Sense Discriminative Patterns for Word Sense Disambiguation

Octavian Popescu
FBK-irst, Trento (Italy)
popescu@itc.it

Bernardo Magnini
FBK-irst, Trento (Italy)
magnini@itc.it

Abstract

Given a target word w_i to be disambiguated, we define a class of local contexts for w_i such that the sense of w_i is univocally determined. We call such local contexts *sense discriminative* and represent them with sense discriminative (SD) patterns of lexico-syntactic features. We describe an algorithm for the automatic acquisition of minimal SD patterns based on training data in SemCor.
We have tested the effectiveness of the approach on a set of 30 highly ambiguous verbs. Results compare favourably with the ones produced by a SVM word sense disambiguation system based on bag of words.

1 Introduction

Leacock, Towell and Voorhes (1993) distinguish two types of contexts for a target word w_i to be disambiguated: a *local context*, which is determined by information on word order, distance and syntactic structure and is not restricted to open-class words, and a *topical context*, which is the list of those words that are likely to co-occur with a particular sense of w_i.
Several recent approaches to Word Sense Disambiguation (WSD) take advantage of the fact that the words the surrounding a target word w_i provide clues for its disambiguation. A number of syntactic and semantic features in a local context [w_{i-n}, … w_{i-1}, w_i, w_{i+1},… w_{i+n}] (where n is usually not higher than 3) are considered, including the token itself, the Part of Speech, the lemma, the semantic domain of the word, syntactic relations and semantic concepts. Results in supervised WSD (see, among the others, Yarowsy 1992, Pederson 1998, Ng&Lee 2002) show that a combination of such features is effective.

We think that the potential of local context information for WSD has not been fully exploited by previous approaches. In particular, this paper addresses the following issues:

1. As our main interest is WSD, we are interested in local contexts which univocally select a sense s_j of w_i. We call such contexts "sense discriminative" and we represent them as *sense discriminative* (SD) patterns of lexico-syntactic features. According to the definition, if a SD pattern matches a portion of the text, then the sense of the target word w_i is univocally determined. We propose a methodology for automatically acquiring SD patterns on a large scale.
2. Intuitively, the size of a local context should vary depending on w_i. For instance, if w_i is a verb, a preposition appearing at w_{i+3} may introduce an adjunct argument, which is relevant for selecting a particular sense of w_i. The same preposition at w_{i+3} may cause just a noise if w_i is an adjective. We propose that the size of the local context C, relevant for selecting a sense s_j of w_i, is dynamically set up, such that C is the minimal context for univocally selecting s_j.
3. An important property of some minimal SD patterns is that each element of the pattern has a specific meaning, which does not change when new words are added. As a consequence, all the words $w_{i+/-n}$ are disambiguated. We call the relations that determine a single sense for each element of a minimal sense discriminative pattern *chain clarifying relationships*. The acquisition method we propose is crucially based on this property.

According to the above mentioned premises the present paper has two goals: (i) design an algorithm for the automatic acquisition of minimal sense discriminative patterns; (ii) evaluate the patterns in a WSD task.
With respect to acquisition, our method is based on the identification of the minimal set of lexico-syntactic features that allow the discrimination of a sense for w_i with respect to the other senses of the word. The algorithm is trained on a sense tagged corpus (experiments have been carried on SemCor) and starts with a dependency-based representation of the syntactic relations in the sentence containing w_i. Then, elements of the sentence that do not bring sense discriminative information are filtered out; we thus obtain a minimal SD pattern.
As for evaluation, we have tested sense discriminative patterns on a set of thirty high polysemous verbs in SemCor. The underlying hypothesis is that SD patterns are effective in particular in the case of the scarcity of the training data. We provide a comparison of the SD-based disambiguation with a simple SVM-based system, and we show that our system fares significantly higher in performance.

The paper is organized as follows. Section 2 introduces sense discriminative patterns and chain clarifying relations in a more formal way. In Section 3 we present the algorithm we have used to identify sense discriminative contexts starting from a sense annotated corpus. In Section 4 we present the results we have obtained applying SD patterns on a WSD task and we compare them against a supervised WSD system based on SVM and the bag of word approach. In section five we review related works and point out the novelty of our approach. We conclude with section six, in which we present our conclusions and directions for further research.

2 Chain Clarifying Relationships (CCR)

Consider the examples below:

1a) *He drove the girl to her father/to the church/ to the institute/to L.A.*

1b) *He drove the girl to ecstasy/to craziness/ to despair/ to euphoria.*

Using a sense repository, such as WordNet 1.6, we can assign a sense to any of the words in both 1a) and 1b). In 1a) the word "drive" has the sense drive#3, "cause someone or something to move by driving" and in 1b) it has the sense drive#5, "to compel or force or urge relentlessly or exert coercive pressure on". By comparing 1a) and 1b) and by consulting an ontology, we can identify a particular feature which characterizes the prepositional complements in 1b), and which we hold responsible for the sense of "drive" in this sentence. The relationship between this feature and the sense of "drive" holds only in the common context of 1a) and 1b), namely the prepositional complement. Example 2) below shows that if this local context is not present, then the word "euphoria" does not have a disambiguating function for "drive".

2) *He drove the girl back home in a state of euphoria.*

However, the syntactic configuration alone does not suffice, because lexical features must be taken into account, too. The particular sense combination is determined by a chain-like relationship: the sense of "girl" is determined by its function as object of the verb "drive"; the sense of "drive" is determined by the nature of the prepositional complement. We call such relationship a chain clarifying relationship (CCR). The importance of CCRs for WSD resides in the fact that by knowing the sense of one component, specific senses are forced for the others components.

In what follows we give a formal definition of the CCR, which will help us to device an algorithm for finding CCR contexts. We start from the primitive notion of *event* (Giorgi and Pianesi, 1997). We assume that there is a set:

$$E=\{e_1, e_2, \ldots e_n\}$$

whose elements are events, and that each event can be described by a sequence of words. Let us now consider three finite sets, *W*, *S* and *G*, where:

$$W = (w_1, w_2, \ldots w_w)$$

is the set of words used to describe events in *E*,

$$S =(w_{11}, w_{12}, \ldots, w_{1m1}, w_{21}, w_{22},\ldots w_{2\,m2}, \ldots w_{wmw})$$

is the set of words with senses, and

$$G=(g_1, g_2, \ldots g_{mg})$$

is the set of grammatical relations.

If *e* is an event described with words w_1, w_2, …w_n we assume that *e* assigns a sense $w_{i\,j}$ and a grammatical relation g_i to any of these words. Therefore we consider *e* to be the function:

$$e:\ P(\{w_1, w_2, \ldots w_w\}) \rightarrow (SxG)^n$$

$$e(w_1, w_2, \ldots w_n) = (w_{1i1}xg_{i1}, w_{2i2}xg_{i2}, \ldots w_{nin}xg_{in})$$

For a given *k* and *l*, such that $1 \leq k \leq l \leq n$, and *k* components of $e(w_1, w_2, \ldots w_n)$ we call the *chain clarifying relation* (*CCR)* of *e* the function:

$$e_{CCR}: (SxG)^{n-k} \text{ x } (WxG)^k \rightarrow (SxG)^l$$

where $e_{CCR}(w_{1i1}xg_{i1}, w_{2i2}xg_{i2}, \ldots w_{kik}xg_{ik}, w_{k+1}xg_{k+1}, w_{k+2}xg_{i2}, \ldots w_nxg_{in}) = (w_{1i1}, w_{2i2}, \ldots w_{lil})$

The above definition captures the intuition that in certain contexts the senses of some of the words impose a restriction on the senses of other words. When $l=n$ we have a complete sense specification, therefore the e_{CCR} function gives a sense for any of the words of *e*.

Let us consider two events *e* and *e'* such that they differ only with respect to two slots:

$e(w_1, w_2, \ldots w_n)=(w_{1i1}xg_{i1}, w_{2i2}xg_{i2}, w_{kik}xg_{ik} \ldots w_{nin}xg_{in})$

$e'(w_{1'}, w_2, \ldots w_n.)=(w'_{1i1}xg_{i1}, w_{2i2}xg_{i2}, \ldots w_{kik'}xg_{ik} \ldots w_{n,in}xg_{in})$.

We infer that there is a lexical difference between w_1 and w_1' which is responsible for the sense difference between w_{kik} and $w_{kik'}$. If precisely this difference is found to be preserved for any $e(w_1,w_2,..,w_n, w_{n+1},w_{n+2},\ldots,w_m)$, then the sequence $(w_{1i1}xg_{i1}, w_{2i2}xg_{i2}, \ldots w_{kik-1}xg_{ik-1,} w_{k,ik+1}xg_{ik+1}\ldots w_{nin}xg_{in})$ is a CCR.

The examples in 1a) are local contexts having the sense constancy property in which a particular type of CCR holds. We can express a CCR under the shape of a pattern, which, by the way in which it has been determined, represents a sense discriminative (SD) pattern. A SD pattern classifies the words that fulfill its elements in classes which are valid only with respect to a particular CCR. A simple partitioning of the nouns, for example, in semantic classes independently of a CCR may not lead to correct predictions. On the one hand, a semantic class which includes "father" and "church" may be misleading with respect to their senses in 1a), and, on the other hand, a semantic class which includes "father", "church", "institute", "L.A" is probably too vague. This suggests that rather than starting with a set of predefined features and syntactic frames, it is more useful to discover these on the basis of an investigation of sense constancy. Also, there is not a strictly one to one relationship between predicate argument structure and CCR: as our experiments showed, there are cases when only some complements or adjuncts in the sentence play an active role in disambiguation.

3 Acquisition of SD Patterns

The algorithm we have used for the acquisition of SD patterns consists mainly in two steps: first, for each sense of a verb, all the potential CCRs are extracted from a sense annotated corpus; second, all the patterns which are not sense discriminative are removed.
In accordance with the definition of CCRs, we have tried to find CCRs for verbs by considering only the words that have a dependency relationship with the verbs. Our working hypothesis is that we may find valid CCRs only by taking into account the external and internal arguments of the verbs. Thus we have considered the dependency chains (DC) rooted in verbs.

3.1 Finding Dependency Chains

In a dependency grammar (Mel'čuk 1988) the syntactic structure of a sentence is represented in terms of dependencies between words. The dependency relationships are between a *head* and a *modifier* and are of the type one to many: a head may have many modifiers but there is only one head for each modifier. The same word may be a head or a modifier of some other words; thus the dependency relationships constitute subtrees. Here we are interested mainly in finding the subtrees rooted in predicative verbs.

After running a set of tests in order to check the accuracy of various parsers, (i.e. Lin 1998, Bikel 2004) we have decided to use the Charniak's parser which is a constituency parser. The choice was determined by the fact that the VP constituents were determined with accuracy

below 70% by the other parsers. In order to extract the dependency relationships from the Charniak's parser output we have relied on previous work on heuristics for finding the heads of the NP constituents and their types of dependency relationships (see, among others, Ratnaparkhi, 1997; Collins, 1999).

3.2. SD Patterns Selection

The extraction of CCRs is an iterative process that starts with the dependency trees for a particular sense of a word. The algorithm builds at each step new candidates through a process of generalization of the entities that fulfil the syntactic slots of a pattern. The candidates which are not sense discriminative are discarded and the process goes on till there are no new candidates.

We start with the dependency chains rooted in verbs extracted from a sense tagged corpus. For each verb sense, the dependency chains are clustered according to their syntactic structure. Initially, all dependency chains are considered candidates. Chains that are found in at least two cluster are removed. After this "remove" procedure, since each chain individuates a unique sense combination, in each cluster remain only the patterns which are SD patterns according to the training examples.

In order to find the minimal SD patterns we build minimal SD candidates from the existing patterns by means of a process of generalization. Inside each cluster, we search for similarities among the entities that fulfil a particular slot. For this purpose we use SUMO (Niles& all 2003), an ontology aligned to WordNet. Two or more entities are deemed to be similar if they share the same SUMO attribute. Similar entities are "generalized" by the common attribute. Then, all the patterns that have similar entities in the same slot and are identical with respect to all the other slots are collapsed into one new candidate. The algorithm repeats the remove procedure for the new candidates; the ones that pass are considered SD patterns. We stop when no new candidates are proposed.

For example the sentences in 1b) lead to to the following minimal SD pattern for the sense 3 of the verb drive:

(V=drive#3 S=[Human], O=[Human] P=to PP_1 =[EmotionalState])

4. Experiments

We have designed an experiment in order to evaluate the effectiveness of the SD patterns approach. We have chosen a set of thirty highly polysemic verbs which are listed in Table 1.

4.1 Training and Test Data

Since the quality of SD patterns is directly correlated with the accuracy of DCs, we have decided to extract the verb rooted DCs from a hand annotated corpus. For training, we considered the part of the Brown corpus which is also a part of the Penn Tree Bank. In this corpus verbs are annotated with the senses of WordNet and all sentences are parsed. For a part of the corpus we have annotated the senses of the nouns which are heads of the verbs' internal and external arguments and we have written a Perl script which transforms the parsed trees into dependency trees. Because in the Penn Tree bank the grammatical function is given, this transformation is accurate.

Some of the senses of the test verbs have only a few occurrences. In order to have a better coverage of less frequent senses we added new examples, such that there are at least ten examples per each verb sense. These new examples are simplified instances of sentences from the BNC. They are made up only from the subject and the respective VP as it appears in the original sentences. The subject has been explicitly written in the cases where in the original sentence there is a trace or a relative pronoun. We parsed them with the Charniak's Parser and we extracted the dependency chains. We manually checked 140 of them and we found 98% accuracy.

The second column of Table 1 represents the number of occurrences of test verbs in the corpus common to the Brown and to the Tree Bank. The third column represents the number of examples for which we have annotated the arguments. The forth column represents the number of the added examples. In the fifth column we list the number of patterns we found in the training corpus for each verb. In the sixth and in the seventh columns we list the minimum and the maximum number of patterns respectively. Number 0 as minimum means that there was no way to find a difference between at least two senses. The test corpus was the part of Brown corpus which is generally known as Semcor.

4.2 Results and Discussion

We compared the results we obtained with SD patterns against a SVM-based WSD system. For each word in a local context, features were the lemma of the word, its PoS, and its relative distance from the target word. The training corpus for the SVM was formed by all the sentences from the common part of the Brown and the Pen Tree Bank corpora and the new added examples from the BNC. Therefore, the training corpus for the SVM includes the training corpus for SD patterns (more than 1000 examples in addition for SVM system).

verb	#occ	#tag	#add	#pat	#min	#max	verb	#occ	#tag	#add	#pat	#min	#max
begin	188	80	3	12	2	3	match	18	18	30	8	0	3
call	108	80	40	25	1	8	move	118	90	40	29	2	8
carry	68	68	40	32	1	6	play	121	80	40	29	0	5
come	317	100	30	36	1	9	pull	24	24	20	13	1	3
develop	80	60	20	17	0	3	run	97	90	50	42	0	11
draw	40	40	60	38	1	3	see	445	120	30	36	0	8
dress	10	10	30	7	1	3	serve	112	70	10	14	1	3
drive	72	40	40	14	1	5	strike	37	37	20	9	1	3
face	66	40	10	9	0	3	train	13	13	40	14	1	4
find	254	100	20	26	0	7	treat	34	34	10	11	0	4
fly	27	27	10	16	1	6	turn	85	40	40	16	1	3
go	229	100	20	35	0	12	use	291	60	40	21	2	5
keep	166	70	30	28	2	8	wander	8	8	10	4	1	3
leave	167	100	30	31	1	9	wash	1	1	30	8	0	3
live	124	70	10	11	1	3	work	120	80	30	24	1	6

Table 1: Training corpus for SD patterns.

The second column of Table 2 lists the total number of the occurrences of the test verbs in Semcor. In the third column we list the results obtained using SD patterns and in the fourth the results obtained using the SVM system. The number of senses the in corpus, which are found by each approach, are listed in the fifth and sixth column respectively. The SD patterns approach has scored better than SVM, 49.32% vs. 42.28%.

verb	#occ	#SDP	#SVM	#senses SDPS	#senses SVM	verb	#occ	#SDP	#SVM	#senses SDPS	#senses SVM
begin	203	178	135	5	3	match	31	14	10	3	1
call	148	73	52	8	6	move	137	61	46	7	5
carry	77	41	29	10	6	play	181	87	61	11	6
come	354	184	130	9	5	pull	46	26	28	4	2
develop	114	42	28	7	4	run	131	72	30	17	5
draw	73	35	16	9	6	see	578	213	259	15	8
dress	36	18	21	3	1	serve	98	39	42	10	8
drive	68	23	21	5	3	strike	43	17	13	8	4
face	196	58	62	4	2	train	47	23	27	4	1
find	420	204	97	6	7	treat	48	13	9	3	1
fly	30	22	15	4	1	turn	130	63	74	8	3
go	256	171	125	13	4	use	439	199	356	4	1
keep	153	103	86	8	4	wander	8	3	5	2	1
leave	222	121	83	10	6	wash	39	20	21	3	2
live	120	45	57	4	3	work	344	185	79	9	5

Table 2: Comparative results for using SD patterns and SVM bag of word in WSD.

The range of the senses the SD patterns approach is able to identify is more than two times greater than the SVM system.
We also show how these two approaches perform in the cases of the less frequent senses in the corpus. Table 3, second column, reports the number of senses considered, the third, the cumulative number of occurrences in the test corpus; the fourth and the fifth columns, report the correct matching for SD patterns and for SVM. Results for SD patterns are higher than the ones obtained with SVM: 34.72% vs.13.74%.

The patterns we have obtained are generally very precise: they identify the correct sense with more than 85% accuracy. However, they are not error proof. We believe there are mainly three reasons for why the SD patterns lead to wrong predictions: (i) the approximation of CCRs with DCs, (ii) the parser accuracy, and (iii) the relative small size of the training corpus. The CCRs are determined only considering the words that have a direct dependency relationship with the target word. However, in some cases, the information which allows word disambiguation may be beyond phrase level (Wilks&Stevenson, 1997 – 2001). The parser accuracy plays an important role in our methodology. While the method of considering only simple sentences in the training phase seems to produce good results, further improvements are required. Finally, the dimension and the diversity of sentences in the training corpus play an important role for the final result. The smaller and the more homogenous the training corpus is, the bigger the probability that a DC, which is not a SD pattern, is considered erroneously as such.

In some cases, such as semantically transparent nouns (Fillmore et al. 2002), the information which allows the correct disambiguation of the nouns that are heads of NPs, is found within the NPs. Our approach cannot handle these cases. Our estimation is that they are not very frequent, but, nevertheless, a proper treatment of such nouns contributes to an increase in accuracy.

verb	#senses	#occ	#SDP	SVM	verb	#senses	#occ	#SDP	SVM
begin	2	11	8	5	match	3	7	1	0
call	3	10	5	2	move	6	26	10	4
carry	12	30	13	4	play	13	31	16	2
come	7	20	9	2	pull	5	17	5	2
develop	10	33	13	3	run	20	46	16	6
draw	20	73	35	16	see	10	40	3	2
dress	3	13	3	2	serve	7	27	12	8
drive	5	16	4	1	strike	8	17	8	4
face	4	16	2	0	train	5	14	3	0
find	2	14	4	1	treat	1	7	2	0
fly	5	9	5	2	turn	11	31	7	4
go	14	45	14	5	use	4	19	2	2
keep	9	24	10	3	wander	2	8	4	5
leave	11	58	22	7	wash	2	9	3	3
live	3	13	2	0	work	10	34	9	3

Table 3: Results for less frequent senses.

5. Related Works

Based on the Harris' Distributional Hypothesis (HDH), many approaches to WSD have focused on the contexts formed by the words surrounding the target word . With respect with verb behaviour, selectional restrictions have been used in WSD (see among others Resnik 1997, McCarthy, Caroll, Preis 2001, Briscoe 2001). Also, Hindle (Hindle 1990) has tried to classify the English nouns in similarity classes by using a mutual information measure with respect to the subject and object roles. Such information is very useful only in certain cases and, as such, it might not be used directly for doing WSD.

Lin and Pantel (Lin,Pantel 2001) transpose the HDH from words to dependency trees. However, their measure of similarity is based on a frequency measure. They maintain that a (slotX, he) is less indicative that a (slotX, sheriff). While this might be true in some cases, the measure of similarity is given by the behaviour of the other components of the contexts: both "he" and "sheriff" act either exactly the same with respect to certain verb meanings, or totally different with respect to some others. A classification of these cases is obviously of great importance for WSD. However, this classification problem cannot be addressed by employing the method the authors present. The same arguments are also valid in connection with the method proposed by Li&Abe, based on MDL (Li&Abe 1998). Another limitation of these methods, which our proposal overcomes, is that they only consider subject and object positions. However, in many cases the relevant entities are complements, and/or prepositions and particles. It has been shown that closed class categories, especially preposition and particles, play an important role in disambiguation and wrong prediction are made if they are not taken into account. (see, among others, Collins and Brooks 1995, Stetina&Nagao 1997). Our results have shown that only a small fraction (27%) of SD patterns include just the subject and/or the object.

Zhao, Meyers and Grishman (Zhao, Meyers and Grishman 2004, Zhao) proposed a SVM application to slot detection, which combines two different kernels, one of them being defined on dependency trees. Their method tries to identify the possible fillers for an event, but it does not attempt to treat ambiguous cases; also, the matching score algorithm makes no distinction between the importance of the words, considering equal matching score for any word within two levels.

Pederson and al. (1997-2005) have clustered together the examples that represent similar contexts for WSD. However, given that they adopt mainly the methodology of ordered pairs of bigrams of substantive words, their technique works only at the word level, which may lead to a data sparseness problem. Ignoring syntactic clues may increase the level of noise, as there is no control over the relevance of a bigram.

Many of the purely syntactic methods have considered the properties of the subcategorization frame of verbs. Verbs have been partitioned in semantic classes based mainly on Levin's classes of alternation. (Dorr&Jones 1996, Palmer&all 1998-2005, Collins, McCarthy, Korhonen 2002, Lapata&Brew 2004). These semantic classes might be used in WSD via a process of alignment with hierarchies of concepts as defined in sense repository resources (Shin&Mihalcea 2005). However the problem of the consistency of alignment is still an open issue and further research must be pursued before applying these methods to WSD.

6. Conclusion and Further Research

We have presented a method for determining a particular type of local context, within which the relevant entities for WSD can be discovered. Our experiment has shown that it is possible to represent such contexts as Sense Discriminative patterns. The results we obtained applying this method to WSD compare favourably with other results.

One of the major limitations in achieving higher results is the small size of the training corpus. The quality of SD patterns depends to a great extend on the variety of examples in the training corpora.

The CCR property of some local context allows a bootstrapping procedure in the acquisition of SD patterns. This remains an issue for further research.

The SD patterns for verbs, characterize the behaviour of words which constitute a VP phrase with respect to the word senses. In fact, to each pattern corresponds a regular expression. Thus a decision list algorithm could be implemented in order to optimize the matching procedure.

References

Brew, L., 2004, "Verb class disambiguation using informative priors Computational Linguistics", Volume 30, pages: 45 – 73.

Briscoe, T., 2001, "From Dictionary to Corpus to Self-Organizing Dictionary: Learning, Valency Associations in the Face of Variation and Change", In Proceedings of Corpus, Linguistics. Lancaster University, UK.

Caroll J., Briscoe T., 2001 "High precision extraction of grammatical relations", Workshop on Parsing Technologies, Beijing.

Collins M., Brooks J., 1995. “Prepositional phrase attachment through a backed-off model”. In Proceedings of the Third Workshop on Very Large Corpora, pages 27--38, Cambridge.

Collins, M. 1999, “Head-Driven Statistical Models for Natural Language Parsing” Ph.D. thesis, University of Pennsylvania.

Dorr, B., Jones, D., 1999“Acquisition of Semantic Lexicons in Breadth and Depth of Semantic Lexicons, edited by Evelyne Viegas. Kluwer Press. .

Hindle, D., 1990, “Noun classification from predicate argument structures”, In Proceedings of the Annual Meeting of the Association for Computational Linguistics, pages 268--275.

Fillmore, C., Baker, C. and Sato, Hiroaki, 2002: “Seeing Arguments through Transparent Structures”. In Proceedings of the Third International Conference on Language Resources and Evaluation (LREC). Las Palmas. 787-91

Korhonen, A., 2002. “Subcategorization Acquisition”, PhD thesis published as Techical Report UCAM-CL-TR-530. Computer Laboratory

Leacock, C., Towell, G., & Voorhes, E., “Towards Building Contextual Representations of Word Senses Using Statistical Models”, In Proceedings, SIGLEX workshop: Acquisition of Lexical Knowledge from Text, ACL., 1993.

Lee, Y., Ng, H., 2002, “An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation”, In Proceedings of EMNLP’02, pages 41–48, Philadelphia, PA, USA.

Li, D., Abe, N. 1998, ”Word Clustering and Disambiguation Based on Co-occurence Data”. COLING-ACL : 749-755.

Lin, D., Pantel, P., 2001, “Discovery of Inference Rules for Question Answering”, Natural Language Engineering 7(4):343-360.

McCarthy, D., Carroll, J. and Preiss, J. (2001) “Disambiguating noun and verb senses using automatically acquired selectional preferences”, In Proceedings of the SENSEVAL-2 Workshop at ACL/EACL'01 , Toulouse, France.

Ratnaparkhi, A., 1997. A Linear Observed Time Statistical Parser Based on Maximum Entropy Models. In Proceedings of the Second Conference on Empirical Methods in Natural LanguageProcessing.

Dang,T., Kipper, K., Palmer, K., Rosenzweig, J., “Investigating regular sense extensions based on intersective Levin classes”,. Coling-ACL98 , Montreal CA, August 11-17, 1998.

Pederson, T., 1998, “Learning Probabilistic Models of Word Sense Disambiguation “, Southern Methodist University, 197 pages (PhD Dissertation)

Pederson T., 2005, “SenseClusters: Unsupervised Clustering and Labeling of Similar Contexts” , Proceedings of the Demonstration and Interactive Poster Session of the 43rd Annual Meeting of the Association for Computational Linguistics.

Resnik,P. 1997, “Selectional Preference y Sense Disambiguation, in Proceedings of the SIGLEX WorkShop Tagging Text with Lexical Semantics: Why, What y How?.” Washington.

Shi, L., Mihalcea, R., 2005, “Putting Pieces Together: Combining FrameNet, VerbNet and WordNet for Robust Semantic Parsing”, in Proceedings of the Sixth International Conference on Intelligent Text Processing and Computational Linguistics, Mexico.

Stevenson K., Wilks., Y.,.2001 “The interaction of knowledge sources in word sense disambiguation”, Computational Linguistics, 27(3):321–349.

Zhao,S., Meyers A., and Grishman, R., 2004 , Proceedings of the 20th International Conference on Computational Linguistices Geneva, Switzerland.

Stetina J, Nagao M 1997 “Corpus based PP attachment ambiguity resolution with a semantic dictionary.” In Zhou J, Church K (eds), Proc. of the 5th Workshop on very large corpora, Beijing and Hongkong, pp 66-80.

Yarowsky, D. 1992. “Word-sense disambiguation using statistical models of Roget's categories trained on large corpora”. In COLING-92.