An Iterative Approach to Word Sense Disambiguation

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

In this paper, we present an iterative algorithm for Word Sense Disambiguation. It combines two sources of information: WordNet and a semantic tagged corpus, for the purpose of identifying the correct sense of the words in a given text. It differs from other standard approaches in that the disambiguation process is performed in an iterative manner: starting from free text, a set of disambiguated words is built, using various methods; new words are sense tagged based on their relation to the already disambiguated words, and then added to the set. This iterative process allows us to identify, in the original text, a set of words which can be disambiguated with high precision; 55% of the verbs and nouns are disambiguated with an accuracy of 92%.

Introduction

Word Sense Disambiguation (WSD) is an open problem in Natural Language Processing (NLP). Its solution impacts other tasks such as information retrieval, machine translation, discourse, reference resolution and others.

WSD methods can be broadly classified into four types:

1. WSD that makes use of the information provided by Machine Readable Dictionaries (MRD) (Miller et al. 1994), (Agirre and Rigau 1995), (Li, Szpakowicz and Matwin), (Leacock, Chodorow and Miller 1998);
2. WSD that uses information gathered from training on a corpus that has already been semantically disambiguated (supervised training methods) (Ng and Lee 1996);
3. WSD that uses information gathered from raw corpora (unsupervised training methods) (Yarowsky 1995) (Resnik 1997).
4. WSD methods using machine learning algorithms (Yarowsky 1995), (Leacock, Chodorow and Miller 1998).

There are also hybrid methods that combine several sources of knowledge such as lexicon information, heuristics, collocations and others (Bruce and Wiebe 1994) (Ng and Lee 1996) (Rigau, Atserias and Agirre 1997) (Mihalcea and Moldovan 1999).

The method proposed here is a hybrid method, and uses information gathered from a MRD, namely WordNet, and from a semantic tagged corpus, i.e. SemCor. It differs from previous approaches in that it uses an iterative approach: the algorithm has as input the set of nouns and verbs extracted from the input text, and incrementally builds a set of disambiguated words. This approach allows us to identify, with high precision, the semantic senses for a subset of the input words. About 55% of the nouns and verbs are disambiguated with a precision of 92%.

The algorithm presented here is part of an ongoing research for the purpose of integrating WSD techniques into Information Retrieval (IR) systems. It is an improvement over our previous work in WSD (Mihalcea and Moldovan 1999). This method can be also used in combination with other WSD algorithms with the purpose of fully disambiguating free text.

Lately, the biggest effort to incorporate WSD into larger applications is performed in the field of IR. The inputs of IR systems usually consist of a question/query and a set of documents from which the information has to be retrieved. This led to two main directions considered so far by researchers, for the purpose of increasing the IR performance with WSD techniques:

1. The disambiguation of the words in the input query. The purpose of this is to expand the query with similar words, and thus to improve the recall of the IR system. (Voorhees 1994), (Voorhees 1998) and (Moldovan and Mihalcea 2000) proved that this technique can be useful if the disambiguation process is highly accurate.
2. The disambiguation of words in the documents. (Schutze and Pedersen 1995) proved that sense-based retrieval can increase the precision of an IR system up to 7%, while a combination of sense-based and word-based retrieval increases the precision up to 14%.

With the algorithm described in this paper, a large subset of the words in the documents can be disambiguated with high precision, allowing for an efficient combined sense-based and word-based retrieval.

Resources

WordNet (WordNet 1.6 has been used in our method) is a Machine Readable Dictionary developed at Princeton University by a group led by George Miller (Fellbaum 1998). WordNet covers the vast majority of nouns, verbs, adjectives and adverbs from the English language. It has a large network of 129,504 words, organized in 98,548 synonym sets, called *synsets*.

The main semantic relation defined in WordNet is the *"is a"* relation; each concept subsumes more specific concepts, called *hyponyms*, and it is subsumed by more general concepts, called *hypernyms*. For example, the concept {`machine`} has the hypernym {`device`}, and one of its hyponyms is {`calculator, calculating machine`}.

WordNet defines one or more senses for each word. Depending on the number of senses it has, a word can be (1) *monosemous*, i.e. it has only one sense, for example the noun `interestingness`, or (2) *polysemous*, i.e. it has two or more senses, for example the noun `interest` which has seven senses defined in WordNet.

SemCor. SemCor (Miller et al. 1993) is a corpus formed with about 25% of the Brown corpus files; all the words in SemCor are part-of-speech tagged and semantically disambiguated. In the algorithm described here, we use the brown1 and brown2 sections of SemCor, containing 185 files; from these, 6 files are used with the purpose of testing our method; the other 179 files form a corpus used to extract rules with procedure 3 and to determine *noun-contexts* for procedure 4 (as described in the next section).

Iterative Word Sense Disambiguation

The algorithm presented in this paper determines, in a given text, a set of nouns and verbs which can be disambiguated with high precision. The semantic tagging is performed using the senses defined in WordNet.

In this section, we are going to present the various methods used to identify the correct sense of a word. Next, we present the main algorithm in which these procedures are invoked in an iterative manner.

PROCEDURE 1. This procedure uses a Named Entity (NE) component to recognize and identify person names, locations, company names and others. The various names are recognized and tagged. Of interest for our purpose are the PER (person), ORG(group) and LOC(location) tags. The words or word collocations marked with such tags are replaced by their role (person, group, location) and marked as having sense #1.

Example. `''Scott Hudson''` is identified as a person name, thus this word group will be replaced with its role, i.e. `person`, and marked with sense #1.

PROCEDURE 2. Identify the words having only one sense in WordNet (*monosemous* words). Mark them with sense #1.

Example. The noun `subcommittee` has one sense defined in WordNet. Thus, it is a *monosemous* word and can be marked as having sense #1.

PROCEDURE 3. With this procedure, we are trying to get contextual clues regarding the usage of the sense of a word. For a given word W_i, at position i in the text, form two pairs, one with the word before W_i (pair W_{i-1}-W_i) and the other one with the word after W_i (pair W_i-W_{i+1}). Determiners or conjunctions cannot be part of these pairs. Then, we extract all the occurrences of these pairs found within the semantic tagged corpus formed with the 179 texts from SemCor. If, in all the occurrences, the word W_i has only one sense #k, and the number of occurrences of this sense is larger than a given threshold, then mark the word W_i as having sense #k.

Example. Consider the word `approval` in the text fragment `''committee approval of''`, and the threshold set to 3. The pairs formed are `''committee approval''` and `''approval of''`. No occurrences of the first pair are found in the corpus. Instead, there are four occurrences of the second pair:

"... with the *approval#1 of* the Farm Credit Association ..."
"... subject to the *approval#1 of* the Secretary of State ..."
"... administrative *approval#1 of* the reclassification ... "
"... recommended *approval#1 of* the 1-A classification ..."

In all these occurrences the sense of `approval` is sense #1. Thus, `approval` is marked with sense #1.

PROCEDURE 4. For a given noun N in the text, determine the *noun-context* of each of its senses. This *noun-context* is actually a list of nouns which can occur within the context of a given sense i of the noun N. In order to form the *noun-context* for every sense N_i, we determine all the concepts in the hypernym synsets of N_i. Also, using SemCor, we determine all the nouns which occur within a window of 10 words respect to N_i.

All of these nouns, determined using WordNet and SemCor, constitute the *noun-context* of N_i. We can now calculate the number of common words between this *noun-context* and the original text in which the noun N is found.

Applying this procedure to all the senses of noun N will provide us with an ordering over its possible senses. We pick up the sense i for the noun N which: (1) is in the top of this ordering and (2) has the distance to the next sense in this ordering larger than a given threshold.

Example. The word `diameter`, as it appears in a text from the aerodynamics field (Cranfield collection), has two senses. The common words found between the *noun-contexts* of its senses and the text are: for `diameter#1`: { property, hole, ratio } and for `diameter#2`: { form}. For this text, the threshold was set to 1, and thus we pick `diameter#1` as the correct sense (there is a difference larger than 1 between the number of nouns in the two sets).

PROCEDURE 5. Find words which are semantically connected to the already disambiguated words for which the connection distance is 0. The semantic distance is computed based on the WordNet hierarchy; two words are semantically connected at a distance of 0 if they belong to the same synset.

Example. Consider these two words appearing in the text to be disambiguated: `authorize` and `clear`. The verb `authorize` is a monosemous word, and thus it is disambiguated with procedure 2. One of the senses of the verb `clear`, namely sense #4, appears in the same synset with `authorize#1`, and thus `clear` is marked as having sense #4.

<u>PROCEDURE 6</u>. Find words which are semantically connected, and for which the connection distance is 0. This procedure is weaker than procedure 5: none of the words considered by this procedure are already disambiguated. We have to consider all the senses of both words in order to determine whether or not the distance between them is 0, and this makes this procedure computationally intensive.

Example. For the words `measure` and `bill`, both of them ambiguous, this procedure tries to find two possible senses for these words, which are at a distance of 0, i.e. they belong to the same synset. The senses found are `measure#4` and `bill#1`, and thus the two words are marked with their corresponding senses.

<u>PROCEDURE 7</u>. Find words which are semantically connected to the already disambiguated words, and for which the connection distance is maximum 1. Again, the distance is computed based on the WordNet hierarchy; two words are semantically connected at a maximum distance of 1 if they are *synonyms* or they belong to a *hypernymy/hyponymy* relation.

Example. Consider the nouns `subcommittee` and `committee`. The first one is disambiguated with procedure 2, and thus it is marked with sense #1. The word `committee` with its sense #1 is semantically linked with the word `subcommittee` by a *hypernymy* relation. Hence, we semantically tag this word with sense #1.

<u>PROCEDURE 8</u>. Find words which are semantically connected among them, and for which the connection distance is maximum 1. This procedure is similar with procedure 6: both words are ambiguous, and thus all their senses have to be considered in the process of finding the distance between them.

Example. The words `gift` and `donation` are both ambiguous. This procedure finds `gift` with sense #1 as being the hypernym of `donation`, also with sense #1. Therefore, both words are disambiguated and marked with their assigned senses.

The procedures presented above are applied iteratively; this allows us to identify a set of nouns and verbs which can be disambiguated with high precision. About 55% of the nouns and verbs are disambiguated with 92% accuracy.

Algorithm

Step 1. Pre-process the text. This implies tokenization and part-of-speech tagging. The part-of-speech tagging task is performed with high accuracy using an improved version of Brill's tagger (Brill 1992). At this step, we also identify the complex nominals, based on WordNet definitions. For example, the word sequence `''pipeline companies''` is found in WordNet and thus it is identified as a single concept. There is also a list of words which we do not attempt to disambiguate. These words are marked with a special flag to indicate that they should not be considered in the disambiguation process. So far, this list consists of three verbs: *be, have, do.*

Step 2. Initialize the Set of Disambiguated Words (SDW) with the empty set SDW={}. Initialize the Set of Ambiguous Words (SAW) with the set formed by all the nouns and verbs in the input text.

Step 3. Apply procedure 1. The named entities identified here are removed from SAW and added to SDW.

Step 4. Apply procedure 2. The monosemous words found here are removed from SAW and added to SDW.

Step 5. Apply procedure 3. This step allows us to disambiguate words based on their occurrence in the semantically tagged corpus. The words whose sense is identified with this procedure are removed from SAW and added to SDW.

Step 6. Apply procedure 4. This will identify a set of nouns which can be disambiguated based on their *noun-contexts*.

Step 7. Apply procedure 5. This procedure tries to identify a *synonymy* relation between the words from SAW and SDW. The words disambiguated are removed from SAW and added to SDW.

Step 8. Apply procedure 6. This step is different from the previous one, as the *synonymy* relation is sought among words in SAW (no SDW words involved). The words disambiguated are removed from SAW and added to SDW.

Step 9. Apply procedure 7. This step tries to identify words from SAW which are linked at a distance of maximum 1 with the words from SDW. Remove the words disambiguated from SAW and add them to SDW.

Step 10. Apply procedure 8. This procedure finds words from SAW connected at a distance of maximum 1. As in step 8, no words from SDW are involved. The words disambiguated are removed from SAW and added to SDW.

An Example

We illustrate here the disambiguation algorithm with the help of an example; for this, we consider the following set of sentences extracted from the file br-m02 from SemCor.

"... Instead of inflecting a verb or using an unattached particle to indicate the past or future, Siddo used an entirely different word. Thus, the masculine animate infinitive dabhumaksanigalu'ahai, meaning to live, was, in the perfect tense, ksu'u'peli'afo, and, in the future, mai'teipa. The same use of an entirely different word applied for all the other tenses. Plus the fact that Siddo not only had the normal (to Earthmen) three genders of masculine, feminine, and neuter, but the two extra of inanimate and spiritual. Fortunately, gender was inflected, though the expression of it would be difficult for anybody not born in Siddo. The system

Set	No. words	Proc.1+2		Proc.3		Proc.4		Proc.5+6		Proc.7+8	
		No.	Acc.	No.	Acc.	No.	Acc.	No.	Acc.	No.	Acc.
1	151	35	100%	35	100%	59	94%	73	87.8%	88	82.5%
2	128	47	100%	50	98.4%	65	93.6%	70	90.4%	85	85.7%
3	108	36	100%	40	100%	48	98.2%	53	98.3%	62	91.2%
4	159	40	100%	43	100%	62	89.6%	64	88.6%	72	88.5%
5	159	52	100%	55	100%	76	91.9%	82	91.3%	89	87.9%
6	89	33	100%	35	100%	41	100%	41	100%	43	100%
AVERAGE	132	40	100%	43	99.7%	58.5	94.6%	63.8	92.7%	73.2	89.3%

Table 1: Results obtained for sets of sentences from file br-a01

File	No. words	Proc.1+2		Proc.3		Proc.4		Proc.5+6		Proc.7+8	
		No.	Acc.	No.	Acc.	No.	Acc.	No.	Acc.	No.	Acc.
br-a01	132	40	100%	43	99.7%	58.5	94.6%	63.8	92.7%	73.2	89.3%
br-a02	135	49	100%	52.5	98.5%	68.6	94%	75.2	92.4%	81.2	91.4%
br-k01	68.1	17.2	100%	23.3	99.7%	38.1	97.4%	40.3	97.4%	41.8	96.4%
br-k18	60.4	18.1	100%	20.7	99.1%	26.6	96.9%	27.8	95.3%	29.8	93.2%
br-m02	63	17.3	100%	20.3	98.1%	26.1	95%	26.8	94.9%	30.1	93.9%
br-r05	72.5	14.3	100%	16.6	98.1%	27	93.2%	30.2	91.5%	34.2	89.1%
AVERAGE	88.5	25.9	100%	29.4	98.8%	40.8	95.2%	44	94%	48.4	92.2%

Table 2: Summary of results for 52 texts

of indicating gender varied according to tense. All the other parts of speech: nouns, pronouns, adjectives, adverbs, and conjunctions operated under the same system as the verbs."

First, the text is tokenized and part of speech tagged. We start by initializing SAW with the set of all nouns and verbs in the text, and SDW is initialized to the empty set. As words are disambiguated using the algorithm described above, they are removed from the SAW set and added to the SDW set.

Using procedure 1, the complex nominals are identified based on WordNet dictionary and the named entities are recognized. The following complex nominals have been identified: `''perfect tense''` and `''parts of speech''`. Siddo is identified as an organization by the Named Entity recognizer and added to the SDW set.

The monosemous words are identified with procedure 2, and at this step the SDW set becomes {infinitive#1, perfect_tense#1, tense#1, Earthman#1, neuter#1, part_of_speech#1, pronoun#1}.

Then, we apply procedure 3, which tries to get rules from SemCor; this will identify `future` as having sense #1; this word is added to SDW. We then apply procedure 4, which identifies `fact` with sense #1, using its noun-contexts.

Next, we apply procedure 5, and find another occurrence of the word `future` and assign to this word the correspondent sense, i.e. sense #1. Procedure 6 cannot be applied on this text.

By applying procedure 7, we try to find words related at a distance of maximum 1 with the words already in SDW. With this procedure, the following words have been disambiguated: `verb#1` (in hypernymy relation with infinitive#1); `past#3` (in hyponymy relation with `tense#1`); `gender#1` (hypernymy relation with `neuter#1`); other two occurrences of `gender` are disambiguated due to the same semantic relation; `noun#2` (hyponymy relation with `part_of_speech#1`); `adjective#2` (hyponymy relation with `part_of_speech#1`); `adverb#1` (hyponymy relation with `part_of_speech#1`); `verb#1` (hyponymy relation with `part_of_speech#1`).

Finally, the SDW set becomes SDW={verb#1, past#3, future#1, infinitive#1, perfect_tense#1, tense#1, Earthman#1, gender#1, neuter#1, part_of_speech#1, noun#2, pronoun#1, adjective#2, adverb#1, verb#1}.

Using this algorithm, we have disambiguated part of the nouns and verbs in the text with high precision. Respect to SemCor, the precision achieved on this text is 92%.

Results

To determine the accuracy and the recall of the disambiguation method presented here, we have performed tests on 6 randomly selected files from SemCor. The following files have been used: br-a01, br-a02, br-k01, br-k18, br-m02, br-r05. Each of these files was split into smaller files with a maximum of 15 lines each. This size limit is based on our observation that small contexts reduce the applicability of procedures 5-8, while large contexts become a source of errors. Thus, we have created a benchmark with 52 texts, on which we have tested the disambiguation method.

In table 1, we present the results obtained for the br-a01 file. The file has been divided into 5 sets of 15 sentences. The number of nouns and verbs considered by the disambiguation algorithm is shown in the first column. In columns 3 and 4, there are presented the number of words disambiguated with procedures 1 and 2, and the accuracy obtained with these procedures. Column 5 and 6 present the number of words disambiguated and the accuracy obtained after applying procedure 3 (cumulative results). The cumulative results obtained after applying procedures 3, 4 and 5, 6 and 7, are shown in columns 7 and 8, 9 and 10, respectively columns 10 and 11. For this file, 55% of the nouns and verbs were disambiguated with 89.3% accuracy.

Table 2 presents the results obtained for the 52 texts created from the 6 SemCor files. The first column indicates the file for which the results are presented; the meaning of the numbers in the other columns is the same as in the previous table.

On average, 55% of the nouns and verbs were disambiguated with 92.2% accuracy.

Conclusion and further work

In this paper, we presented a method for disambiguating the nouns and verbs in an input text. The novelty of this method consists of the fact that the disambiguation process is done in an iterative manner. Several procedures, described in the paper, are applied such as to build a set of words which are disambiguated with high accuracy: 55% of the nouns and verbs are disambiguated with a precision of 92.22%.

The most important improvements which are expected to be achieved on the WSD problem are *precision* and *speed*. In the case of our approach to WSD, we can also talk about the need for an increased *recall*, meaning that we want to obtain a larger number of words which can be disambiguated in the input text.

The precision of more than 92% obtained during our experiments is very high, considering the fact that WordNet, which is the dictionary used for sense identification, is very fine grained and sometime the senses are very close to each other. The accuracy of 92% obtained is close to the precision achieved by humans in sense disambiguation.

As stated earlier in this paper, IR systems can benefit from a WSD method which enables the disambiguation of some of the words with high accuracy. This enables an efficient word-based and sense-based combined indexing, without having the errors introduced by a complete disambiguation process with a lower accuracy.

References

Agirre, E. and Rigau, G. A proposal for word sense disambiguation using conceptual distance. *Proceedings of the International Conference "Recent Advances in Natural Language Processing" RANLP'95*, Tzigov Chark, Bulgaria, 1995.

Brill, E. A simple rule-based part of speech tagger. *Proceedings of the 3rd Conference on Applied Natural Language Processing*, Trento, Italy, 1992

Bruce, R. and Wiebe, J. Word sense disambiguation using decomposable models. *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL-94)*, pages 139-146, LasCruces, New Mexico, 1994.

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

Leacock, C.; Chodorow, M. and Miller, G.A. Using Corpus Statistics and WordNet Relations for Sense Identification, *Computational Linguistics vol.24 no.1*, pages 147-165, 1998.

Li, X., Szpakowicz, S. and Matwin, M. A Wordnet-based algorithm for word semantic sense disambiguation. *Proceedings of the 14th International Joint Conference on Artificial Intelligence IJCAI-95*, pages 1368-1374, Montreal, Canada, 1995.

Mihalcea, R. and Moldovan D. A method for Word Sense Disambiguation of unrestricted text *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-99)*, pages 152-158, College Park, MD, 1999.

Miller, G.A., Leacock, C., Randee, T. and Bunker, R. A Semantic Concordance. *Proceedings of the 3rd DARPA Workshop on Human Language Technology*, pages 303-308, Plainsboro, New Jersey. 1993.

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

Moldovan, D. and Mihalcea, R. Using WordNet and lexical operators to improve Internet Searches, *IEEE Internet Computing vol.4 no.1*, pages 34-43, 2000.

Ng, H.T. and Lee, H.B. Integrating multiple knowledge sources to disambiguate word sense: An examplar-based approach. *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL-96)*, pages 40-47, Santa Cruz, 1996.

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

Rigau, G., Atserias, J. and Agirre, E. Combining unsupervised lexical knowledge methods for word sense disambiguation. *Proceedings of joint 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics ACL/EACL'97*, pages 48-55, Madrid, Spain, 1997.

Schutze, H. and Pedersen, J. Information Retrieval based on word senses, *in Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 161-175, 1995.

Voorhees, E. Query expansion using lexical-semantic relations, *Proceedings of the 17th Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval*, pages 61-69, Dublin, Ireland, 1994.

Voorhees, E. Using WordNet for text retrieval. In *WordNet, An Electronic Lexical Database.* The MIT Press, pages 285-303, 1998.

Yarowsky, D. Unsupervised word sense disambiguation rivaling supervised methods. *Proceedings of the 33rd Annual Meeting of the Association of Computational Linguistics (ACL-95)*, pages 189-196, Cambridge, MA, 1995.