An integrated approach to Word Sense Disambiguation

Jordi Atserias and **Lluís Padró** and **German Rigau**

TALP Research Center
Jordi Girona Salgado, 1-3.
08036 Barcelona
{batalla,padro}@lsi.upc.edu

IXA Group
Euskal Herriko Unibertsitatea
Donostia
{rigau}@si.ehu.es

Abstract

This paper presents an extension to perform Word Sense Disambiguation of an integrated architecture designed for Semantic Parsing. In the proposed collaborative framework, both tasks are addressed simultaneously. The feasibility and robustness of the proposed architecture for Semantic Parsing have been tested against a well-defined task on Word Sense Disambiguation (the *SENSEVAL-II English Lexical Sample*) using automatically acquired models from SemCor.

1 Introduction

This paper explores the use of new robust and flexible architectures towards Natural Language Understanding (NLU). The work here presented focuses in one of the main steps in NLU, Semantic Interpretation. As a first step, our main goal is to integrate two of the tasks involved in Semantic Interpretation: Word Sense Disambiguation and Semantic Parsing.

Word Sense Disambiguation (WSD hereafter), can be defined as the process of deciding the meaning of a word in its context. Our approach uses the possible senses of a word previously defined in a sense repository. In particular, we use WordNet (Fellbaum 98), a lexical taxonomy built at Princeton University that has become *de facto* the standard sense repository in the NLP community.

The goal of *Semantic Parsing* is to identify semantic relations between words in text, resulting in structures denoting various levels of semantic interpretation. For instance, trying to identify the semantic roles of the entities, (such as *Agent* or *Patient*) (Brill & Mooney 97). In this case, the process, named *Semantic Role Labeling* (SRL), has been the goal of the *shared tasks* of the last editions of SENSEVAL[1] and CONLL[2].

In this paper we will integrate WSD in an architecture already used for Semantic Parsing (Atserias *et al.* 01), allowing both tasks to be done simultaneously. Although, this architecture allows this integration, the lack of wide coverage resources for SRL which can be related to WordNet synsets has forced us to acquire automatically those models. Although, the models acquired are based on syntactic dependencies not roles, they allow to test the flexibility and robustness of our approach against a well established WSD task.

2 Semantic Parsing & WSD

Despite the fact that *WSD* and *Semantic Parsing* are strongly correlated, traditionally, most of the systems treat both separately. Paradoxically, *WSD* can improve *Semantic Parsing*, as the different senses of a word could present different syntactic structures (specially verbs) and the other way round, Semantic Parsing can help WSD (e.g. selectional preferences could determine the right sense of the verb (Carroll & McCarthy 00)). In this paper we present a robust and flexible architecture that aims to integrate both in a collaborative way.

Our approach to *WSD* follows the same formalization for *Semantic Parsing* that of (Atserias *et al.* 01) . This formalization was based on the application of lexicalized verbal models. Those models combine syntactic information (preposition, agreement, etc) and semantic information (roles, selectional preferences, etc.) as the model shown in Table 1.

In this system *Semantic Parsing* was carried out by means of finding the model/s which are the most similar/s to the input sentence. Following this approach and connecting those models to WordNet senses, at the same time that we identify the most similar model, the correct sense of the word will be also determined. In that way, we formalize a framework where *Semantic Parsing* and *WSD* are performed simultaneously.

During a pre-processing step, the input sentence containing the word to disambiguate is syn-

[1] http://www.senseval.org/
[2] http://www.cnts.ua.ac.be/conll/

model *impersonal* for "hablar" (to talk)					
Synt.	Prep.	Rol	Semantics	Agree.	Optional.
SE	x	se	Top	no	no
PP	de, sobre	entity	Top	no	yes
PP	con	destination	Top	no	yes

Table 1: Example of LEXPIR Syntatic-Semantic model for *Semantic Parsing*

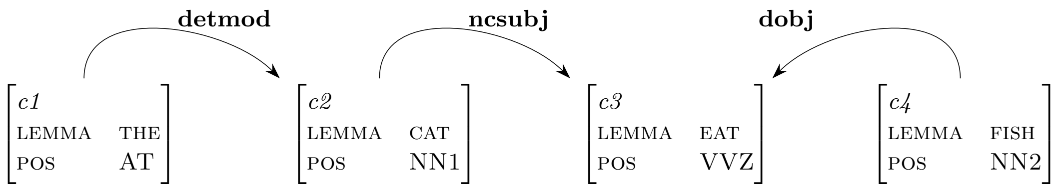

Figure 1: Syntactic Dependencies for "*The cat eats fish*"

tactically parsed and obtaining the syntactic dependencies between their elements using RASP (Carroll *et al.* 98). Figure 1 shows the dependency analysis obtained for the sentence *The cat eats fish*. Then each word is tagged with all its possible senses in WordNet. We use an specific tool for recognizing multi–word expressions (MWEs) according to WordNet (Arranz *et al.* 05) instead of the lemmatization/tokenization provided by RASP.

Once all possible senses in Wordnet are added for each word, the input is also enriched with all the information associated to each sense using the *Multilingual Central Repository* (MCR)(Atserias *et al.* 04b): the expanded (Atserias *et al.* 04a) EuroWordNet's Top Concept Ontology (Vossen 98), Suggested Upper Merged Ontology (SUMO) (Niles & Pease 01) and MultiWordNet Domains (Magnini & Cavaglia 00).

The resulting information (syntactic dependencies and semantic information) for each word is converted to a feature structure which is the input to our system. The Figure 2 shows the feature structures obtained for the two different senses of *fish*: the food sense (*fish#n#1*) and the animal sense (*fish#n#2*). Henceforth, we will use the term *object* to refer to those feature structures.

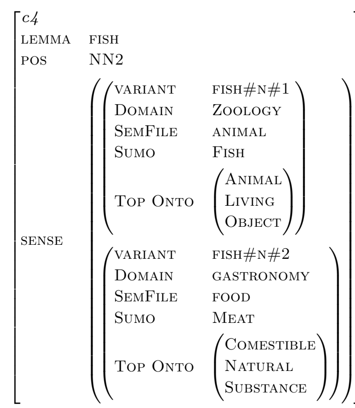

Figure 2: Object *Fish*

3 NLP as a CLP

Once the set of objects corresponding to the input sentence is obtained, it can be compared with the models. However, due to the richness of the language, robust methods to carry out this comparison are needed. Those methods should be capable to deal with semantic preferences or even to relax the syntactic structure. Thus, we formalize the problem of finding the most similar model for the input sentence as a Constraint Satisfaction Problem (CSP). CSPs have been already used in other NLP task: Part of Speech tagging (Padró 98), syntactic analysis (*Weighted Constraint Dependency Grammars* (Foth *et al.* 03)) or Machine Translation (Mikrokosmos (Beale 96)).

In most NLP tasks, and specially in *WSD*, we need to express fuzziness, possibilities, preferences, costs, that is, soft constraints, and then the problem to be solved became over–constrained. Despite the advances in the area of solving efficiently these kind of CSP with soft constraints (or preferences) (Rudova 01), to find the best solution still remains an open issue.

A natural way to model Constraint Satisfaction Problem (CSP) is by means of *Consistent Labeling Problems* (CLP)(Messeguer & Larossa 95). Consistent Labeling Problems (CLP) can be solved efficiently via Relaxation Labeling. Relaxation labeling is a generic name for a family of iterative algorithms which perform function optimization, based on local information.

4 Consistent Labeling Problems

A *Consistent Labeling Problem* (CLP) basically stands for the problem of finding the most consistent assignments of a set of variables, given a set of constraints. Formally, a Labeling Problem is defined by a set of variables V_i, a set of labels (domain) for each variable D_i, a compatibility relation over tuples. Compatibilities are real-valued functions $r_{ij} : DxD \longrightarrow \Re$ where $r_{i,j}(a, b)$ refers to the compatibility of the simultaneous assignment of a to V_i and b to V_j. In a similar way than CSP aims to find total assignments where constraints are not violated, CLP looks for labeling where variables are highly compatible with respect to compatibility functions.

The feature structures (objects) that are the input of the system are represented in the CLP by means of a set of assignments. That is, each feature of the object is represented by a variable whose domain is the set of values of that feature. The variable $c1.att$ stands for the feature *att* of the object $c1$.

However, as can be seen in figure 2 the input objects contains complex features related to the different senses. In the CLP, these objects are amalgamated. That is, the representation associated to the different senses is combined. Figure 3 shows a simplified CLP representation for the sentence *"The cat eats fish"*. The variables amalgamate all the values of the same features for different senses. For instance, the *domain* features related to the *fish* object (c3) in figure 2 are mapped into the C3.DOMAIN variable, and the possible labels of the variable C3.DOMAIN corresponds to the union of the values for the domain feature.

The main idea of this amalgamation, is that, in a similar way than *Polaroid Words* (Hirst 87), when a model is chosen the representation of the object is selected and viceversa. The consistence between the sense selected and the selection of the corresponding labels in the variable are assured by a set of constrains.

However, not only the object features have to be represented in the CLP but also the relations between those objects. Most of the problems which are naturally modeled as a CLP do not have and implicit structure. Thus, to represent a structure between objects we need to use a kind-of dependency representation.

Variable	**Values**
C1.POS*	{ NN1 }
C1.LEMMA*	{ cat }
C1.SENSE	{ cat#n#1, cat#n#2 ...}
C1.DOMAIN	{ Zoology, Factotum, Person, Transport}
C1.MODEL	{ NONE }
C1.ROLE	{ subj.m1.c2, subj.m2.c2}
C2.POS*	{ VVZ }
C2.LEMMA*	{ eat }
C2.SENSE	{ eat#v#1, eat#v#2 ...}
C3.DOMAIN	{Gastronomy, Chemistry, Factotum, Psychology, Zoology}
C2.MODEL	{ transitive }
C2.ROLE	{ TOP }
C3.POS*	{ NN1 }
C3.LEMMA*	{ fish }
C3.SENSE	{ fish#n#1, fish#n#2 }
C3.DOMAIN	{ Animal, Food}
C3.MODEL	{ NONE }
C3.ROLE	{ dobj.m1.c3}

Figure 3: CLP para *The cat eats fish*

The combination of objects by means of a model is represented using two variables, a variable named *model* which represents the model which is applied and another variable named *role* which represents the dependency between the two objects. There is one special model, named NONE, to represent the null-model (that is the no application of any model) and one special role, named TOP, to represent the null-role (that is that the object do not take part in any model).

In order to identify a role from a model label we need a triplet *(role, object, model)*. For instance, the role *dobj* of the *m1* model for the object *eat* is represented as *(dobj, c3, m1)*.

Since a CLP always assigns a label to all the variables, we will use the two null-labels defined previously: NONE for the model variables (objects which do not have/use a model, usually leaf semantic objects with no sub-constituents) and the label TOP for the role variables (objects not playing a role in the model of a higher constituent, e.g. the sentence head).

4.1 Matching Roles and Objects

In order to see whether a model can be applied or not, we should determine which combination of objects could be used to fulfill the roles of the model. First we will establish which roles an object can play in isolation, that is regardless which objects fulfill the other roles of the model by means of a similarity measure between an *object* and a *role*: $sim(obj, role)$. Once determined which pairs of role-object can be instantiated, it must be established which objects can be used together to best fulfill a model.

Some of the assignments/features which determine how much an object suits a role do not depend on the sense/model chosen and do not change in our amalgamated representation (static). For instance, in our representation the attributes *pos* or *lemma* are shared by all the senses.

Thus, the function *sim* could be split in two: a dynamic part sim_{dyn} and a static part sim_{static} which can be calculated only once (e.g. when building the CLP) and can be used to determine which objects can play a role initially in the CLP. On the other hand, the dynamic part, which depends on the sim_{dyn} could be represented as a set of constraint which takes into account the current state of the CLP (that is the weight associated to the assignment at each iteration). In the experiments carried out, the dynamic attributes are the sense, domain, SUMO and Top Onto. For simplicity we have chosen a similarity function which combines independently the similarity of each attribute:

$$sim(obj, role) = \frac{\sum_{a \in Atts} sim_{att}(role.a, obj.a)}{\#Atts}$$

Next section describes the set of constraints which ensures a) that the model are well-formed (structural) and b) the good application of both, models and roles (matching). These constraints have a weight associated standing the compatibility ($\sim$) or the incompatibility ($\nsim$).

4.2 Structural Constraints

- **Object Uniqueness**: This first axiom ensures that an object can only fulfill a role:
  $[c_x.role = a] \nsim [c_x.role = b]$
  $\forall x \in Obj\ \forall a, b \in Roles(c_x) \mid a \neq b$

- **Role Uniqueness**: A role can only be fulfilled by one object: $[c_x.role = a] \nsim [c_y.role = a]$
  $\forall x, y \in Obj\ \forall a \in Roles \mid x \neq y$
  This constraint will avoid for instance that the object *cat* and *fish* fulfill the same role simultaneously.

- **Model Uniqueness**: The models are incompatible among them:
  $[c_x.model = a] \nsim [c_x.model = b]$
  $\forall x \in Obj\ \forall a, b \in Models, x \neq y$

- **Model Inconsistence**: A role can not be fulfilled by an object if the model to which the role belongs is not being instantiated: $[c_x.model = m_b] \nsim [c_y.role = (r, m_a, x)]$
  $\forall x, y \in Obj\ (r, x, m_a) \in Roles(y)$
  $m_b, m_a \in Modelos(x) \mid m_a \neq m_b$

- **TOP Uniqueness** Only one TOP:
  $[c_x.model = TOP] \nsim [c_y.model = TOP]$
  $\forall x, y \in Obj, x \neq y$

- **TOP Existence** At least a TOP:
  $[c_x.model = TOP] \sim \nexists\ [c_y.model = TOP]$
  $\forall x, y \in Obj \mid x \neq y$

- **NONE Support** The model NONE is compatible with the inexistence of the role assignments:
  $[c_y.model = NONE] \sim \nexists\ [c_y.role = a]$
  $\forall y \in Obj$

4.2.1 Matching Constraints

- **Model Support** In order to not penalise smaller models, the support a model receives is normalized by the number of its roles.
  $[c_x.model = m] \sim [c_y.role = (r, m, x)]$
  $\forall (r, m, x) \in Roles$
  For instance, if the model *eat-V4* has three possible roles (subj, dobj, dobj2), the constraint which supports this model depending on assignment of the role *dobj2* will be $[c3_{model} = eat\text{-}V4] \sim^{\frac{1}{3}} [c3_{role} = (dobj2, eat\text{-}V4, c2)]$. The model will have also two similar constraints for the other two roles.

- **Role Support** The role support must take into account the sense which are associated to the object. Thus we need to compare each sense and the role:
  $[c_{role} = (r, m, x)] \sim^{w} [c_{sense} = s]$
  $\forall c, x \in Obj\ \forall s \in c.sense$ where w is sim_{dyn} between the senses of the object and the role. For instance, the constraint $[c3_{role} =$

(dob#2, eat-V4, c2)] $\sim^{2.45}$ *[c3$_{sense}$ = fish#n#2]* will give support to the assignment *(dob#2, eat-V, c2)* taking into account the current weight of the assignment representing the sense *fish#v#2* and their similarity in WordNet[3] with the sense/s of the role *(dob#2, eat-V4, c2)*.

4.3 Sense Constraints

The following set of constraints ensures that at the same time a model is applied, the sense associated with this model is also selected, for both the head of the model and the rest of roles. As the current formalization does not include any constraint that modifies the *domain*, SUMO or *Top Onto*, these features do not need to be represented in the CLP and can be considered as *static* in the sense that we will not have to keep their consistence.

- **Head Sense Disambiguation** This set of constraints associate the application of a model with the selection of its sense for the *head* of the model:
  $[c_{sense} = s] \sim^{100} \; Or_{i=1}^{n} \; [c_{model} = m_i]$
  $\forall s \in c.sense$ and where $m_1 ... m_n$ is the set of models of c whose sense is s
  For instance, the constraint *[c2$_{sense}$ = eat#v#3]* $\sim^{100}$ *[c2$_{model}$ = eat-V17] or [c2$_{model}$ = eat-V52] or [c2$_{model}$ = eat-V50]* would give support to the assign of the third sense of *eat* if any of the models associated to the third sense (eat-V17, eat-V52, eat-V50) is selected.

- **Role Sense Disambiguation** This set of constraints associates the sense of the role with the sense of the object which fulfills the role:
  $[c_{sense} = r.sense] \sim^{w} [c_{role} = (r, m, x)]$
  $\forall c \in Obj$ where w is $sim_{static}(obj_{r.sense}, role)$
  Where $obj_{r.sense}$ is the representation of the object corresponding to sense $r.sense$, for instance, *[c3$_{sense}$ = fish#n#2]* $\sim^{2}$ *[c3$_{role}$ = (dob2, eat-V4, c2)]* will select the second sense of *fish* if the object *c3* fulfills the role *dobj2* of model *eat-V4*. The sim_{static} will be calculated comparing the attributes associated to the object representing the second sense of *fish* and the role.

[3]For the experiments we use the level of the first common ancestor

4.4 Initial Labeling

As *relaxation labeling* is an algorithm with local convergence, one of the main issues when using this algorithm is to establish the initial labeling from where the iterative process starts. Heuristically we initialize the role and model assignments according to the static similarity function, while for the sense assignments the SemCor frequency is been used.

5 Experiments

To prove the flexibility and robustness of our approach against *WSD* we applied our system to *English Lexical Sample* of SENSEVAL-II. This tasks consists onn disambiguating the occurrences of 73 different words (noun, verbs and adjectives) in a corpus of 4,328 paragraphs. We choose this specific task because we plan to acquire the models from the examples of the training corpora and also because in SENSEVAL-III do not used WordNet senses for verbs directly.

In order to apply our system to this task, we need syntactic models which also contain semantic information about WordNet senses. Although there has been remarkable efforts to relate FrameNet and VerbNet with WordNet (Shi & Mihalcea 05), the coverage is still very low to face even a small Lexical Sample task (only 50 senses of the test are directly associated to a frame and only 640 sentences of the 4,328 could be solved correctly).

Thus, although its inherent complexity, we decide to build automatically these models from corpus. The acquisition this kind of models has many difficulties. First, the lack of disambiguated corpus, or when existing their small size which makes impossible: a) to have a wide coverage of the senses in WordNet and b) to have models of all the syntactic subcategorization patterns for a sense. Moreover, state-of-the-art WSD systems and parsers still have a significant error rate that machine Learning algorithms could not cope with.

5.1 Model Acquisition

In order to obtain models from semantically tagged corpus we used the same pre–processing than for the input (see section 2), obtaining for each sentence a set of syntactic dependencies enriched with semantic information from MCR. For each sentence, we extract for each word, the feature structures associated to its direct syntactic dependences (e.g. subj / obj / dobj). We take these set of relations as the set roles of a model for this word. For instance, taking the dependency analysis of sentence *The cat eats fish* in figure 1, two models could be acquired. One associated to *cat* (head) obtained from the dependency *The* —detmod→ *cat* and another associated to *eat*, using the dependencies *cat* —subj→ eat ←dobj— fish. Due to the big amount of models, our first approach for the experiments is to constraint the models to those having their *head* disambiguated.

The models have been obtained from two corpus with different characteristics. On one hand SemCor (Miller *et al.* 93), which is mostly disambiguated but due to his relatively small size (about 250.000 words) has a low sense coverage. On the other hand, Senseval-II training corpus for the *English Lexical Sample* task (*Senseval*) whose 8,611 examples has only one word disambiguated. Table 2 shows the figures of the models obtained from each corpus for the words to be disambiguated in the test.

Notice that even we have obtained more models from Semcor, their sense distribution and coverage is different than for the *Training*. While *Training* models are distributed among all the senses in the test corpus, the models obtained from Semcor are associated to the most frequents.

	Number of Models
Semcor	7,344
Senseval	4,438

Table 2: Models acquired

6 Results

Table 3 shows the results (**P**recision and **R**ecall) obtained for the SENSEVAL-II *English Lexical Sample* test using the models obtained from Semcor and the *Senseval* corpus respectively.

Using the models obtained for each corpus, three different experiments have been performed, varying the level of semantic information used to determine the similarity between object and role: without any semantic information (**Syntax**), using only the information from WordNet (**Synset**) and using the information associated to each sense in the MCR.

For the syntactic attributes, we constraint the object that could instantiate a role, to those whose syntactic relation and preposition is the same. This restriction is probably too strong and drastically reduces the impact of increasing the semantic information.

	Models			
	Senseval		**Semcor**	
	P	**R**	**P**	**R**
MCR	48.3	26.9	28.3	15.9
Synset	48.2	26.9	27.5	15.5
Syntax	47.9	26.8	27.0	15.2

Table 3: Results in **P**recision and **R**ecall

Although at a synset level, the results of the system seem to be modest, when using the (*coarse*) grained evaluation of SENSEVAL-II our system reach the 59% of precision (41% using Semcor). We believe that this big diference in the figures is due to the lack of applicable models of the right sense, specially when using Semcor (a close-world-assumption is implicit in our formalization and the system chosses the most similar model among all the applicable).

Checking if each test sentence has at least a role with the same syntactic relation and preposition for a model associated to the correct sense to be disambiguated, we establish an upper bound of 70% for our system using the actual models.

We consider that the results obtained prove the feasibility of our approach, although they are slightly below the state-of-the-art of *WSD*, but highly above on the current figures for *Semantic Parsing*. Moreover, we should take into account than we have made no tuning (neither on the attributes nor on the similarity functions) and that the models used where obtained automatically.

7 Discussion

The automatically obtained models suffer several limitations and do not always allow to build an adequate semantic representation. For instance for a piece of sentence like ... *clean dental surface* ... with a the dependency analysis (*dental* —mod→ *surface* —dobj→ *clean*), the system will build a representation for *dental* —mod→ *surface* which is basically associated to the semantic of his head, *surface*. As a consequence the verb *clean* is wrongly disambiguated, as the models associated to *clean#v#3* (to clean a house) are the ones more related to clean a *surface*. The fundamental piece of information that a *dental surface* is also a *body_part* is not captured by our automatically obtained models, while more simple WSD systems, such as using a bag of words, are able to capture and use that relation.

On the other hand, the current prototype makes a shallow integration of the syntactic and semantic level, so the system is sensitive to errors in the syntactic analysis being not able to disambiguate a word if a dependency analysis is not obtained.

Regarding the models acquired for Semcor, although fully disambiguated, they do not provide enough coverage. This sparseness makes more difficult to cope with inconsistencies or errors from the corpus.

The disambiguation capability of the system also depends greatly on the information available to discriminate the senses. Thus, it could be difficult be able to distinguish between senses whose MCR representation is almost the same (e.g. the five senses of *child*)).

8 Conclusions & Future Work

We have shown that it is possible to develop a more robust and flexible architecture for SEMANTIC PARSING using CSP techniques and that it can be solved efficiently using well-known optimization algorithms (such as relaxation labeling algorithms). Moreover, this formalization can be extended to other models that combine syntactic and semantic information (e.g. FrameNet).

In this paper we have presented an architecture able to integrate *Semantic Parsing* and *WSD*, where both tasks could collaborate. The system has been tested in a *WSD* task (SENSEVAL-II English Lexical Sample) using automatically acquired models.

Future lines of research include, first to extend the level of integration of *Semantic Parsing* and *WSD* using richer semantic models, and second to improve the system itself (e.g. tuning the similarity functions, propagating semantic information, etc.).

References

(Arranz *et al.* 05) V. Arranz, J. Atserias, and M. Castillo. Multiword expressions and word sense disambiguation. In Alexander Gelbukh, editor, *CICLING'05*, volume LNCS 3406, 2005.

(Atserias *et al.* 01) J. Atserias, L. Padró, and G. Rigau. Integrating multiple knowledge sources for robust semantic parsing. In *Proceedings of the International Conference, Recent Advances on Natural Language Processing RANLP'01*, Bulgaria, 2001.

(Atserias *et al.* 04a) J. Atserias, S. Climent, and G. Rigau. Towards the meaning top ontology: Sources of ontological meaning. In *4rd International Conference on Language Resources and Evaluations (LREC)*, 2004.

(Atserias *et al.* 04b) Jordi Atserias, Luis Villarejo, German Rigau, Eneko Agirre, John Carroll, Bernardo Magnini, and Piek Vossen. The MEANING multilingual central repository. In *Proceedings of the Second International Global WordNet Conference (GWC'04)*, Brno, Czech Republic, January 2004.

(Beale 96) Stephen Beale. *Hunther-Gatheter: Applying Constraint Satisfactioon, Branch-and-Bound and Solution Synthesis to computational Semantics.* Unpublished PhD thesis, Computer Research Laboratory, New Mexico State University, 1996.

(Brill & Mooney 97) Eric Brill and Raymond J. Mooney. An Overview of Empirical Natural Language Processing. *Artificial Intelligence Magazine*, 18(14):13–24, 1997. Special Issue on Empirical Natural Language Processing.

(Carroll & McCarthy 00) J. Carroll and D. McCarthy. Word sense disambiguation using automatically acquired verbal preferences. *Computers and the Humanities. Senseval*, 34(1-2), 2000.

(Carroll *et al.* 98) J. Carroll, G. Minnen, and E. Briscoe. Can subcategorisation probabilities help a statistical parser? In *Proceedings of the Sixth ACL/SIGDAT Workshop on Very Large Corpora*, pages 118–126, 1998.

(Fellbaum 98) C. Fellbaum, editor. *WordNet. An Electronic Lexical Database.* The MIT Press, 1998.

(Foth *et al.* 03) Kilian Foth, Wolfgang Menzel, and Ingo Schröder. Robust parsing with weighted constraints. to appear in Natural Language Engineering, 2003.

(Hirst 87) Graeme Hirst. *Semantic Interpretation and the Resolution of the ambiguity.* Studies in Natural Language Processing. Cambridge University Press, 1987.

(Magnini & Cavaglia 00) B. Magnini and G. Cavaglia. Integrating subject field codes into wordnet. In *Proceedings of the Second Internatgional Conference on Language Resources and Evaluation LREC'2000*, Athens. Greece, 2000.

(Messeguer & Larossa 95) Pedro Messeguer and Javier Larossa. Constraint satisfaction as global optimization. In *IJCAI*, 1995.

(Miller *et al.* 93) G. Miller, C. Leacock, R. Tengi, and R. Bunker. A Semantic Concordance. In *Proceedings of the ARPA Workshop on Human Language Technology*, 1993.

(Niles & Pease 01) I. Niles and A. Pease. Towards a standard upper ontology. In *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems*, pages 17–19. Chris Welty and Barry Smith, eds, 2001.

(Padró 98) Lluís Padró. *A Hybrid Environment for Syntax-Semantic Tagging.* Unpublished PhD thesis, Departament de Llenguatges i Sistemes Informàtics. Universitat Politècnica de Catalunya. Barcelona, 1998.

(Rudova 01) Hana Rudova. *Constraint Satisfaction with Preferences.* Unpublished PhD thesis, Masaryk University, 2001.

(Shi & Mihalcea 05) Lei Shi and Rada Mihalcea. Putting pieces together: Combining framenet, verbnet and wordnet for robust semantic parsing. In *CICLING'05*, Mexico, 2005.

(Vossen 98) P. Vossen, editor. *EuroWordNet: A Multilingual Database with Lexical Semantic Networks* . Kluwer Academic Publishers , 1998.