NEAR$, a larger number of hits is reported, but the sense ranking remains more or less the same.

3.2 Conceptual density algorithm

A measure of the relatedness between words can be a knowledge source for several decisions in NLP applications. The approach we take here is to construct a linguistic context for each sense of the verb and noun, and to measure the number of the common nouns shared by the verb and the noun contexts. In WordNet each concept has a gloss that acts as a micro-context for that concept. This is a rich source of linguistic information that we found useful in determining conceptual density between words.

3.2.1 Algorithm 2

INPUT: semantically untagged verb - noun pair and a ranking of noun senses (as determined by Algorithm 1)
OUTPUT: sense tagged verb - noun pair
PROCEDURE:
STEP 1. Given a verb-noun pair $V - N$, denote with $< v_1, v_2, ..., v_h >$ and $< n_1, n_2, ..., n_l >$ the possible senses of the verb and the noun using WordNet.
STEP 2. Using *Algorithm 1*, the senses of the noun are ranked. Only the first t possible senses indicated by this ranking will be considered. The rest are dropped to reduce the computational complexity.
STEP 3. For each possible pair $v_i - n_j$, the conceptual density is computed as follows:
(a) Extract all the glosses from the sub-hierarchy including v_i (the rationale for selecting the sub-hierarchy is explained below)
(b) Determine the nouns from these glosses. These constitute the noun-context of the verb. Each such noun is stored together with a weight w that indicates the level in the sub-hierarchy of the verb concept in whose gloss the noun was found.
(c) Determine the nouns from the noun sub-hierarchy including n_j.
(d) Determine the conceptual density C_{ij} of common concepts between the nouns obtained at (b) and the nouns obtained at (c) using the metric:

$$C_{ij} = \frac{\sum_{k}^{|cd_{ij}|} w_k}{log(descendents_j)} \qquad (1)$$

where:

- $|cd_{ij}|$ is the number of common concepts between the hierarchies of v_i and n_j

- w_k are the levels of the nouns in the hierarchy of verb v_i
- $descendents_j$ is the total number of words within the hierarchy of noun n_j

STEP 4. C_{ij} ranks each pair $v_i - n_j$, for all i and j.

Rationale
1. In WordNet, a gloss explains a concept and provides one or more examples with typical usage of that concept. In order to determine the most appropriate noun and verb hierarchies, we performed some experiments using SemCor and concluded that the noun sub-hierarchy should include all the nouns in the class of n_j. The sub-hierarchy of verb v_i is taken as the hierarchy of the highest hypernym h_i of the verb v_i. It is necessary to consider a larger hierarchy then just the one provided by synonyms and direct hyponyms. As we replaced the role of a corpora with glosses, better results are achieved if more glosses are considered. Still, we do not want to enlarge the context too much.

2. As the nouns with a big hierarchy tend to have a larger value for $|cd_{ij}|$, the weighted sum of common concepts is normalized with respect to the dimension of the noun hierarchy. Since the size of a hierarchy grows exponentially with its depth, we used the logarithm of the total number of descendants in the hierarchy, i.e. $log(descendents_j)$.

3. We also took into consideration and have experimented with a few other metrics. But after running the program on several examples, the formula from Algorithm 2 provided the best results.

4 An Example

As an example, let us consider the verb-noun collocation *revise law*. The verb *revise* has two possible senses in WordNet 1.6 and the noun *law* has seven senses. Figure 1 presents the synsets in which the different meanings of this verb and noun appear.

First, Algorithm 1 was applied and search the Internet using AltaVista, for all possible pairs V-N that may be created using *revise* and the words from the similarity lists of *law*. The following ranking of senses was obtained: *law#2*(2829), *law#3*(648), *law#4*(640), *law#6*(397), *law#1*(224), *law#5*(37), *law#7*(0),

REVISE
1. {revise#1}
=> { rewrite}
2. {retool, revise#2}
=> { reorganize, shake up}

LAW
1. { law#1, jurisprudence}
=> {collection, aggregation, accumulation, assemblage}
2. {law#2}
=> {rule, prescript} ...
3. {law#3, natural law}
=> { concept, conception, abstract}
4. {law#4, law of nature}
=> { concept, conception, abstract}
5. {jurisprudence, law#5, legal philosophy}
=> { philosophy}
6. {law#6, practice of law}
=> { learned profession}
7. {police, police force, constabulary, law#7}
=> { force, personnel}

Figure 1: Synsets and hypernyms for the different meanings , as defined in WordNet

where the numbers in parentheses indicate the number of hits. By setting the threshold at $t = 2$, we keep only sense #2 and #3.

Next, Algorithm 2 is applied to rank the four possible combinations (two for the verb times two for the noun). The results are summarized in Table 2: (1) $|cd_{ij}|$ - the number of common concepts between the verb and noun hierarchies; (2) $descendants_j$ the total number of nouns within the hierarchy of each sense n_j; and (3) the conceptual density C_{ij} for each pair $n_i - v_j$ derived using the formula presented above.

	$\lvert cd_{ij}\rvert$		$descendants_j$		C_{ij}	
	n_2	n_3	n_2	n_3	n_2	n_3
v_1	5	4	975	1265	0.30	0.28
v_2	0	0	975	1265	0	0

Table 2: Values used in computing the conceptual density and the conceptual density C_{ij}

The largest conceptual density $C_{12} = 0.30$ corresponds to $v_1 - n_2$: $revise\#1/2 - law\#2/5$ (the notation $\#i/n$ means sense i out of n possible senses given by WordNet). This combination of verb-noun senses also appears in SemCor, file br-a01.

5 Evaluation and comparison with other methods

5.1 Tests against SemCor

The method was tested on 384 pairs selected from the first two tagged files of SemCor 1.6 (file br-a01, br-a02). From these, there are 200 verb-noun pairs, 127 adjective-noun pairs and 57 adverb-verb pairs.
In Table 3, we present a summary of the results.

	top 1	top 2	top 3	top 4
noun	86.5%	96%	97%	98%
verb	67%	79%	86%	87%
adjective	79.8%	93%		
adverb	87%	97%		

Table 3: Final results obtained for 384 word pairs using both algorithms.

Table 3 shows the results obtained using both algorithms; for nouns and verbs, these results are improved with respect to those shown in Table 1, where only the first algorithm was applied. The results for adjectives and adverbs are the same in both these tables; this is because the second algorithm is not used with adjectives and adverbs, as words having this part of speech are not structured in hierarchies in WordNet, but in clusters; the small size of the clusters limits the applicability of the second algorithm.

Discussion of results When evaluating these results, one should take into consideration that:

1. Using the glosses as a base for calculating the conceptual density has the advantage of eliminating the use of a large corpus. But a disadvantage that comes from the use of glosses is that they are not part-of-speech tagged, like some corpora are (i.e. Treebank). For this reason, when determining the nouns from the verb glosses, an error rate is introduced, as some verbs (like *make, have, go, do*) are lexically ambiguous having a noun representation in WordNet as well. We believe that future work on part-of-speech tagging the glosses of WordNet will improve our results.

2. The determination of senses in SemCor was done of course within a larger context, the context of sentence and discourse. By working only with a pair of words we do not take advantage of such a broader context. For example, when disambiguating the pair *protect court* our method picked the court meaning *"a room in which a law court sits"* which seems reasonable given only two words, whereas SemCor gives the court meaning *"an assembly to conduct judicial business"* which results from the sentence context (this was our second choice). In the next section we extend our method to more than two words disambiguated at the same time.

5.2 Comparison with other methods

As indicated in (Resnik and Yarowsky, 1997), it is difficult to compare the WSD methods, as long as distinctions reside in the approach considered (MRD based methods, supervised or unsupervised statistical methods), and in the words that are disambiguated. A method that disambiguates unrestricted nouns, verbs, adverbs and adjectives in texts is presented in (Stetina et al., 1998); it attempts to exploit sentential and discourse contexts and is based on the idea of semantic distance between words, and lexical relations. It uses WordNet and it was tested on SemCor.

Table 4 presents the accuracy obtained by other WSD methods. The baseline of this comparison is considered to be the simplest method for WSD, in which each word is tagged with its most common sense, i.e. the first sense as defined in WordNet.

	Base line	Stetina	Yarowsky	Our method
noun	80.3%	85.7%	93.9%	86.5%
verb	62.5%	63.9%	-	67%
adjective	81.8%	83.6%	-	79.8
adverb	84.3%	86.5%	-	87%
AVERAGE	77%	80%	-	80.1%

Table 4: A comparison with other WSD methods.

As it can be seen from this table, (Stetina et al., 1998) reported an average accuracy of 85.7% for nouns, 63.9% for verbs, 83.6% for adjectives and 86.5% for adverbs, slightly less than our results. Moreover, for applications such as information retrieval we can use more than one sense combination; if we take the top 2 ranked combinations our average accuracy is 91.5% (from Table 3).

Other methods that were reported in the literature disambiguate either one part of speech word (i.e. nouns), or in the case of purely statistical methods focus on very limited number of words. Some of the best results were reported in (Yarowsky, 1995) who uses a large training corpus. For the noun *drug* Yarowsky obtains 91.4% correct performance and when considering the restriction "one sense per discourse" the accuracy increases to 93.9%, result represented in the third column in Table 4.

6 Extensions

6.1 Noun-noun and verb-verb pairs

The method presented here can be applied in a similar way to determine the conceptual density within noun-noun pairs, or verb-verb pairs (in these cases, the $NEAR$ operator should be used for the first step of this algorithm).

6.2 Larger window size

We have extended the disambiguation method to more than two words co-occurrences. Consider for example:

The bombs caused damage but no injuries.

The senses specified in SemCor, are:

1a. bomb(#1/3) cause(#1/2) damage(#1/5) injury(#1/4)

For each word X, we considered all possible combinations with the other words Y from the sentence, two at a time. The conceptual density C was computed for the combinations $X-Y$ as a summation of the conceptual densities between the sense i of the word X and all the senses of the words Y. The results are shown in the tables below where the conceptual density calculated for the sense $\#i$ of word X is presented in the column denoted by $C\#i$:

$X-Y$	$C\#1$	$C\#2$	$C\#3$
bomb-cause	0.57	0	0
bomb-damage	5.09	0.13	0
bomb-injury	2.69	0.15	0
SCORE	8.35	0.28	0

By selecting the largest values for the conceptual density, the words are tagged with their senses as follows:

1b. bomb(#1/3) cause(#1/2) damage(#1/5) injury(#2/4)

$X-Y$	$C\#1$	$C\#2$
cause-bomb	5.16	1.34
cause-damage	12.83	2.64
cause-injury	12.63	1.75
SCORE	30.62	5.73

$X-Y$	$C\#1$	$C\#2$	$C\#3$	$C\#4$	$C\#5$
damage-bomb	5.60	2.14	1.95	0.88	2.16
damage-cause	1.73	2.63	0.17	0.16	3.80
damage-injury	9.87	2.57	3.24	1.56	7.59
SCORE	17.20	7.34	5.36	2.60	13.55

Note that the senses for word *injury* differ from *1a.* to *1b.*; the one determined by our method *(#2/4)* is described in WordNet as *"an accident that results in physical damage or hurt"* (hypernym: accident), and the sense provided in SemCor *(#1/4)* is defined as *"any physical damage"*(hypernym: health problem).

This is a typical example of a mismatch caused by the fine granularity of senses in WordNet which translates into a human judgment that is not a clear cut. We think that the sense selection provided by our method is justified, as both *damage* and *injury* are objects of the same verb *cause*; the relatedness of *damage(#1/5)* and *injury(#2/4)* is larger, as both are of the same class noun.event as opposed to *injury(#1/4)* which is of class noun.state.

Some other randomly selected examples considered were:

2a. The terrorists(#1/1) bombed(#1/3) the embassies(#1/1).
2b. terrorist(#1/1) bomb(#1/3) embassy(#1/1)
3a. A car-bomb(#1/1) exploded(#2/10) in front of PRC(#1/1) embassy(#1/1).
3b. car-bomb(#1/1) explode(#2/10) PRC(#1/1) embassy(#1/1)
4a. The bombs(#1/3) broke(#23/27) windows(#1/4) and destroyed(#2/4) the two vehicles(#1/2).
4b. bomb(#1/3) break(#3/27) window(#1/4) destroy(#2/4) vehicle(#1/2)

where sentences *2a*, *3a* and *4a* are extracted from SemCor, with the associated senses for each word, and sentences *2b*, *3b* and *4b* show the verbs and the nouns tagged with their senses by our method. The only discrepancy is for the

$X-Y$	$C\#1$	$C\#2$	$C\#3$	$C\#4$
injury-bomb	2.35	5.35	0.41	2.28
injury-cause	0	4.48	0.05	0.01
injury-damage	5.05	10.40	0.81	9.69
SCORE	7.40	20.23	1.27	11.98

word *broke* and perhaps this is due to the large number of its senses. The other word with a large number of senses *explode* was tagged correctly, which was encouraging.

7 Conclusion

WordNet is a fine grain MRD and this makes it more difficult to pinpoint the correct sense combination since there are many to choose from and many are semantically close. For applications such as machine translation, fine grain disambiguation works well but for information extraction and some other applications this is an overkill, and some senses may be lumped together. The ranking of senses is useful for many applications.

References

E. Agirre and G. Rigau. 1995. A proposal for word sense disambiguation using conceptual distance. In *Proceedings of the First International Conference on Recent Advances in Natural Language Processing*, Velingrad.

Altavista. 1996. Digital equipment corporation. "http://www.altavista.com".

R. Bruce and J. Wiebe. 1994. Word sense disambiguation using decomposable models. In *Proceedings of the Thirty Second Annual Meeting of the Association for Computational Linguistics (ACL-94)*, pages 139–146, LasCruces, NM, June.

J. Cowie, L. Guthrie, and J. Guthrie. 1992. Lexical disambiguation using simulated annealing. In *Proceedings of the Fifth International Conference on Computational Linguistics COLING-92*, pages 157–161.

C. Fellbaum. 1998. *WordNet, An Electronic Lexical Database.* The MIT Press.

W. Gale, K. Church, and D. Yarowsky. 1992. One sense per discourse. In *Proceedings of the DARPA Speech and Natural Language Workshop*, Harriman, New York.

X. Li, S. Szpakowicz, and M. Matwin. 1995. A wordnet-based algorithm for word semantic sense disambiguation. In *Proceedings of the Forteen International Joint Conference on Artificial Intelligence IJCAI-95*, Montreal, Canada.

S. McRoy. 1992. Using multiple knowledge sources for word sense disambiguation. *Computational Linguistics*, 18(1):1–30.

R. Mihalcea and D.I. Moldovan. 1999. An automatic method for generating sense tagged corpora. In *Proceedings of AAAI-99*, Orlando, FL, July. (to appear).

G. Miller, M. Chodorow, S. Landes, C. Leacock, and R. Thomas. 1994. Using a semantic concordance for sense identification. In *Proceedings of the ARPA Human Language Technology Workshop*, pages 240–243.

H.T. Ng and H.B. Lee. 1996. Integrating multiple knowledge sources to disambiguate word sense: An examplar-based approach. In *Proceedings of the Thirtyfour Annual Meeting of the Association for Computational Linguistics (ACL-96)*, Santa Cruz.

P. Resnik and D. Yarowsky. 1997. A perspective on word sense disambiguation methods and their evaluation. In *Proceedings of ACL Siglex Workshop on Tagging Text with Lexical Semantics, Why, What and How?*, Washington DC, April.

P. Resnik. 1997. Selectional preference and sense disambiguation. In *Proceedings of ACL Siglex Workshop on Tagging Text with Lexical Semantics, Why, What and How?*, Washington DC, April.

G. Rigau, J. Atserias, and E. Agirre. 1997. Combining unsupervised lexical knowledge methods for word sense disambiguation. *Computational Linguistics.*

J. Stetina, S. Kurohashi, and M. Nagao. 1998. General word sense disambiguation method based on a full sentential context. In *Usage of WordNet in Natural Language Processing, Proceedings of COLING-ACL Workshop*, Montreal, Canada, July.

D. Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the Thirtythird Association of Computational Linguistics.*