Automatic Noun Sense Disambiguation

Paolo Rosso[1], Francesco Masulli[2], Davide Buscaldi[3], and Antonio Molina[1]

[1]Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia (Spain)
{prosso,amolina}@dsic.upv.es
[2]INFM-Genova and Dipartimento di Informatica
Università di Pisa (Italy)
masulli@disi.unige.it
[3]Dipartimento di Informatica e Scienze dell'Informazione
Università di Genova (Italy)
buscaldi@disi.unige.it

Abstract. Word Sense Disambiguation is one of the most important open problems in Natural Language Processing. The absence of sense tagged training data is a real problem for the Word Sense Disambiguation task. This paper explores a fully automatic method which performs the noun sense disambiguation without any kind of training process and relying only on the the WordNet ontology. The knowledge-based method relies on the concepts of semantic relatedness and sense frequency. The noun disambiguation experiments were automatically evaluated against the SemCor, the sense-tagged version of the Brown Corpus. The results are promising: a precision of 81.48% and a recall of approximately 60.17% with a very small context window size of two nouns. The attempt of expanding the context with the gloss of the noun to be disambiguated was also investigated in order to increase recall and coverage measures.

1 Introduction

The task of Word Sense Disambiguation (WSD) consists in examining word tokens and specifying exactly which sense of each word is being used [4]. The WordNet (WN) ontology, based on synsets (*sets* of *syn*onymous), is the external lexical resource which is often used to perform the WSD task [3]. The selection of the proper sense is a non-trivial undertaking given the phenomenon of polysemy. In fact, because of the multiple related meanings, a single word can belong to more than one synset of WN. Systems who carry out the task of semantic tagging, operate in a stand-alone fashion, making minimal assumptions about what information is available from other processes. For instance, from Part-Of-Speech (POS) taggers, they receive information about the syntactic category (noun, verb, adjective or adverb) of a certain word [7].

In most of the approaches to the problem of WSD, a word is disambiguated along with a portion of the text in which it is embedded, that is, its context. When the initial input source of information (i.e., the word and its context) is only processed together with the lexical knowledge source (e.g. WordNet), a fully

automatic method which do not require any kind of training process is needed to perform the word sense disambiguation [1]. In this paper we present a method for the automatic noun sense disambiguation task.

In the second section of this paper, we illustrate the notion of conceptual relatedness among concepts (i.e., WordNet synsets). In Section Three we introduce the new conceptual density formula our automatic method is based on. In the conceptual density formula the frequency of the different meanings of a word (information available in the WN ontology) is also taken into acount. In the fourth section different correction models are applied to our approach in order to improve recall and coverage. The results of the automatic evaluation which was carried out on the SemCor corpus are presented in Section Five. Finally, the last section concludes the work by discussing the results and presenting possible extensions.

2 Conceptual Density

At the basis of our work stands the idea of *Conceptual Density (CD)*. CD is a measure of the correlation among the sense of a given word and its context. The foundation of this measure is the *Conceptual Distance*, defined as the length of the shortest path which connects two concepts in a hierarchical semantic net. The semantic net used for our purposes is WordNet 1.6. The WN ontology is partitioned into three databases, each one associated to different lexical categories (noun, verb, adjective+adverb). A synset constitutes the sense related to one or more words in WordNet (i.e., a polysemic lexeme belongs to more than one synset), and provides pointers to other synsets linked by a lexical relation (hypernymy, hyponymy, meronymy, etc.). In our study we considered only *nouns* and the relations of hypernymy and hyponymy among them. A *synset path*, for a given synset, is the path connecting this synset to the root in WordNet, where each node is a synset, and each edge connects two nodes in a hypernymy/hyponymy relation (if read respectively bottom-up or top-down).

The starting point for our work was the Conceptual Density formula of [1], which compares areas of subhierarchies:

$$CD(c,m) = \frac{\sum_{i=0}^{m-1} nhyp^i}{\sum_{i=0}^{h-1} nhyp^i} \tag{1}$$

where c is the synset at the top of subhierarchy, m the number of words senses falling within a subhierarchy, h the height of the subhierarchy, and $nhyp$ the averaged number of hyponyms for each node (synset) in the subhierarchy. The numerator expresses the expected area for a subhierarchy containing m marks (word senses), while the divisor is the actual area.

The synsets of the senses of the word to be disambiguated fall in different places in the hierarchy, and in most cases this means that the hierarchy can be partitioned in subhierarchies, each containing exactly one sense of the word to be disambiguated (therefore, a word having six senses in WordNet will determine six partitions). We refer to the partitions of the relevant synsets as *clusters*,

though they are not clusters in the strict sense of the term. When two or more senses of the word are one hyponym of each other the partition cannot be done. Therefore, in such conditions the word sense disambiguation cannot be carried out. An example of subhierarchy induced by the synsets paths is shown in Figure 1.

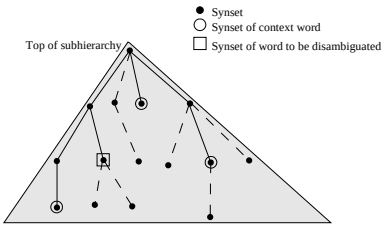

Fig. 1. The subhierarchy induced by the synsets paths

The solid paths are those determined by the word to be disambiguated and its context. The ending nodes of these paths are the *relevant synsets*, the ones needed to evaluate the density of the partition. The dashed paths refer to those synsets contained in the full WordNet hierarchy but that are not relevant for the CD formula. The strategy of Agirre and Rigau [1] was to consider the synsets belonging to the dashed paths too, and the averaged number of hyponyms of each node in the subhierarchy. At first, we also considered the averaged number of hyponyms, that can be viewed as a measure of the sparsity of a subhierarchy. Anyway, due to the fact that the averaged number of hyponyms for each node in the more fine-graned version 1.6 of WordNet is greater than in the version 1.4 (which was used in the original work presented in [1]), we di not obtain the same results. Therefore, we decided not to consider the full subhierarchy determined by the synset c at the top of subhierarchy, but only the part of the subhierarchy determined by the synset paths of the senses of both the word to be disambiguated and its context. The formula is based on the number M of relevant synsets (corresponding to the *marks* m in Formula 1) divided by the total number nh of synsets of the subhierarchy.

$$baseCD(M, nh) = M/nh \tag{2}$$

3 Combining Conceptual Density and Frequency

Figure 2 shows the subhierarchies obtained for the disambiguation of the noun "irregularity" with context nouns "investigation", "Atlanta", "primary-election",

"evidence"[1]. The subhierarchies roots correspond to the following synsets (where offset is a number identifying uniquely a synset): "act" (offset 17487), "quality" (3714294), "property" (3848700) and "psychological-feature" (12865).

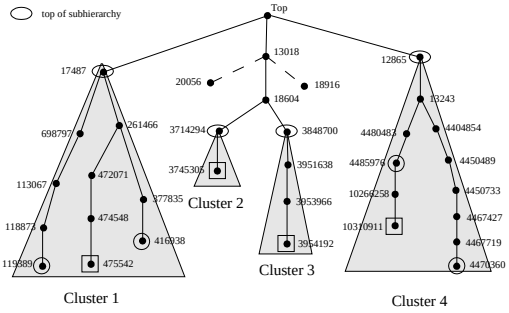

Fig. 2. Subhierarchies for the disambiguation of the noun "irregularity"

The correct sense is the one belonging to the first subhierachy. Let us consider the subhierarchies having context nouns senses inside, that is, the first and the fourth. Their conceptual densities, evaluated with our CD formula, are respectively 0.27 and 0.25. If we look at the second and the third clusters we see that they have a density of respectively 0.50 and 0.25. Therefore, the selected cluster should be (erroneously) the second one. This means that the context is not enough rich to allow a correct disambiguation of the noun. We observed that considering the averaged number of hyponyms for each node, the result was even worse, since the first subhierarchy has more than 100 hyponyms, while the fourth one has only 9. Therefore, the selected subhierarchy was the fourth one, which corresponds to the least frequent sense of the word. This pushed us to improve the CD formula by including also the information about frequency that comes from WordNet:

$$CD(M, nh, f) = M^{\alpha}(baseCD)^{\log f} \tag{3}$$

where M is the number of relevant synsets, α is a constant (the best results were obtained in those experiments with α near 0.10), and f is an integer (ranged 1-25) representing the frequency of the subhierarchy-related sense in WordNet (1 means the most frequent, 2 the second most frequent, etc.). This means that the first sense of the word (i.e., the most frequent) gets at least a density of 1 and one of the less frequent senses will be chosen only if it will exceed the density of the first sense. The M^{α} factor was introduced to give more weigth to the subhierarchies with a greater number of relevant synsets, when the same

[1] sentence of the SemCor file br-a01: "Fulton-County-Grand-jury said Friday an investigation of Atlanta's recent primary-election produced no evidence that any irregularities took-place"

density is obtained among many subhierarchies. With these refinements, the noun "irregularity" of the example is disambiguated correctly. In fact, the first subhierarchy gets a conceptual density of 1.19 whereas the fourth one gets 0.19.

4 Combining Conceptual Density, Frequency and Gloss

In this setion we illustrate the different correction models which were applied to our automatic approach in order to refine the noun sense disambiguation. The results of the evaluation of these techniques are presented in the next section.

4.1 Specific Context Correction and Cluster Depth Correction

In the second section we described that, during the disambiguation of a noun, the hierarchy is partitioned in subhierachies and that our method does not consider each full subhierarchy but only the part determined by the synset paths of the senses of both the noun to be disambiguated and its context (relevants synsets). In our first attempt to improve the performance of our automatic method we included some adjustment factors based on context hyponyms, in order to assign an higher conceptual density to the related cluster in which a context noun is an hyponym of a sense of the noun to be disambiguated (the hyponymy relation reflects a certain correlation between the two lexemes). We refer to this technique as to the *Specific Context Correction (SCC)*. The idea is to select as the winning cluster the one where one or more senses of the context nouns fall beneath the synset of the noun to be disambiguated.

An idea connected to the previous one, was to give more weight to the clusters placed in deeper positions. When a cluster is below a certain depth (that was determined in an empirical way to be about 7) and, therefore, its sense of the noun to be disambiguated is more specific, its conceptual density is augmented porportionally to the number of the contained relevant synsets. We named this technique as *Cluster Depth Correction (CDC)*. The formula which takes into account the CDC correction is:

$$CD * (depth(cl) - avgdepth + 1)^{\beta} \tag{4}$$

where:

- cl is the current cluster;
- $depth(cl)$ returns the depth of cl with respect to the top of the WN noun hierarchy;
- $avgdepth$ is the averaged depth of all clusters in the subhierarchies obtained from Semcor; its value was empirically determined to be equal to 4;
- β is a constant (the best results were obtained in those experiments with β near 0.70).

4.2 Part-Of-Speech Tagging of Noun Gloss

We investigated the possibility of expanding the context with the gloss of the noun to be disambiguated. This led to worse results, since the gloss was examined without considering the syntactic category of its words and a a certain "noise" was introduced as consequence of considering all lexemes as possible nouns. A refinement was done by considering only monosemic words from the gloss, but in spite of that the performance for the noun disambiguation task did not increase. In order to consider only the nouns, we first Part-Of-Speech tagged the gloss.

A *Part-Of-Speech Tagger* assigns the corresponding sequence of POS tags to a sentence. The POS Tagger used in this work is based on Lexicalised Hidden Markov Models (HMM) [6]. This technique consists of incorporating a set of selected words into the Language Model (LM) in addition to the POS tags. These words have be selected empirically from the training set. Although this lexicalisation increases the size of the LM, the performance of the tagging process improves. The lexicalisation process is carried out during the training set, providing a new training set in which a POS-tag s is replaced by the new tag (w, s), if w is tagged with s in the training data set and it was selected as a word to specialise the model.

The parameters of the *Lexicalised-HMM*, transition and output probabilities, can be estimated by Maximum Likelihood from this new training set. The tagging process is carried out by Dynamic Programming Decoding using the Viterbi algorithm.

As the learnig and tagging process are not modified, we used the TnT tagger [2] which is a very efficient statistical POS tagger based on Hidden Markov models. To deal with sparse problems, it uses linear interpolation as smoothing technique to estimate the LM [8]. To handle unknown words, it uses a probabilistic method based on the analysis of the suffix of the words.

The tagger was trained with the part of the Wall Street Journal corpus which was processed in the *Penn Treebank release 2*. The training set consisted of sections 00 to 19 (956,549 words) and the test set included sections 23 and 24 (89,529 words). One of the best lexicalisation criteria, which was reported in [6], is based on the frequency of the words in the training data set. Using the set of words whose frequency was higher than 2000 (about 30 words), the tagger achieved a precision of 96.80% on the test set.

5 Experimental Results

The example of Figure 2 shows the importance of context. In fact, the choice of context words can affect heavily the performance of our automatic method. Therefore, the first goal of our work was to determine in which way we could get an effective context. The first step was to decide if we should work on sentences, on paragraphs, or on the entire text. Sentences are often too short (the worst case is when only one noun, the subject, is found in the sentence), whereas the entire text may not have an homogeneous context (i.e., the context may differ

significantly from a paragraph to another). The choice was to carry out the work on a single paragraph at each time, as context should always be homogeneous within the same paragraph. The second step was to choose a suitable size for the context window. This aspect needed several experiments whose results are shown in Figure 3. These implicitly demonstrate that the choice taken at the previous step (working on paragraphs) was correct, since the values obtained with a window size greater than six were calculated considering the whole text and not the single paragraph (it is quite common that a paragraph has less than six nouns).

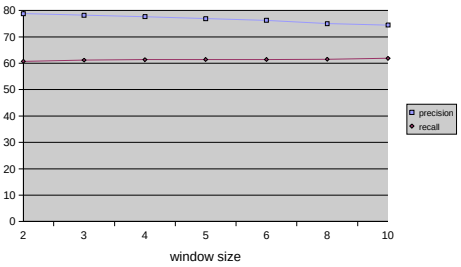

Fig. 3. Precision and recall for different context window sizes

The results were obtained over 19 randomly selected SemCor files[2]. The SemCor (*Sem*antic Conco*cor*dance) [5] is a set of SGML formatted files of the Brown Corpus whose words have been syntactically and semantically tagged with their POS and synset tags. The best results in term of precision[3] were obtained with the smallest window size, confirming that closer nouns give a more precise definition of the context than farther ones. The drawback of this approach is that the average recall[4]: results are fixed around a threshold of about 60%, and vary slightly even when considering many context nouns. This is mainly due to the fact that many nouns have senses that differ slightly one from each other. This can be viewed in a hierarchy as deep clusters with only one synset inside them (corresponding to the sense of the noun to be disambiguated). In most cases, there are no context nouns falling in these "singular" clusters, and the result is that sense disambiguation cannot be done.

We performed a series of tests to obtain a more accurate value for the α parameter used in Formula 3. This work was done after the initial evaluation of context window size which was conducted with a fixed alpha value of 0.25. The examined values were in the interval [0, 1] and the model selected for these tests was the one with window size 4 (which obtained a good balance in terms

[2] br-a01,b13,c01,d02,e22,r05,g14,h21,j01,k01,k11,l09,m02,n05,p07,r04,r06,r08,r09

[3] *precision* = number of correctly disambiguated nouns / number of disambiguated nouns

[4] *recall* = number of correctly disambiguated nouns / number of nouns in the files

of precision / recall). The best performance was obtained with the value 0.10. Similar tests were performed to tune the β parameter used in Formula 4 of the cluster depth correction technique. The best performance was obtained with the value 0.70. In order to understand how effective the CDC technique was, some experiments were carried out for different depth values obtaining that the deeper the cluster is the higher is the precision obtained using the CDC techniques.

Figure 4 shows the results obtained combining different correction models (specific context correction and cluster depth correction) over the whole brown (1 and 2) files of SemCor corpus and for different window sizes (two, four and six). The baseline precision was calculated assigning the most frequent sense to every noun, whereas the baseline recall (1) was calculated for monosemic nouns. The best precision measure of 81.48% was obtained without any correction factor and with a very small window of size two (recall 60.17% and coverage[5] 73.81%). Using the SCC technique, although precision was not affected significantly, we obtained only small improvements on recall and coverage measures. With regard to the CDC technique, the results did not differ significantly to those obtained with the previous correction factor. Improvements on recall (61.27%) and coverage (77.87%) measures are obtained increasing the size of the context window. Recall remains approximately around a threshold of about 60% and vary slightly even when considering many context nouns (e.g. six), whereas coverage improves even if at the price of obtaining a lower precision measure.

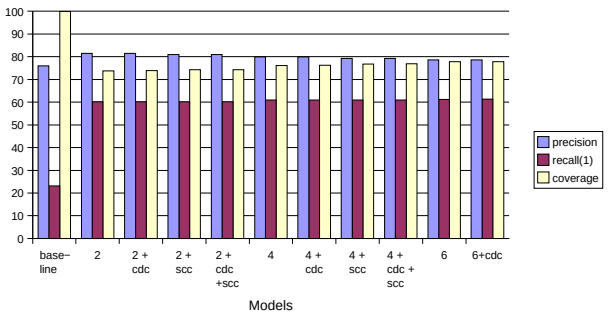

Fig. 4. Precision, recall and coverage obtained with the different correction models

For each noun to be disambiguated, we investigated the possibility of expanding its context adding the gloss, once cleaned from example phrases. In the first approximation we considered all words of the gloss as nouns and this led us to a worse precision. In the second approximation, in order to reduce the "noise" introduced considering all the words of the gloss, only monosemic words were added to the context of the noun to be disambiguated. Finally, we POS-tagged the words of the gloss and extracted only its monosemic nouns which were in-

[5] *coverage* = number of disambiguated nouns / number of nouns (brown 1-2 files)

cluded in the context. The tests with this "expanded context" were conducted over the first 10 files from brown1 of SemCor. The cluster depth correction factor was also employed. Table 1 shows P(recision), R(ecall) and C(overage) for the three models:

1. model with POS-tagged gloss and CDC technique
2. model with gloss and CDC technique
3. model with CDC technique and no gloss

In order to have a certain balance in terms of precision / recall, a window size of 4 (previous to its expansion with the monosemic nouns of the gloss) was used for the experiments. The size of the expanded context was 5.92 on average (i.e., it contained 6 nouns approximately).

Without POS-tagging the gloss, even considering only its monosemic words, the recall decreased slowly and the precision decreased by an average of more than 2% with respect to the precision obtained without the gloss. The POS-tagging preprocess of the gloss permitted to obtain improvements both on recall and coverage without practically losing in precision.

file	P1	P2	P3	R1	R2	R3	C1	C2	C3
a01	81.97	80.38	81.97	69.02	69.70	68.87	84.19	86.72	84.02
a02	85.01	82.57	85.37	68.68	68.26	68.26	80.79	82.67	79.95
a11	82.90	82.05	82.84	63.45	63.20	63.20	76.54	77.03	76.29
a12	79.20	74.83	79.53	56.47	53.88	55.76	71.29	72.00	70.11
a13	84.69	84.12	84.98	64.17	64.17	64.17	75.77	76.28	75.51
a14	79.73	77.88	79.66	60.60	59.59	60.35	76.01	76.51	75.75
a15	76.58	72.48	76.80	55.14	53.42	54.84	72.00	73.71	71.42
b13	78.81	74.72	79.03	60.38	59.52	60.38	76.62	79.65	76.40
b20	81.06	78.77	81.00	63.54	63.80	63.28	78.38	80.98	78.12
c01	77.71	76.40	77.94	62.77	63.01	62.77	80.77	82.48	80.53
average	80.77	78.42	80.91	62.42	61.86	62.19	77.24	78.80	76.81

Table 1. Precision, recall and coverage measures with and without (POS-tagged) gloss

6 Conclusions and Further Work

The automatic method for the disambiguation of nouns presented in this paper can be used on previously POS-tagged text of any general domain. It uses WordNet sense tags and it does not need any training. The algorithm is based on a general measure of semantic relatedness for nouns which considers also sense frequency, sense depth in the WordNet ontology and POS-tagged gloss.

The algorithm disambiguated the text of brown 1 and 2 of the SemCor corpus. The results were obtained automatically comparing the tags in SemCor with those computed by the algorithm. The results are promising if we compare them

to those obtained using the original conceptual density formula (precision 81.97% vs. 66.4% and recall 69.02% vs. 58.8%)[6] especially if we consider that the much more fine-grained 1.6 version of WordNet was used.

At the moment of writing this paper more extensive experiments, which include POS-tagging the gloss of all WordNet nouns, are under way. With these experiments we would like to fully evaluate the effect of extending the context of the noun to disambiguate with its gloss.

Further work needs to be done to perform the all-word disambiguation task and the evaluation of the methods against the Senseval corpus. The final aim is the application of this WSD technique to the disambiguation of Spanish and Italian using the EuroWordNet ontology and, finally, the integration of the WSD tool into NLP systems and their application to conceptual information retrieval problems.

Acknowledgements

The work of Paolo Rosso and Antonio Molina was partially supported by the Spanish Research Department (CICYT project TIC2000- 0664-C02). The research period of Paolo Rosso at the University of Genova (Italy) was funded by a research grant of the Vicerrectorado de Investigación, Desarrollo e Innovación (UPV).

References

1. Agirre, E. and Rigau, G., 1996. A Proposal for Word Sense Disambiguation using Conceptual Distance. In Proceedings of the International Conference on Recents Advances in Natural Language Processing.
2. Brants, T., 2000. TnT - a Statistical Part-of-Speech Tagger. In Proceedings of the Sixth Applied Natural Language Processing, Seattle, WA.
3. Fellbaum, C. (Ed.), 1998. WordNet: An Electronic Lexical Database. MIT Press, Cambridge, MA, USA.
4. Jurafsky, D. and Martin, J., 2000. Speech and Language Processing. Prentince Hall.
5. Landes, S., Leacock., C. and Tengi, R.I., 1998. Building Semantic Concordance. In Fellbaum C. (Ed.), WordNet: An Electronic Lexical Database, pp. 199-216, MIT Press, Cambridge, MA, USA.
6. Pla, F. and Molina, A., 2001. Part-of-Speech Tagging with Lexicalized HMM. In Proceedings of International Conference on Recents Advances in Natural Language Processing, Tzigov Chark, Bulgaria.
7. Stevenson, M. and Wilks, Y., 2001. The Interaction of Knowledge Sources in Word Sense Disambiguation. Computational Linguistic, 3(27), pp. 321-349, September.
8. Young, S. and Bloothooft, G. , 1997. Corpus-based Methods in Language and Speech Pocessing. ELSNET book edition.

[6] results obtained for the file br-a01 of SemCor (2079 words long)