Learning Expressive Models for Word Sense Disambiguation

Lucia Specia
NILC/ICMC
University of São Paulo
Caixa Postal 668, 13560-970
São Carlos, SP, Brazil
lspecia@icmc.usp.br

Mark Stevenson
Department of Computer Science
University of Sheffield
Regent Court, 211 Portobello St.
Sheffield, S1 4DP, UK
marks@dcs.shef.ac.uk

Maria das Graças V. Nunes
NILC/ICMC
University of São Paulo
Caixa Postal 668, 13560-970
São Carlos, SP, Brazil
gracan@icmc.usp.br

Abstract

We present a novel approach to the word sense disambiguation problem which makes use of corpus-based evidence combined with background knowledge. Employing an inductive logic programming algorithm, the approach generates expressive disambiguation rules which exploit several knowledge sources and can also model relations between them. The approach is evaluated in two tasks: identification of the correct translation for a set of highly ambiguous verbs in English-Portuguese translation and disambiguation of verbs from the Senseval-3 lexical sample task. The average accuracy obtained for the multilingual task outperforms the other machine learning techniques investigated. In the monolingual task, the approach performs as well as the state-of-the-art systems which reported results for the same set of verbs.

1 Introduction

Word Sense Disambiguation (WSD) is concerned with the identification of the meaning of ambiguous words in context. For example, among the possible senses of the verb "run" are "to move fast by using one's feet" and "to direct or control". WSD can be useful for many applications, including information retrieval, information extraction and machine translation. Sense ambiguity has been recognized as one of the most important obstacles to successful language understanding since the early 1960's and many techniques have been proposed to solve the problem. Recent approaches focus on the use of various lexical resources and corpus-based techniques in order to avoid the substantial effort required to codify linguistic knowledge. These approaches have shown good results; particularly those using supervised learning (see Mihalcea et al., 2004 for an overview of state-of-the-art systems). However, current approaches rely on limited knowledge representation and modeling techniques: traditional machine learning algorithms and attribute-value vectors to represent disambiguation instances. This has made it difficult to exploit deep knowledge sources in the generation of the disambiguation models, that is, knowledge that goes beyond simple features extracted directly from the corpus, like bags-of-words and collocations, or provided by shallow natural language tools like part-of-speech taggers.

In this paper we present a novel approach for WSD that follows a hybrid strategy, i.e. combines knowledge and corpus-based evidence, and employs a first-order formalism to allow the representation of deep knowledge about disambiguation examples together with a powerful modeling technique to induce theories based on the examples and background knowledge. This is achieved using Inductive Logic Programming (ILP) (Muggleton, 1991), which has not yet been applied to WSD.

Our hypothesis is that by using a very expressive representation formalism, a range of (shallow and deep) knowledge sources and ILP as learning technique, it is possible to generate models that, when compared to models produced by machine learning algorithms conventionally applied to

Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 41–48,
Prague, Czech Republic, June 2007.

WSD, are both more accurate for fine-grained distinctions, and "interesting", from a knowledge acquisition point of view (i.e., convey potentially new knowledge that can be easily interpreted by humans).

WSD systems have generally been more successful in the disambiguation of nouns than other grammatical categories (Mihalcea et al., 2004). A common approach to the disambiguation of nouns has been to consider a wide context around the ambiguous word and treat it as a bag of words or limited set of collocates. However, disambiguation of verbs generally benefits from more specific knowledge sources, such as the verb's relation to other items in the sentence (for example, by analysing the semantic type of its subject and object). Consequently, we believe that the disambiguation of verbs is task to which ILP is particularly well-suited. Therefore, this paper focuses on the disambiguation of verbs, which is an interesting task since much of the previous work on WSD has concentrated on the disambiguation of nouns.

WSD is usually approached as an independent task, however, it has been argued that different applications may have specific requirements (Resnik and Yarowsky, 1997). For example, in machine translation, WSD, or *translation disambiguation*, is responsible for identifying the correct *translation* for an ambiguous source word. There is not always a direct relation between the possible senses for a word in a (monolingual) lexicon and its translations to a particular language, so this represents a different task to WSD against a (monolingual) lexicon (Hutchins and Somers, 1992). Although it has been argued that WSD does not yield better translation quality than a machine translation system alone, it has been recently shown that a WSD module that is developed following specific multilingual requirements can significantly improve the performance of a machine translation system (Carpuat et al., 2006).

This paper focuses on the application of our approach to the translation of verbs in English to Portuguese translation, specifically for a set of 10 mainly light and highly ambiguous verbs. We also experiment with a monolingual task by using the verbs from Senseval-3 lexical sample task. We explore knowledge from 12 syntactic, semantic and pragmatic sources. In principle, the proposed approach could also be applied to any lexical disambiguation task by customizing the sense repository and knowledge sources.

In the remainder of this paper we first present related approaches to WSD and discuss their limitations (Section 2). We then describe some basic concepts on ILP and our application of this technique to WSD (Section 3). Finally, we described our experiments and their results (Section 4).

2 Related Work

WSD approaches can be classified as (a) knowledge-based approaches, which make use of linguistic knowledge, manually coded or extracted from lexical resources (Agirre and Rigau, 1996; Lesk 1986); (b) corpus-based approaches, which make use of shallow knowledge automatically acquired from corpus and statistical or machine learning algorithms to induce disambiguation models (Yarowsky, 1995; Schütze 1998); and (c) hybrid approaches, which mix characteristics from the two other approaches to automatically acquire disambiguation models from corpus supported by linguistic knowledge (Ng and Lee 1996; Stevenson and Wilks, 2001).

Hybrid approaches can combine advantages from both strategies, potentially yielding accurate and comprehensive systems, particularly when deep knowledge is explored. Linguistic knowledge is available in electronic resources suitable for practical use, such as WordNet (Fellbaum, 1998), dictionaries and parsers. However, the use of this information has been hampered by the limitations of the modeling techniques that have been explored so far: using deep sources of domain knowledge is beyond the capabilities of such techniques, which are in general based on attribute-value vector representations.

Attribute-value vectors consist of a set of attributes intended to represent properties of the examples. Each attribute has a type (its name) and a single value for a given example. Therefore, attribute-value vectors have the same expressiveness as propositional formalisms, that is, they only allow the representation of atomic propositions and constants. These are the representations used by most of the machine learning algorithms conventionally employed to WSD, for example Naïve Bayes and decision-trees. First-order logic, a more expressive formalism which is employed by ILP, allows the representation of variables and n-ary predicates, i.e., relational knowledge.

In the hybrid approaches that have been explored so far, deep knowledge, like selectional preferences, is either pre-processed into a vector representation to accommodate machine learning algorithms, or used in previous steps to filter out possible senses e.g. (Stevenson and Wilks, 2001). This may cause information to be lost and, in addition, deep knowledge sources cannot interact in the learning process. As a consequence, the models produced reflect only the shallow knowledge that is provided to the learning algorithm.

Another limitation of attribute-value vectors is the need for a unique representation for all the examples: one attribute is created for every knowledge feature and the same structure is used to characterize all the examples. This usually results in a very sparse representation of the data, given that values for certain features will not be available for many examples. The problem of data sparseness increases as more knowledge is exploited and this can cause problems for the machine learning algorithms.

A final disadvantage of attribute-value vectors is that equivalent features may have to be bounded to distinct identifiers. An example of this occurs when the syntactic relations between words in a sentence are represented by attributes for each possible relation, sentences in which there is more than one instantiation for a particular grammatical role cannot be easily represented. For example, the sentence "John and Anna gave Mary a present." contains a coordinate subject and, since each feature requires a unique identifier, two are required ($subj_1$-$verb_1$, $subj_2$-$verb_1$). These will be treated as two independent pieces of knowledge by the learning algorithm.

First-order formalisms allow a generic predicate to be created for every possible syntactic role, relating two or more elements. For example *has_subject(verb, subject)*, which could then have two instantiations: *has_subject(give, john)* and *has_subject(give, anna)*. Since each example is represented independently from the others, the data sparseness problem is minimized. Therefore, ILP seems to provide the most general-purpose framework for dealing with such data: it does not suffer from the limitations mentioned above since there are explicit provisions made for the inclusion of background knowledge of any form, and the representation language is powerful enough to capture contextual relationships.

3 A hybrid relational approach to WSD

In what follows we provide an introduction to ILP and then outline how it is applied to WSD by presenting the sample corpus and knowledge sources used in our experiments.

3.1 Inductive Logic Programming

Inductive Logic Programming (Muggleton, 1991) employs techniques from Machine Learning and Logic Programming to build first-order theories from examples and background knowledge, which are also represented by first-order clauses. It allows the efficient representation of substantial knowledge about the problem, which is used during the learning process, and produces disambiguation models that can make use of this knowledge. The general approach underlying ILP can be outlined as follows:

Given:

- a set of positive and negative examples $\boldsymbol{E} = \boldsymbol{E}^+ \cup \boldsymbol{E}^-$
- a predicate $\boldsymbol{p}$ specifying the target relation to be learned
- knowledge $\boldsymbol{K}$ of the domain, described according to a language $\boldsymbol{L_k}$, which specifies which predicates $\boldsymbol{q_i}$ can be part of the definition of $\boldsymbol{p}$.

The goal is: to induce a hypothesis (or theory) $\boldsymbol{h}$ for $\boldsymbol{p}$, with relation to $\boldsymbol{E}$ and $\boldsymbol{K}$, which covers most of the $\boldsymbol{E}^+$, without covering the $\boldsymbol{E}^-$, i.e., $\boldsymbol{K} \wedge \boldsymbol{h} \models \boldsymbol{E}^+$ and $\boldsymbol{K} \wedge \boldsymbol{h} \not\models \boldsymbol{E}^-$.

We use the Aleph ILP system (Srinivasan, 2000), which provides a complete inference engine and can be customized in various ways. The default inference engine induces a theory iteratively using the following steps:

1. One instance is randomly selected to be generalized.
2. A more specific clause (the bottom clause) is built using inverse entailment (Muggleton, 1995), generally consisting of the representation of all the knowledge about that example.
3. A clause that is more generic than the bottom clause is searched for using a given search (e.g., best-first) and evaluation strategy (e.g., number of positive examples covered).
4. The best clause is added to the theory and the examples covered by that clause are removed from the sample set. Stop if there are more no examples in the training set, otherwise return to step 1.

3.2 Sample data

This approach was evaluated using two scenarios: (1) an English-Portuguese multilingual setting addressing 10 very frequent and problematic verbs selected in a previous study (Specia et. al., 2005); and (2) an English setting consisting of 32 verbs from Senseval-3 lexical sample task (Mihalcea et. al. 2004).

For the first scenario a corpus containing 500 sentences for each of the 10 verbs was constructed. The text was randomly selected from corpora of different domains and genres, including literary fiction, Bible, computer science dissertation abstracts, operational system user manuals, newspapers and European Parliament proceedings. This corpus was automatically annotated with the translation of the verb using a tagging system based on parallel corpus, statistical information and translation dictionaries (Specia et al., 2005), followed by a manual revision. For each verb, the sense repository was defined as the set of all the possible translations of that verb in the corpus. 80% of the corpus was randomly selected and used for training, with the remainder retained for testing. The 10 verbs, number of possible translations and the percentage of sentences for each verb which use the most frequent translation are shown in Table 1.

For the monolingual scenario, we use the sense tagged corpus and sense repositories provided for verbs in Senseval-3. There are 32 verbs with between 40 and 398 examples each. The number of senses varies between 3 and 10 and the average percentage of examples with the majority (most frequent) sense is 55%.

Verb	# Translations	Most frequent translation - %
ask	7	53
come	29	36
get	41	13
give	22	72
go	30	53
live	8	66
look	12	41
make	21	70
take	32	25
tell	8	66

Table 1. Verbs and possible senses in our corpus

Both corpora were lemmatized and part-of-speech (POS) tagged using Minipar (Lin, 1993) and Mxpost (Ratnaparkhi, 1996), respectivelly. Additionally, proper nouns identified by the tagger were replaced by a single identifier (*proper_noun*) and pronouns replaced by identifiers representing classes of pronouns (*relative_pronoun*, etc.).

3.3 Knowledge sources

We now describe the background knowledge sources used by the learning algorithm, having as an example sentence (1), in which the word "coming" is the target verb being disambiguated.

(1) "If there is such a thing as reincarnation, I would not mind **coming** back as a squirrel".

KS$_1$. Bag-of-words consisting of 5 words to the right and left of the verb (excluding stop words), represented using definitions of the form *has_bag*(*snt, word*):

has_bag(*snt*$_1$, *mind*).
has_bag(*snt*$_1$, *not*). …

KS$_2$. Frequent bigrams consisting of pairs of adjacent words in a sentence (other than the target verb) which occur more than 10 times in the corpus, represented by *has_bigram(snt, word$_1$, word$_2$)*:

has_bigram(snt$_1$, back, as).
has_bigram(snt$_1$, such, a). …

KS$_3$. Narrow context containing 5 content words to the right and left of the verb, identified using POS tags, represented by *has_narrow*(*snt, word_position, word*):

has_narrow(snt$_1$, 1st_word_left, mind).
has_narrow(snt$_1$, 1st_word_right, back). …

KS$_4$. POS tags of 5 words to the right and left of the verb, represented by *has_pos*(*snt, word_position, pos*):

has pos(snt$_1$, 1st_word_left, nn).
has pos(snt$_1$, 1st_word_right, rb). …

KS$_5$. 11 collocations of the verb: 1st preposition to the right, 1st and 2nd words to the left and right, 1st noun, 1st adjective, and 1st verb to the left and right. These are represented using definitions of the form *has_collocation*(*snt, type, collocation*):

has_collocation(*snt$_1$*, *1st_prep_right, back*).
has_collocation(*snt$_1$*, *1st_noun_left, mind*)….

KS$_6$. Subject and object of the verb obtained using Minipar and represented by *has_rel*(*snt, type, word*):

has_rel(snt$_1$, subject, i).
has_rel(snt$_1$, object, nil). …

KS$_7$. Grammatical relations not including the target verb also identified using Minipar. The relations (verb-subject, verb-object, verb-modifier, subject-modifier, and object-modifier) occurring more than 10 times in the corpus are represented by *has_related_pair(snt, word$_1$, word$_2$)*:

has_related_pair(snt$_1$, there, be). …

KS$_8$. The sense with the highest count of overlapping words in its dictionary definition and in the sentence containing the target verb (excluding stop words) (Lesk, 1986), represented by *has_overlapping*(*sentence, translation*):

has_overlapping(snt$_1$, voltar).

KS$_9$. Selectional restrictions of the verbs defined using LDOCE (Procter, 1978). WordNet is used when the restrictions imposed by the verb are not part of the description of its arguments, but can be satisfied by synonyms or hyperonyms of those arguments. A hierarchy of feature types is used to account for restrictions established by the verb that are more generic than the features describing its arguments in the sentence. This information is represented by definitions of the form *satisfy_restriction*(*snt, rest_subject, rest_object*):

satisfy_restriction(*snt$_1$,* [*human*]*, nil*).
satisfy_restriction(*snt$_1$,* [*animal, human*]*, nil*).

KS$_1$-KS$_9$ can be applied to both multilingual and monolingual disambiguation tasks. The following knowledge sources were specifically designed for multilingual applications:

KS$_{10}$. Phrasal verbs in the sentence identified using a list extracted from various dictionaries. (This information was not used in the monolingual task because phrasal constructions are not considered verb senses in Senseval data.) These are represented by definitions of the form *has_expression*(*snt, verbal_expression*):

has_expression(snt$_1$, "come back").

KS$_{11}$. Five words to the right and left of the target verb in the Portuguese translation. This could be obtained using a machine translation system that would first translate the non-ambiguous words in the sentence. In our experiments it was extracted using a parallel corpus and represented using definitions of the form *has_bag_trns*(*snt, portuguese_word*):

has_bag_trns(snt$_1$, coelho).
has_bag_trns(snt$_1$, reincarnação). …

KS$_{12}$. Narrow context consisting of 5 collocations of the verb in the Portuguese translation, which take into account the positions of the words, represented by *has_narrow_trns*(*snt, word_position, portuguese_word*):

has_narrow_trns(*snt*$_1$*, 1st_word_right, como*).
has_narrow_trns(*snt*$_1$*, 2nd_word_right, um*). …

In addition to background knowledge, the system learns from a set of examples. Since all knowledge about them is expressed as background knowledge, their representation is very simple, containing only the sentence identifier and the sense of the verb in that sentence, i.e. *sense(snt, sense)*:

sense(snt$_1$,voltar).
sense(snt$_2$,ir). …

Based on the examples, background knowledge and a series of settings specifying the predicate to be learned (i.e., the heads of the rules), the predicates that can be in the conditional part of the rules, how the arguments can be shared among different predicates and several other parameters, the inference engine produces a set of symbolic rules. Figure 1 shows examples of the rules induced for the verb "to come" in the multilingual task.

Rule_1. sense(A, voltar) :-
has_collocation(A, 1st_prep_right, back).
Rule_2. sense(A, chegar) :-
has_rel(A, subj, B), has_bigram(A, today, B),
has_bag_trans(A, hoje).
Rule_3. sense(A, chegar) :-
satisfy_restriction(A, [animal, human], [concrete]);
has_expression(A, 'come at').
Rule_4. sense(A, vir) :-
satisfy_restriction(A, [animate], nil);
(has_rel(A, subj, B),
(has_pos(A, B, nnp); has_pos(A, B, prp))).

Figure 1. Examples of rules produced for the verb "come" in the multilingual task

Models learned with ILP are symbolic and can be easily interpreted. Additionally, innovative knowledge about the problem can emerge from the rules learned by the system. Although some rules simply test shallow features such as collocates, others pose conditions on sets of knowledge sources, including relational sources, and allow non-instantiated arguments to be shared amongst them by means of variables. For example, in Figure 1, **Rule_1** states that the translation of the verb in a sentence *A* will be "voltar" (*return*) if the first preposition to the right of the verb in that sentence is "back". **Rule_2** states that the translation of the verb will be "chegar" (*arrive*) if it has a certain subject *B*, which occurs frequently with the word "today" as a bigram, and if the partially translated sentence contains the word "hoje" (the translation of "today"). **Rule_3** says that the translation of the verb will be "chegar" (*reach*) if the subject of the verb has the features "animal" or "human" and the object has the feature "concrete", or if the verb occurs in the expression "come at". **Rule_4** states that the translation of the verb will be "vir" (*move toward*) if the subject of the verb has the feature "animate" and there is no object, or if the verb has a subject *B* that is a proper noun (*nnp*) or a personal pronoun (*prp*).

4 Experiments and results

To assess the performance of the approach the model produced for each verb was tested on the corresponding set of test cases by applying the rules in a decision-list like approach, i.e., retaining the order in which they were produced and backing off to the most frequent sense in the training set to classify cases that were not covered by any of the rules. All the knowledge sources were made available to be used by the inference engine, since previous experiments showed that they are all relevant (Specia, 2006). In what follows we present the results and discuss each task.

4.1 Multilingual task

Table 2 shows the accuracies (in terms of percentage of corpus instances which were correctly disambiguated) obtained by the Aleph models. Results are compared against the accuracy that would be obtained by using the most frequent translation in the training set to classify all the examples of the test set (in the column labeled "Majority sense"). For comparison, we ran experiments with three learning algorithms frequently used for WSD, which rely on knowledge represented as attribute-value vectors: C4.5 (decision-trees), Naive Bayes and Support Vector Machine (SVM)[1]. In order to represent all knowledge sources in attribute-value vectors, KS_2, KS_7, KS_9 and KS_{10} had to be pre-processed to be transformed into binary attributes. For example, in the case of selectional restrictions (KS_9), one attribute was created for each possible sense of the verb and a true/false value was assigned to it depending on whether the arguments of the verb satisfied any restrictions referring to that sense. Results for each of these algorithms are also shown in Table 2.

As we can see in Table 2, the accuracy of the ILP approach is considerably better than the most frequent sense baseline and also outperforms the other learning algorithms. This improvement is statistically significant (paired t-test; $p < 0.05$). As expected, accuracy is generally higher for verbs with fewer possible translations.

The models produced by Aleph for all the verbs are reasonably compact, containing 50 to 96 rules. In those models the various knowledge sources appear in different rules and all are used. This demonstrates that they are all useful for the disambiguation of verbs.

Verb	Majority sense	C4.5	Naïve Bayes	SVM	Aleph
ask	0.68	0.68	0.82	0.88	0.92
come	0.46	0.57	0.61	0.68	0.73
get	0.03	0.25	0.46	0.47	0.49
give	0.72	0.71	0.74	0.74	0.74
go	0.49	0.61	0.66	0.66	0.66
live	0.71	0.72	0.64	0.73	0.87
look	0.48	0.69	0.81	0.83	0.93
make	0.64	0.62	0.60	0.64	0.68
take	0.14	0.41	0.50	0.51	0.59
tell	0.65	0.67	0.66	0.68	0.82
Average	**0.50**	**0.59**	**0.65**	**0.68**	**0.74**

Table 2. Accuracies obtained by Aleph and other learning algorithms in the multilingual task

These results are very positive, particularly if we consider the characteristics of the multilingual scenario: (1) the verbs addressed are highly ambiguous; (2) the corpus was automatically tagged and thus distinct synonym translations were sometimes

[1] The implementations provided by Weka were used. Weka is available from http://www.cs.waikato.ac.nz/ml/weka/

used to annotate different examples (these count as different senses for the inference engine); and (3) certain translations occur very infrequently (just 1 or 2 examples in the whole corpus). It is likely that a less strict evaluation regime, such as one which takes account of synonym translations, would result in higher accuracies.

It is worth noticing that we experimented with a few relevant parameters for both Aleph and the other learning algorithms. Values that yielded the best average predictive accuracy in the training sets were assumed to be optimal and used to evaluate the test sets.

4.2 Monolingual task

Table 3 shows the average accuracy obtained by Aleph in the monolingual task (Senseval-3 verbs with fine-grained sense distinctions and using the evaluation system provided by Senseval). It also shows the average accuracy of the most frequent sense and accuracies reported on the same set of verbs by the best systems submitted by the sites which participated in this task. Syntalex-3 (Mohammad and Pedersen, 2004) is based on an ensemble of bagged decision trees with narrow context part-of-speech features and bigrams. CLaC1 (Lamjiri et al., 2004) uses a Naive Bayes algorithm with a dynamically adjusted context window around the target word. Finally, MC-WSD (Ciaramita and Johnson, 2004) is a multi-class averaged perceptron classifier using syntactic and narrow context features, with one component trained on the data provided by Senseval and other trained on WordNet glosses.

System	% Average accuracy
Majority sense	0.56
Syntalex-3	0.67
CLaC1	0.67
MC-WSD	0.72
Aleph	0.72

Table 3. Accuracies obtained by Aleph and other approaches in the monolingual task

As we can see in Table 3, results are very encouraging: even without being particularly customized for this monolingual task, the ILP approach significantly outperforms the majority sense baseline and performs as well as the state-of-the-art system reporting results for the same set of verbs. As with the multilingual task, the models produced contain a small number of rules (from 6, for verbs with a few examples, to 88) and all knowledge sources are used across different rules and verbs.

In general, results from both multilingual and monolingual tasks demonstrate that the hypothesis put forward in Section 1, that ILP's ability to generate expressive rules which combine and integrate a wide range of knowledge sources is beneficial for WSD systems, is correct.

5 Conclusion

We have introduced a new hybrid approach to WSD which uses ILP to combine deep and shallow knowledge sources. ILP induces expressive disambiguation models which include relations between knowledge sources. It is an interesting approach to learning which has been considered promising for several applications in natural language processing and has been explored for a few of them, namely POS-tagging, grammar acquisition and semantic parsing (Cussens et al., 1997; Mooney, 1997). This paper has demonstrated that ILP also yields good results for WSD, in particular for the disambiguation of verbs.

We plan to further evaluate our approach for other sets of words, including other parts-of-speech to allow further comparisons with other approaches. For example, Dang and Palmer (2005) also use a rich set of features with a traditional learning algorithm (maximum entropy). Currently, we are evaluating the role of the WSD models for the 10 verbs of the multilingual task in an English-Portuguese statistical machine translation system.

References

Eneko Agirre and German Rigau. 1996. Word Sense Disambiguation using Conceptual Density. *Proceedings of the 15th Conference on Computational Linguistics (COLING-96).* Copenhagen, pages 16-22.

Marine Carpuat, Yihai Shen, Xiaofeng Yu, and Dekai WU. 2006. Toward Integrating Word Sense and Entity Disambiguation into Statistical Machine Translation. *Proceedings of the Third International Workshop on Spoken Language Translation*,. Kyoto, pages 37-44.

Massimiliano Ciaramita and Mark Johnson. 2004. Multi-component Word Sense Disambiguation. *Proceedings of Senseval-3: 3rd International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, Barcelona, pages 97-100.

James Cussens, David Page, Stephen Muggleton, and Ashwin Srinivasan. 1997. Using Inductive Logic Programming for Natural Language Processing. *Workshop Notes on Empirical Learning of Natural Language Tasks*, Prague, pages 25-34.

Hoa T. Dang and Martha Palmer. 2005. The Role of Semantic Roles in Disambiguating Verb Senses. *Proceedings of the 43rd Meeting of the Association for Computational Linguistics (ACL-05)*, Ann Arbor, pages 42–49.

Christiane Fellbaum. 1998. WordNet: *An Electronic Lexical Database*. MIT Press, Massachusetts.

W. John Hutchins and Harold L. Somers. 1992. *An Introduction to Machine Translation*. Academic Press, Great Britain.

Abolfazl K. Lamjiri, Osama El Demerdash, Leila Kosseim. 2004. Simple features for statistical Word Sense Disambiguation. *Proceedings of Senseval-3: 3rd International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, Barcelona, pages 133-136.

Michael Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. *ACM SIGDOC Conference*, Toronto, pages 24-26.

Dekang Lin. 1993. Principle based parsing without overgeneration. *Proceedings of the 31st Meeting of the Association for Computational Linguistics (ACL-93),* Columbus, pages 112-120.

Rada Mihalcea, Timothy Chklovski and Adam Kilgariff. 2004. The Senseval-3 English Lexical Sample Task. *Proceedings of Senseval-3: 3rd International Workshop on the Evaluation of Systems for Semantic Analysis of Text*, Barcelona, pages 25-28.

Saif Mohammad and Ted Pedersen. 2004. Complementarity of Lexical and Simple Syntactic Features: The SyntaLex Approach to Senseval-3. *Proceedings of Senseval-3: 3rd International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, Barcelona, pages 159-162.

Raymond J. Mooney. 1997. Inductive Logic Programming for Natural Language Processing. *Proceedings of the 6th International Workshop on ILP*, LNAI 1314, Stockolm, pages 3-24.

Stephen Muggleton. 1991. Inductive Logic Programming. *New Generation Computing*, 8(4):295-318.

Stephen Muggleton. 1995. Inverse Entailment and Progol. *New Generation Computing*, 13:245-286.

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

Paul Procter (editor). 1978. *Longman Dictionary of Contemporary English*. Longman Group, Essex.

Adwait Ratnaparkhi. 1996. A Maximum Entropy Part-Of-Speech Tagger. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, New Jersey, pages 133-142.

Phillip Resnik and David Yarowsky. 1997. A Perspective on Word Sense Disambiguation Methods and their Evaluating. *Proceedings of the ACL-SIGLEX Workshop Tagging Texts with Lexical Semantics: Why, What and How*?, Washington.

Hinrich Schütze. 1998. Automatic Word Sense Discrimination. *Computational Linguistics*, 24(1): 97-123.

Lucia Specia, Maria G.V. Nunes, and Mark Stevenson. 2005. Exploiting Parallel Texts to Produce a Multilingual Sense Tagged Corpus for Word Sense Disambiguation. *Proceedings of the Conference on Recent Advances on Natural Language Processing (RANLP-2005)*, Borovets, pages 525-531.

Lucia Specia. 2006. A Hybrid Relational Approach for WSD - First Results. *Proceedings of the COLING/ACL 06 Student Research Workshop*, Sydney, pages 55-60.

Ashwin Srinivasan. 2000. The Aleph Manual. *Technical Report*. Computing Laboratory, Oxford University.

Mark Stevenson and Yorick Wilks. 2001. The Interaction of Knowledge Sources for Word Sense Disambiguation. *Computational Linguistics*, 27(3):321-349.

Yorick Wilks and Mark Stevenson. 1998. The Grammar of Sense: Using Part-of-speech Tags as a First Step in Semantic Disambiguation. *Journal of Natural Language Engineering*, 4(1):1-9

David Yarowsky. 1995. Unsupervised Word-Sense Disambiguation Rivaling Supervised Methods. *Proceedings of the 33rd Meeting of the Association for Computational Linguistics (ACL-05)*, Cambridge, MA, pages 189-196.