

Euskal Herriko Unibertsitatea/Universidad del País Vasco



Lengoaia eta Sistema Informatikoak Saila
Departamento de Lenguajes y Sistemas Informáticos

Towards Robustness in Natural Language Understanding

Jordi Atserias i Batalla

Informatikan Doktore titulua eskuratzeko aurkezturiko

Tesia

Donostia, 2006ko ekaina

Euskal Herriko Unibertsitatea/Universidad del País Vasco



Lengoaia eta Sistema Informatikoak Saila
Departamento de Lenguajes y Sistemas Informáticos

Towards Robustness in Natural Language Understanding

Jordi Atserias i Batalla, Lluís Padró
Cirera and German Rigau Claramunt
zuzendaritzapean egindako tesiaren
txostena, Euskal Herriko Unibersi-
tatean Informatikan Doktore titulua
eskuratzeko aurkeztua

Donostia, 2006ko ekaina

To my family

Abstract

Most of the different tasks included in Natural Language Processing (NLP) (such as, Word Sense Disambiguation, Information Retrieval, Information Extraction, Question Answering, Information Filtering, Natural Language Interfaces, Story Understanding or Machine Translation) apply different levels of Natural Language Understanding (NLU).

This thesis explores a new integrated architecture for robust NLU, exploiting constraint-based optimization techniques. The goal of this work is to find robust and flexible architectures able to deal with the complexity of advanced NLP.

In particular, we present a novel architecture (PARDON), orthogonal to the traditional NLP task decomposition, which applies any kind of knowledge (syntactic, semantic, linguistic, statistical) at the earliest opportunity while retaining an independent representation of the different kinds of knowledge.

The different architectures proposed for NLU can be classified based on two main dimensions, namely, the level of integration of their processes and the level of integration of their data.

An easier modularization aimed at focusing on a particular NLP task and competitions (e.g. MUC, TREC, etc) have lead most of the researchers to adopt a pipelined or stratified architecture. However, this architecture shows several drawbacks which has made us consider the use of integrated and interactive approaches. In order to implement such approaches, we will also introduce the Consistent Labeling Problems (CLPs), a specific case of Constraint Satisfaction Problems that can be solved efficiently by a set of iterative algorithms (e.g. relaxation labeling).

Constraints allow us to integrate both processes and knowledge in the same framework. On the one hand, many forms of ambiguity can be represented in a compact and elegant manner, and processed efficiently by means of constraints. On the other hand, many NLP processes (e.g., many WSD techniques) could also be represented as constraints.

Inside the PARDON architecture, an object uses its models to combine itself with other objects. During this combination, some of its attribute values are determined (in a similar way to Hearst's Polaroid Words [Hirst, 1987]). Roughly speaking, PARDON combines objects from one level in order to build the objects corresponding to the next level of the task under consideration. This combination is carried out by using lexicalized models. That is, these models must be anchored-in/triggered-by a first-level object. PARDON represents the relationships between objects in a

dependency-like style, with models and roles. In order to avoid the combinatorial explosion of possible object combinations, this framework is formalized as a Consistent Labeling Problem (CLP). Thus, it can be solved using optimization methods (e.g. the relaxation labeling algorithm) to find the most consistent solution.

PARDON aims to give a general framework, that is multilingual and open domain, in which different NLP tasks can be easily formalized. These different tasks can be tested separately or carried out simultaneously following an integrated approach.

Pursuing this goal, we have also integrated several resources in a multilingual knowledge base, named Multilingual Central Repository (MCR). MCR has been built around WordNet, using the EuroWordNet architecture. This multilingual repository integrates different resources, ontologies (SUMO, Top Concept Ontology), thematic classifications (Domains), local wordnets of five different languages, and so on.

The new architecture proposed by PARDON has been successfully applied to two different NLU tasks involved in Semantic Interpretation, namely Semantic Role Labeling (SRL) and Word Sense Disambiguation (WSD).

Usually, Word Sense Disambiguation and Semantic Role Labeling are considered separately although they are strongly related. WSD can improve results in SRL (as different senses have different syntactic behaviours, specially verbs) and vice-versa (e.g. using verbal preferences for WSD).

Acknowledgements / Agraïments / Esker onak

I have to admit that this thesis would have never been finished without Maribi Arranz, Eli Comelles, Gerard Escudero and Luis Talavera bugging me constantly until I had written down the full document.

Firstly, I want to thank the whole people in the IXA group for adopting me and my thesis but also for showing me their friendship and healthy way of life during those conferences all around the world, from Areso to Mexico City, specially to Aitor, Aitziber, Arantza, David, Eneko, Iñaki, Izaskun, Nerea, Maite, Xabier (*mila esker*).

No thesis is an island, and for that reason I would also like to thank many people who, indirectly, have collaborated in this work, most of them by being involved in the research projects mentioned below: Salvador Climent, Bernardo Magnini, Luisa Bentivogli, Christian Girardi, Emanuele Pianta and Piek Vossen. I am highly indebted to John Carroll and Rob Koelling for their patience and help in showing me how to use RASP; to Diana McCarthy for her comments and support; to Maribi Arranz, Irene Castellón, Montse Civit, Eli Comelles and Toni Martí for giving me the linguistic background that, no doubt, I needed.

I also want to thank those people whose friendship has made this long way much more worth it: Mauro Castillo, Montse Cuadros, Jesús Giménez, Muntsa Padró, Francis Real, Luis Villarejo, the people sharing office 305, those who also shared the UPC undergrounds with me (office s107) and, mainly, those who have shared my early-morning coffees. I would like to mention very specially both Horacio Rodríguez and Núria Castell, not only for their help all throughout all these years but also for their huge human value.

I am also grateful to my two Ph.D. advisors, Lluís Padró and German Rigau, for all their help and time dedicated to me.

This research has been partially funded by the European Commission through several projects (EuroWordNet (LE4003) and MEANING (IST-2001-34460)), by the Spanish Research Department through project ITEM (TIC96-1243-c03-03), by scholarships from Spanish Ministry of Education and Culture, and finally, some time by myself.

Table of Contents

Acronyms and abbreviations	xiii
I Introduction	1
I.1 Towards Natural Language Understanding	1
I.1.1 Semantic Interpretation	2
I.2 NLU Open Issues and Current Challenges	3
I.2.1 Knowledge and Processes interaction	5
I.2.2 The Need of Ontologies and Reasoning	6
I.2.3 A Multilingual World	6
I.2.4 New Architectures for NLU	7
I.3 Goals of this Thesis	7
I.4 Contributions	8
I.5 Overview	8
II Knowledge, Data and Architectures for NLU	11
II.1 Introduction	11
II.2 Towards a General NLU Architecture	12
II.3 NLP Processes	14
II.3.1 NLP Process Integration	19
II.4 Knowledge for NLP	21
II.4.1 Lexical Acquisition	23
II.4.2 NLP Knowledge Integration	26
II.5 Integrating NLP Processes and Knowledge	28
III Knowledge Integration for NLU	31
III.1 MCR Overview	32
III.1.1 MCR Structure	32
III.1.2 MCR Content	35
III.2 The Uploading Process	40
III.2.1 Uploading Base Concepts	41

III.2.2	Uploading Top Concept Ontology	41
III.2.3	Evaluating the Uploading Process	41
III.3	The Integration Process	47
III.3.1	Realisation	47
III.3.2	Generalisation	56
III.3.3	Cross-Checking	56
III.4	Porting Process	59
IV	Process Integration in PARDON	61
IV.1	Introduction	61
IV.2	PARDON's Architecture	64
IV.3	Knowledge Representation in PARDON	65
IV.4	Role and Model Application	67
IV.5	Model Application Constraints	69
IV.6	Inference Engine	70
IV.7	Derivational Sequences	72
IV.7.1	Model Combination Constraints	72
IV.7.2	Amalgamating the Search Space	75
IV.8	Formalization as a CLP	82
IV.9	Conclusions	83
V	PARDON Semantic Role Labeler	85
V.1	Different Approaches to Semantic Interpretation	86
V.1.1	ABSITY	86
V.1.2	Hunter-Gatherer	87
V.1.3	Fernando Gomez's Semantic Parser	87
V.1.4	Compansion	89
V.1.5	Machine Learning approaches to SRL	89
V.2	Applying the PARDON's approach	90
V.3	Lexical Models for Semantic Role Labeling	91
V.3.1	LEXPIR	92
V.4	PARDON's Formalization for Semantic Role Labeling	94
V.4.1	Knowledge Representation	95
V.4.2	Role and Model Application	97
V.4.3	Model Application Constraints	98
V.4.4	Model Combination constraints	99
V.4.5	PP-attachment constraints	100
V.4.6	Structural Constraints	100

V.4.7	Modeling Lexical Attraction	101
V.4.8	Initial State	102
V.5	Experiments	102
V.5.1	Results	103
V.6	Discussion	104
VI	A PARDON prototype for Word Sense Disambiguation	107
VI.1	Different Approaches to WSD	107
VI.2	Applying the PARDON approach	109
VI.2.1	PARDON's input	109
VI.2.2	Lexicalized Models for SRL and WSD	110
VI.2.3	WSD Methods using PARDON's models	113
VI.3	PARDON's Formalization for WSD	116
VI.3.1	Knowledge Representation	117
VI.3.2	Attribute Representation	117
VI.3.3	Role and Model Application	118
VI.3.4	Model Application Constraints	121
VI.3.5	Structural Constraints	122
VI.3.6	Initial Labeling	123
VI.4	Experiments	123
VI.4.1	Obtaining Lexical Models for SRL and WSD	124
VI.5	Results	131
VI.5.1	Senseval-II Evaluation Issues	131
VI.5.2	Baselines and Upper bounds	132
VI.5.3	Results using the models acquired from English Lexical Sample training corpus	133
VI.6	Discussion	137
VI.6.1	What PARDON can not do	139
VII	Conclusions and Future Work	141
VII.1	Contributions	141
VII.1.1	Proposing a novel NLP Architecture	141
VII.1.2	NLU Knowledge Integration	142
VII.1.3	NLU Process Integration	142
VII.1.4	Use of Optimization Techniques in NLU	143
VII.1.5	Robust NLU	143
VII.2	Further Work	143
VII.2.1	Regarding PARDON's Architecture	143

VII.2.2	Regarding PARDON as a Semantic Parser	143
VII.2.3	Regarding PARDON as a Word Sense Disambiguator	144
A	Author’s Most Relevant Publication	165
A.1	Book Chapters	165
A.2	Journals	165
A.3	Conferences	166
A	Consistent Labeling Problems and Relaxation Labeling	169
A.1	Consistent Labeling Problems	169
A.2	Algorithms to solve CLP	170
B	Integration Example	173
C	MCR Examples	187
C.1	The “Vaso” Example	187
C.2	The “Pasta” Example	192
D	Lexicographer File - Top Concept Ontology	195
E	Senseval-II issues	197

List of Figures

I.1	Example of different semantic representations for “ <i>The cat eats fish</i> ”	3
I.2	Example of semantic roles.	3
II.1	Parsed Tree for “The cat eats fish”	16
II.2	Revision Based Architecture	19
II.3	Feedback Architecture	20
II.4	Blackboard Architecture	20
III.1	EuroWordNet architecture	33
III.2	The EuroWordNet Top-Ontology	34
III.3	Wn1.6 Extends the Wn1.5 variants	44
III.4	Wn1.6 Reduces the Wn1.5 variants	44
III.5	Wn1.6 and Wn1.5 variant set are not subsets of each other	44

III.6	Overlap Glosses	45
III.7	Multiple inheritance for <i>piece_of_leather#1</i>	49
III.8	Multiple inheritance for <i>atropine#1</i>	50
III.9	Partial view of WN1.6	53
IV.1	Semantic Representation for <i>cat</i> and <i>fish</i>	62
IV.2	Example of compositionally	63
IV.3	Variables associated with the frame-like representation of <i>cat</i>	65
IV.4	Two different CLP formalization of <i>The cat eats fish</i>	65
IV.5	A simple Context Free Grammar	66
IV.6	Representation of the possible instantiations of the rule $D, NP \implies NP$	66
IV.7	CLP representation for CFG parsing of “The cat eats fish”	67
IV.8	A complete scheme of all possible derivations	71
IV.9	Violation of the Uniqueness Object Instantiation	72
IV.10	Violation of Role Uniqueness	73
IV.11	Violation of the Model Uniqueness	74
IV.12	Role Inconsistence	74
IV.13	Consistent Partial Objects generated from “The cat eats fish”	76
IV.14	Some of the different possible solutions for the “The cat eats fish”	76
IV.15	CLP representation	79
IV.16	A Complex Model with Propagation of Attributes	80
V.1	Example of what PW can not do.	87
V.2	Example of Fernando Gomez’s Semantic Predicates	88
V.3	Chunks for “Este año en el congreso del partido se habló de las pensiones”	90
V.4	Case-role structures obtained for the sentence in Figure V.3	91
V.5	CLP associated to the objects in Figure V.3	95
VI.1	Dependencies for “ <i>The cat eats fish</i> ”	109
VI.2	Object <i>Fish</i> enriched with MCR information	110
VI.3	Model Matching	111
VI.4	play.131 example of the SENSEVAL-II English lexical sample	112
VI.5	Dependency Analysis	112
VI.6	Model Matching	114
VI.7	Role Matching	114
VI.8	Senses for the noun dog in WordNet	115
VI.9	Role Matching	115
VI.10	Model Matching	116
VI.11	CLP for <i>The cat eats fish</i>	117

VI.12	The grammatical relation hierarchy.	118
VI.13	Example of semantic similarity over the WordNet hierarchy	120
VI.14	SENSEVAL-II English Lexical Task Test Paragraph <i>play.009</i> corresponding to sense <i>play#v#4</i>	125
VI.15	Models for the words <i>play</i> , <i>relative</i> and <i>Dennis Price</i> obtained from the senseval example <i>play.009</i>	125
VI.16	Extracting models from a set of dependencies	126
VI.17	Sentence example from SemCor <i>brown1/br-k09.xml p4 s9</i>	128
VI.18	Extracting models from SemCor	128
VI.19	Models obtained for <i>play#v#3</i>	130
VI.20	Model obtained by joining all the models for <i>play#v#3</i>	130
VI.21	SENSEVAL-II English Lexical Task Results	133
VI.22	Different senses for the noun <i>child</i> in the WordNet hierarchy	139
VI.23	Sentence example from SENSEVAL-II English Lexical Sample test (<i>play.131</i>)	140
B.1	Example of WordNet relations	175
B.2	Inherited parts of <i>fish_1</i> according to <i>Wn1.6</i>	176
B.3	Roles for VerbNet Class for <i>eat-39.1</i>	179
B.4	The four frames of VerbNet Class for <i>eat-39.1</i>	180
B.5	Two LCS entries associated to the verb <i>eat</i>	181
B.6	SUMO Eating	182
B.7	SUMO Feline	183
B.8	TCO, MW Domains, SUMO and LF for the verb <i>eat</i>	184
B.9	TCO, MW Domains, SUMO and LF for the noun <i>cat</i>	185

List of Tables

III.1	Semantic File distribution in <i>WN1.6</i>	37
III.2	Mapping <i>WN1.5</i> → <i>WN1.6</i> for Princeton WordNet version	40
III.3	Mapping synsets <i>WN1.5</i> → <i>WN1.6</i> figures for SpWN	42
III.4	Quality measure for 1:1 <i>WN1.5</i> to <i>WN1.6</i> for Spanish wordnet.	46
III.5	Figures for Spanish WordNet aligned to <i>wn1.6</i>	46
III.6	<i>lenti1_1</i>	47

III.7	00660718-v process_1	48
III.8	00660718 process_1 and 00661612 stiffening_1	52
III.9	tree_1 synset	54
III.10	finger_1 synset	55
III.11	apple_1 synset	55
III.12	SUMO vs. Domain labels	56
III.13	Instances overlapping for wn1.6 ILIs	57
III.14	Hypernym chain for all senses of the noun church in WN1.6	58
IV.1	Possible CLP Assignments using the <i>match</i> function	68
V.1	Basic Model for trajectory verbs	93
V.2	Models for the verb “ <i>hablar</i> ”	94
V.3	Model for noun modifiers	98
V.4	Verbal Model identification results	103
V.5	Verbal case-role filling results	104
VI.1	Example of LEXPIR Syntactic-Semantic model for SRL	111
VI.2	Models acquired for the 73 words included in SENSEVAL-II test corpus	127
VI.3	Number of examples for <i>play</i> in the SENSEVAL-II training corpus . . .	129
VI.4	Baseline using MFS for SENSEVAL-II English Lexical task	132
VI.5	Upper Bounds using the SENSEVAL-II Training	132
VI.6	Results for SENSEVAL-II English Lexical Sample task	134
VI.7	Results in Fine P recision and R ecall	135
VI.8	Results in P recision and R ecall for each PoS	136
VI.9	Results in P recision and R ecall for MWEs	137
VI.10	Results in P recision and R ecall for Phrasal Verbs	137
B.1	The verb “eat” in WordNet1.6	174
B.2	The noun “cat” in WordNet1.6	174
B.3	The noun “fish” in WordNet1.6	174
B.4	FrameNet Frames for “Ingest”	177
B.5	Valences from Frame “Ingest”	178
C.1	Food senses for the Spanish word <i>pasta</i>	192
C.2	Food senses for the Spanish word <i>pasta</i>	193
C.3	New Selectional Preferences for Food senses of “pasta”	194
D.1	LF -TCO Equivalences	195

E.1	Non existing variants and their correct form	197
E.2	MWE variant which are not in the text	197
E.3	Senses which appears on the test corpus but not on the training I . . .	198
E.4	Senses which appears on the test corpus but not on the training II . .	199

Acronyms and abbreviations

AI: Artificial Intelligence
BB: Blackboard
BCs: Base Concepts
BNC: British National Corpus
CFG: Context Free Grammar
CLP: Consistent Labeling Problem
CSP: Constraint Satisfaction Problem
DL: Decision Lists
DRS: Discourse Representation Structure
DRT: Discourse Representation Theory
DSO: Defense Science Organization
EWn: EuroWordNet
GATE: Generic Architecture for Text Engineering
GUI: Graphical User Interface
GR: Grammatical Relations
IBL: Instance Based Learning
IE: Information Extraction
IGR: Instantiated Grammatical Relations
IR: Information Retrieval
LCS: Lexical Conceptual Structures
LF: (WordNet) Lexicographer File
MBL: Memory Based Learning
MCR: Multilingual Central Repository
MDL: Minimum Description Length
ME: Maximum Entropy
MFS: Most Frequent Sense
ML: Machine Learning
MLE: Maximum Likelihood Estimation

MT: Machine Translation
MUC: Message Understanding Conference
MWE: MultiWord Expression
NB: Naive Bayes
NE: Named Entity
NER: Named Entity Recognition
NERC: Named Entity Recognition and Classification
NLG: Natural Language Generation
NLP: Natural Language Processing
NLU: Natural Language Understanding
PoS: Part of Speech
QA: Question Answering
RASP: Robust Accurate Statistical Parsing
SCF: Subcategorization Frame
SOM: Sources of Ontological Meaning
SpWn: Spanish WordNet
SUMO: Suggested Upper Merged Ontology
SVM: Support Vector Machines
TBL: Transformation Based Learning
TCM: Tree Cut Model
TCO: Top Concept Ontology
TREC: Text Retrieval Conference
VSM: Vector Space Model
Wn: Princeton WordNet
Wn1.6: Princeton WordNet 1.6
WSD: Word Sense Disambiguation
WSJ: Wall Street Journal

CHAPTER I.

Introduction

A child of five would understand this. Send someone to fetch a child of five.

Groucho Marx

An English professor wrote the words, “Woman without her man is nothing” on the blackboard and directed his students to punctuate it correctly.

The men wrote: *“Woman, without her man, is nothing.”*

The women wrote: *“Woman: Without her, man is nothing.”*

Unknown

1.1 Towards Natural Language Understanding

There is no doubt about the complexity of any human language, and the inherent difficulty of its automatic understanding. A single comma can change the meaning of a sentence completely (e.g. “*eats shoots and leaves*” versus “*eats, shoots and leaves*”¹) or even worse, make it mean just the opposite (“*Don’t stop*” versus “*Don’t, stop*”). The aim of this work is to explore new natural language processing architectures that are as robust and flexible as possible.

Most of the different tasks included in Natural Language Processing (NLP), such as Word Sense Disambiguation (WSD), Information Retrieval (IR), Information Extraction (IE), Question Answering (QA), Information Filtering, Natural Language Interfaces, Story Understanding [Riloff, 1999] or Machine Translation (MT), apply different levels of Natural Language Understanding (NLU). For instance, in the case of Information Extraction and Question Answering, the Natural Language Understanding component plays a major role. This is due to the fact that most of the

¹The example is borrowed from the title of a U.S. bestseller about punctuation

information to be extracted can only be identified by recognising all the conceptual components and the roles these components play.

1.1.1 Semantic Interpretation

An important step in any process that implies Natural Language Understanding is Semantic Interpretation. *Semantic Interpretation* can be defined as the process of obtaining a suitable representation for the meaning of a text [Brill and Mooney, 1997]. The input of the Semantic Interpreter can vary significantly, going from raw text to full parse trees. Likewise, the output of the Semantic Interpreter can also vary considerably (logical formulae, case-frames, SQL, Text Meaning Representation), mostly influenced by the type of application.

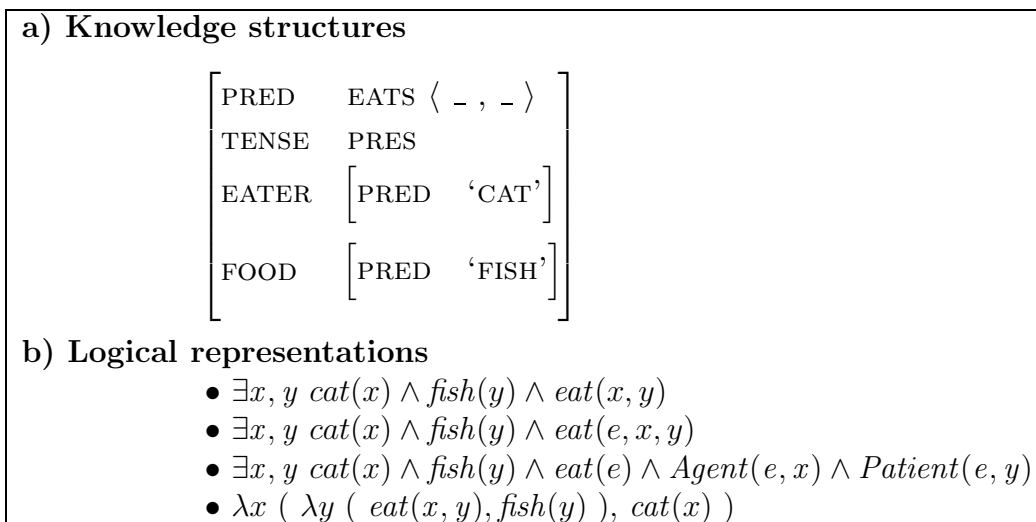
Multiple knowledge representations and formalisms for Semantic Interpretation have been developed. Mainly, they can be divided into *Knowledge structures* and *Logical representations*. *Knowledge structures* are widely used in the AI community, for example, frames, scripts or semantic nets [Woods, 1985]. *Logical representations* are unambiguous formal languages with well-defined rules of interpretation and inference. Generally, NLP tasks require high-order logics with modalities, which are either extensions of predicate logics to treat phenomena like fuzziness, believes, model operators, temporal reasoning, etc or adaptation of logics to NLP. Some examples of the former are Intentional Logics (possible worlds semantics), Episodic Logic (e.g. TRAINS [Poesio et al., 1994]), Description Logics (e.g. see [Franconi, 2002]). For an example of the later we can refer to GATE [Cunningham et al., 1996] which uses an under-specified semantic representation, named Quasi Logical Form [Alshawi, 1990], [Alshawi, 1992], [Alshawi et al., 1992]. Figure I.1 shows some examples of possible representations for the sentence *The cat eats fish*.

There also exist more complex formalisms to represent semantics beyond the sentence level, e.g. Discourse Representation Theory [Kamp and Reyle, 1993] (DRT). DRT is a powerful method for semantic representation that attempts to bridge the gap between syntax and semantics, which are probably two of the most important areas of research within Natural Language Processing. DRT is an overall theory of discourse representation, and it is associated with Discourse Representation Structures (DRSs), where formal objects realise the dynamic notion of the meaning in discourse. DRSs provide logical language-like features to DRT.

In order to obtain a representation of a context-independent meaning of a sentence, two important sub-tasks can be distinguished within Semantic Interpretation: *Word Sense Disambiguation* (WSD) and *Semantic Role Labeling* (SRL). Usually WSD and SRL are considered separately although they are strongly related. WSD can improve results in SRL (as different senses have different syntactic behaviours (specially verbs) and vice-versa (e.g. using selectional preferences to WSD [Carroll and McCarthy, 2000])).

Semantic Role Labeling (SRL) consists in the production of a case-role analysis in which the semantic roles² of the entities, such as *Starter* or *Entity*, are identified [Brill and Mooney, 1997].

²Also called thematic roles.

Figure I.1: Example of different semantic representations for “*The cat eats fish*”

<i>El gato</i>	<i>come</i>	<i>pescado</i>
The cat	eats	fish
Starter		Entity

Figure I.2: Example of semantic roles.

Even for a simple sentence as the one shown in figure I.2 (*The cat eats fish*), obtaining a semantic representation is not an easy task. Different Knowledge and Processes must be applied to the sentence in order to identify the semantic roles. Generally semantic analysis is not directly approached. First, several processes are performed on the sentence, for instance: dividing the sentence into words (*/the/ /cat/ /eats/ /fish/*), lemmatizing and Part of Speech tagging each word (*/the_the_AT/ /cat_cat_NN1/ /eats_eat_VVZ/ /fish_fish_NN2/*), semantically disambiguating the content words (*/the/ /cat#n#1/ /eat#v#3/ /fish#n#2/*) or performing some level of syntactic analysis (e.g. chunking: $\{The\ cat\}_{NP} \{eats\}_{VP} \{fish\}_{NP}$).

I.2 NLU Open Issues and Current Challenges

Some of the previous processes or tasks mentioned in the previous section seem to have reached an acceptable level of performance (e.g. lemmatization, PoS tagging) for NLP applications. However, the results in Word Sense Disambiguation are still around 70%, those for dependency parsing are about 75%, and performance is slightly over 80% for correct bracketing. Furthermore, other tasks such as Multiword Expressions are still an open issue.

The difficulty of achieving NLU, and also the difficulty of evaluating the answers of a NLU system (i.e. **black box evaluation**), led the NLP community to face and evaluate these tasks independently. In the last decade, the NLP community has focused on the evaluation of much simpler and well defined tasks, WSD (SENSEVAL), parsing (PARSEVAL), IE (MUCs) and IR (TRECc and CLEFs), etc.

Works on IE³ organized by TIPSTER⁴ [Grishman, 1995; Yangarber and Grishman, 1998; Appelt et al., 1996] have shown the need for syntax-semantics interaction. The MUC conferences showed the tendency of the Information Extraction Systems to be less domain oriented [Wilks and Catizone, 1999] and also more language independent [Humphreys et al., 1998] [Kilgarriff, 1997], making Information Extraction stand closer to Natural Language Understanding⁵.

These tendencies are also present in the Pascal⁶ challenge (evaluating machine learning for IE) and the initiatives of the American Automatic Content Extraction program (ACE)⁷ whose aim is to develop extraction technology to support automatic processing of language.

In 1999, the TREC competition included a Question Answering task for the first time. Open-domain QA is a complex application that encompasses many aspects of NLP and AI, such as the use of ontologies, reasoning and inference engines. The current state of the art QA systems can provide answers only to simple questions. However, the complexity of QA systems is rapidly evolving and extending their limits, for instance, to solve questions whose answer is distributed along several documents, questions that need non trivial inferences, to become incremental or dialog guided and so on.

In the frame of the Cross-Language Evaluation Forum (CLEF)⁸ multilinguality is also addressed in IR/QA. The objective of CLEF is to develop and maintain an infrastructure for the testing and evaluation of information retrieval systems for European languages, in both monolingual and cross-lingual contexts.

In order to improve not only the performance in all these tasks but also the general understanding capabilities of current NLP systems, the NLP community has to face all the issues raised by these tasks, such as the integration of different knowledge and the interaction of the different NLP processes, the need of ontologies and reasoning capabilities or multilinguality. The work presented in this thesis focuses on the integration of different types of knowledge and the interaction of the different NLP processes, as well as addressing other issues such as the use of ontologies and multilinguality.

The following subsections will be devoted to introducing some of these current NLP issues: Knowledge and Processes interaction, the need of ontologies, reasoning capabilities and multilinguality.

³IE was greatly promoted by the Message Understanding Conferences [MUC, 1991; MUC, 1992; MUC, 1993; MUC, 1995; MUC, 1998]. See <http://www.muc.saic.com/>

⁴http://www.itl.nist.gov/iaui/894.02/related_projects/tipster

⁵Currently, other related areas (such as Story Understanding) and Question Answering have begun to adapt recent improvements from the Information Extraction field.

⁶Pattern Analysis, Statistical Modelling, and Computational Learning

⁷<http://www.nist.gov/speech/tests/ace>

⁸<http://www.clef-campaign.org>

1.2.1 Knowledge and Processes interaction

There is not a well established consensus on which must be the components or processes of a general NLU system. However, most of the NLP systems usually involve several processes, such as tokenization, lemmatization, Part of Speech (PoS) tagging, parsing or, at least, some level of syntactic analysis. At any stage, any of the processors could bring up the vital piece of information to understand the meaning of a sentence by means of its different knowledge resources. For instance, identifying the subcategorization frame or the diathesis alternation of the verb, using the semantic information associated to an event (e.g. Ingestor:Animal ingest Ingested:Food) or applying a piece of world knowledge (e.g. “predators eat animals”) could help to grasp the meaning of the whole sentence.

NLU systems need to use different types of knowledge to accomplish their task. Allen [Allen, 1995] classifies the relevant knowledge for Natural Language Understanding into seven classes: *Phonetic and phonological Knowledge* (e.g. what the pronunciation of the word *cat* is), *Morphological Knowledge* (e.g. the root of *cats* is *cat*), *Syntactic Knowledge* (e.g. subcategorization frames, NP eat NP), *Semantic Knowledge* (the meaning of a word, e.g. eat#v#1 is to take in solid food), *Pragmatic Knowledge* (i.e. the use of sentences in different situations and how their use affects interpretation), *Discourse Knowledge* (i.e. how immediately preceding sentences affect the interpretation of the next sentence: pronouns, temporal aspects, etc.), *World Knowledge* (e.g. cats are predators).

In order to successfully understand a text, different types of knowledge and probably different knowledge resources must be used together. The compatibility among different knowledge resources is crucial to obtain a sound interpretation. The integration of different knowledge sources (information fusion [Menzel, 2002]) is an open issue in many fields (ontologies, speech recognition, image recognition). Integration of already acquired knowledge has multiple difficulties: whether the knowledge components came from different data sources, the different knowledge components have been developed based on completely different paradigms or the different components have been designed to keep separate the representation of different knowledge (e.g. to improve performance, portability). All these different types of knowledge could become incomplete or even contradictory. The representation for *hominids* in different semantic classifications is a simple example of this incompatibility caused by different levels of granularity and different ontological criteria. For instance, while the Top Concept Ontology classifies them as *Human*, WordNet’s lexicographer files assigned them to *Animal* and SUMO defines its own category, *Hominids*. This incompleteness and inconsistency of the knowledge become a real issue when NLU needs to deal with ontologies and reasoning.

1.2.2 The Need of Ontologies and Reasoning

Nowadays, there is a wide consensus that all NLP systems that seek to represent and manipulate meanings of texts need an ontology and some reasoning capabilities.

Ontologies and reasoning are the touchstones to build open domain NLU systems. Up to a few years ago, the major NLU efforts had focused on local domains (IE) or centred their understanding components on pre-established types of tasks (QA).

Ontologies are becoming a crucial issue in NLP. A recent example is the so called **Ontological Semantics** [Nirenburg and Raskin, 2004], a theory of meaning in natural language which uses an ontology as the central resource for extracting and representing meaning of natural language text, reasoning about knowledge derived from text as well as generating natural language text based on representations of their meaning.

1.2.3 A Multilingual World

Ontologies are also appealing for NLU due to their language independent nature. Multilinguality is an open issue that NLP has to face. There are between 3,000 and 5,000 human languages but only about 600 of these languages have more than 100.000 native speakers. Although initially the NLP community had mainly focused on English, in the past decades there has been an increasing interest for many other languages (Basque, Catalan, French, German, Greek, Spanish, Turkish and so on).

The difficulty to handle multilinguality lies not only on the number of different languages. It lies on the variety, the different behaviour, phenomena and richness of each one of these human languages. The complexity of an NLP process could vary remarkably depending on the language. For instance, in a free word order language, such as Spanish, you can say “*El gato come pescado*” (the cat eats fish) but also, “*Pescado come el gato*” (literally *Fish eats the cat*). The syntactic structure of both sentences is almost the same, making more difficult to determine the semantic roles, that is, to identify which the **agent** (the eater) and the **patient** (the thing being eaten) are. On the other hand, for instance, the case marking used in other languages (such as in Basque “*katuak arraia jaten du*”) could ease, in some sentences the establishment of the semantic roles. Moreover, given a sentence, the complexity of ambiguity resolution (WSD) could be completely different depending on the language, for instance the word “*cat*” in Spanish has three main senses while it has only four main senses in English. Multilinguality is a big challenge but it could also encourage the NLP community to join efforts and results so as to better understand the nature of the different human languages [Rigau et al., 2002].

Many semantic resources are now becoming multilingual (e.g. WordNet has been extended to a multilingual architecture with EuroWordNet and now there are wordnets for at least 37 languages) or have their equivalent in different languages, such as, Bonnie Dorr’s LCS, SemCor (which has an Italian equivalent [Bentivogli et al., 2005]) and recent initiatives like FrameNet (which has now a parallel project for Spanish and Japanese).

1.2.4 *New Architectures for NLU*

In order to face all these new challenges (that is, knowledge integration, the use of ontologies, multilinguality, interaction of the different NLP processes, etc.), new NLP architectures must be designed which are able to integrate all these different types of knowledge and processes in a more robust and flexible manner.

Even though Allen’s classification of knowledge types is widely accepted, there is no general agreement about how and when these different types of knowledge should be used. The Integrated Processing Hypothesis [Birnbaum, 1989] states that the language processor applies any kind of knowledge at the earliest opportunity. However, most of the current NLP architectures use a sequential approach, where knowledge is used locally in a pre-established order.

1.3 *Goals of this Thesis*

We will present a novel architecture (PARDON), orthogonal to the traditional NLP task decomposition and which applies any kind of knowledge (syntactic, semantic, linguistic, statistical) at the earliest opportunity but retaining an independent representation of the different kinds of knowledge.

PARDON aims to give a general framework that is multilingual and open domain, in which different NLP tasks can be easily formalized. These different tasks can be tested separately or carried out simultaneously following an integrated approach.

We will try to be as neutral as possible in our representation of the meaning, although our formalization stays closer to Frames used in AI or to Conceptual Graphs [Sowa, 1976]. In a similar way to that of the Object Oriented paradigm in software development, the knowledge representation of the different levels/stages inside PARDON are *objects*. Those objects, a-kind-of case-role representation, may have associated different *models* and attributes. An object uses its models to combine itself with other objects, meanwhile during this combination process some of its attribute values are determined (in a similar way to Hearst’s *Polaroid Words* [Hirst, 1987]).

Roughly speaking, PARDON combines objects from one level in order to build the objects corresponding to the next level of the task under consideration. This combination is carried out using *lexicalized* models. That is, these models must be anchored-in/trigged-by a first-level object. PARDON represents the relationships between objects in a dependency-like style, with *models* and *roles*. In order to avoid the combinatorial explosion of possible combinations of objects, this framework is formalized as a Consistent Labeling Problem (CLP). Then, it can be solved using optimization methods (e.g. the *relaxation labeling algorithm*) to find the most consistent solution.

We have successfully applied this new architecture to two NLU tasks, a) the process of obtaining a representation of the meaning of a sentence without taking its context into consideration (i.e. *Semantic Role Labeling*) and b) the selection of the appropriate sense for a word (Word Sense Disambiguation).

1.4 Contributions

The main contributions of this thesis are, on the one hand, to present a new architecture that could be a new frame to study several NLP tasks and that uses state-of-the-art optimization techniques. This architecture addresses some NLU current issues, such as the integration of different types of knowledge and processes, the use of ontologies, multilinguality and robustness in NLU. It also presents an approach to NLU knowledge integration build around the *de facto* standard resource (i.e. WordNet). On the other hand, the two chosen NLP tasks do not only show that this architecture can be applied successfully but also that it could be a new model for a general NLU architecture.

1.5 Overview

After this introduction, **Chapter II** will overview the different architectures existing in NLP. The easier modularization and the trend to focus on a particular NLP task and competition (e.g. MUC, TREC, etc.) have led most of the researchers to adopt a pipelined or stratified architecture. However, this architecture shows several drawbacks which have made us consider the use of integrated and interactive approaches. In order to implement such approaches, we will also introduce the Consistent Labeling Problems (CLPs), an specific case of Constraint Satisfaction Problems which can be solved efficiently by a set of iterative algorithms (e.g. *relaxation labeling*).

Chapter III is devoted to Knowledge Integration and our approach to integrate several resources in a multilingual knowledge base, named Multilingual Central Repository (MCR). MCR has been built around WordNet, using the EuroWordNet architecture. This multilingual repository integrates different resources, ontologies (SUMO, Top Concept Ontology), thematic classifications (Domains), local wordnets for five different languages, etc.

Chapter IV is devoted to describing PARDON's architecture, a powerful framework that takes advantage of Constraint Satisfaction techniques. PARDON aims to explore the limits of current NLP technology. That is, the main goal of PARDON is to provide a robust architecture to semantically process unrestricted text without wrongly filtering partial solutions, or over-constraining the interaction between modules and knowledge. We use the framework of Consistent Labeling Problem (CLP) (see section II.5) to integrate different NLP processes and to apply any kind of knowledge (syntactic, semantic, linguistic, statistical) at the earliest opportunity, while retaining an independent representation of the different kinds of knowledge.

Generally, Word Sense Disambiguation and Semantic Role Labeling are considered separately although they are strongly related. WSD can improve results in SRL (as different senses have different syntactic behaviours, specially verbs) and vice-versa (e.g. using verbal preferences for WSD [Carroll and McCarthy, 2000]). We decided to test PARDON's architecture on two main tasks involved in *Semantic Interpretation*, namely *Semantic Role Labeling* and *Word Sense Disambiguation*.

While **Chapter V** presents an empirical study of the performance of PARDON's architecture on a Semantic Role Labeling evaluation framework, **Chapter VI** presents a similar study regarding Word Sense Disambiguation.

Finally, **Chapter VII** draws some conclusions and highlights the future research lines that may outcome from the work presented here.

CHAPTER II.

Knowlege, Data and Architectures for NLU

“There is always an easy solution to every human problem, neat, plausible and wrong.”

H. L. Mencken

II.1 Introduction

There are several inherent difficulties to most NLU tasks:

1. The open and compositional nature of language (i.e. the difficulty of building complete repositories, or broad coverage grammars, etc).
2. The inconsistencies (either coming from the models, the knowledge or the speaker).
3. The complex interaction between different NLP levels (e.g. syntax and semantics [Grishman, 1995; Yangarber and Grishman, 1998; Appelt et al., 1996]).
4. The combinatorial explosion of possibilities produced by all these issues.

Although Semantic Interpretation includes other issues such as anaphora resolution or quantifier scope resolution, our research will focus on two important sub-tasks within Semantic Interpretation for testing our architecture: Semantic Role Labeling (SRL) and Word Sense Disambiguation (WSD). The traditional approaches consider Word Sense Disambiguation and Semantic Role Labeling separately but they are strongly related. WSD can improve results in SRL (as different senses have different syntactic behaviours, specially verbs) and vice-versa (e.g. using selectional preferences to improve WSD [Carroll and McCarthy, 2000]).

Semantic Role Labeling consists in the production of a case-role analysis in which the semantic roles –such as *agent* or *instrument*– played by each entity are identified

[Brill and Mooney, 1997]. This is a crucial task in any application that involves some level of Natural Language Understanding.

This chapter will focus on two major aspects of NLP architectures: the integration of NLP processes and the integration of NLP data, to later introduce two complex and interrelated tasks involved in NLU, SRL and WSD.

II.2 Towards a General NLU Architecture

There has been a general trend towards the development of reference architectures in NLP. While in NLU (e.g. Tipster architecture [Grishman, 1995], GATE [Cunningham et al., 1996; Cunningham et al., 2002]¹, LT-XML toolkit², etc) this trend has been quite successful in maintaining the neutrality with regard to linguistic theories, in Natural Language Generation (NLG) it has just begun (e.g. RAGS³ [Cahill et al., 2001a]) and it will probably be harder to achieve a similar success [Cahill et al., 1999a].

Neither designing an NLP system [Leidner, 2003] nor to choosing a general architecture are an easy tasks. [Callaway, 2003] claims that the architecture must be chosen according to the specific task, taking into account their **Processing aspects** and the **Representation aspects**.

That is, on the one hand, we should consider the **Processing aspects**, which are both the decomposition of the whole process into subtasks and the control structure that coordinates the various modules for efficiency requirements. On the other hand, we should also look at the **Representation aspects**, which are the relevant information for each level, the intermediate representations between modules, and the adequate formalisms to represent and manage the various kinds of knowledge involved.

Although practical applications must constrain the feasible architectures within the current technology (e.g. performance in a real-time application), there is also no doubt either about the complexity of such analysis or the impact of choosing an architecture that was too specific, when attempting a new NLP task. Reusability has also been generally neglected in NLP [Leidner, 2003].

Systems are generally easier to build and debug if they are decomposed into distinct, well-defined and easily-integrated modules with a well defined interface. A modular approach is useful both from a psychological and from an engineering point of view. Modularisation does more than facilitate the construction of a given application. It also enables the reusing of components for different applications, and makes it easier to change and update an application by restricting the scope of the modifications required in particular modules. Also from a psychological point of view, there is evidence for the existence and interaction of autonomous modules: it seems that in human language generation, the mode of operation of each module is minimally affected by the others. On the other hand, it could be argued that

¹GATE2 is compliant with TIPSTER

²<http://www.ltg.ed.ac.uk/software/xml/>

³<http://www.itri.brighton.ac.uk/projects/rags>

the speed at which humans deal with language requires different modules operating simultaneously on different pieces of the utterance.

A paradigmatic and successful example of a modular approach for processing human language is the *Generic Architecture for Text Engineering* (GATE)⁴. GATE has been in development at the University of Sheffield since 1995 and has been used in a wide variety of research and development projects, including Information Extraction (IE) for several Languages. As an architecture, GATE suggests that the elements of software systems that process natural language can be broken down into various types of components. GATE's components come in three flavours: *Language Resources* (lexicons, corpora, or ontologies), *Processing Resources* (parsers, generators, or n-gram modellers) and *VisualResources* (visualisation and editing components that participate in GUIs).

Modular architectures have allowed NLP to evolve greatly. However, each of the “standard” tasks (e.g. morphology, syntax, semantics, pragmatics, ...) still remains difficult to solve or study, even in isolation. Moreover, it remains to be seen to what extent it is realistic to have these different modules operating independently, and how they should communicate.

Beyond efficiency considerations and specific task criteria, we need to explore the limits of the current NLP technologies so that the impact of these simplifications can be evaluated in some way.

The different architectures proposed for NLU can be classified based on two main dimensions, the level of integration of their processes and the level of integration of their data. For example, following a similar paradigm, Mahesh [Mahesh, 1993] proposes a classification of the models of natural language understanding as *Sequential*, *Integrated* or *Interactive*, depending on how they relate to each other at different levels of knowledge representation and processing.

The key difference between **Interactive** and **Integrated** models is the different level of knowledge integration. The **Interactive** model retains an independent representation for each different kind of knowledge, integrating them during processing. On the other hand, The **Integrated** model⁵ assumes a full integration of knowledge even if still performing all processes simultaneously.

Although *integration* is not always contradictory with modularity (e.g. the *Reference Architecture for Generation Systems* Project (RAGS) [Cahill et al., 1999b], [Cahill et al., 2001b] introduces a framework for the representation of data in NLG systems), in general, the assumption of integration not only makes it difficult to acquire knowledge for this kind of systems but also to maintain the modularization (e.g. the NLU system SAL [Jurafsky, 1992] uses integrated knowledge stored in a monolithic knowledge base).

Before focusing on how an architecture can integrate knowledge and processes, section II.3 will present which processes are usually involved in NLP, while section II.4 will focus on the different kinds of knowledge these processes use.

⁴<http://gate.ac.uk>

⁵Also known as Flat model.

11.3 NLP Processes

As previously mentioned, there is no well-established consensus about which are the optimal components or processes of a NLU system. However, most NLP systems usually involve several processes, such as lemmatization, Part of Speech (PoS) tagging or Parsing. The difficulty of achieving NLU, and also the difficulty of evaluating the answers of a NLU system (i.e. **black box evaluation**), led the NLP community to focus and evaluate these tasks independently and use them as a **glass evaluation** of the whole NLU system.

At a semantic level, most of the NLU systems use a *Multilevel semantics architecture* [Lavelli and Magnini, 1991] where a sequence of processing phases are distinguished:

- **Lexical Discrimination:** The consistency check of the semantic part of the constituent performed when a new constituent is built (normally done by the parser).
- **Anaphora and Quantifier Scoping Resolution:** The identification of semantically plausible referents for linguistic expressions such as pronouns, deictic references, etc., as well as solving the scope of quantifiers.
- **Contextual Interpretation** The decisions of how to react in a given dialogic situation, considering the type of request, the context, etc. Generally, this requires knowledge about the speech acts, the dialog and the user model.

In order to build a “constituent”, several subtasks are needed. Among all these common tasks or processes inside NLU, we would like to emphasize the following: Tokenization, Part of Speech Tagging, Lemmatization, Word Sense Disambiguation, Parsing, Semantic Role Labeling and Anaphora Resolution:

- **Tokenization:** This refers to the splitting of a sentence into words (e.g. */the/ /cat/ /eats/ /fish/*). This division of a sentence in words has to face two major problems:
 - **Multiword Expressions:** Multiword expressions (MWEs) include a large range of linguistic phenomena, such as phrasal verbs (e.g. “*add up*”), nominal compounds (e.g. “*telephone box*”), and institutionalized phrases (e.g. “*salt and pepper*”), and they can be syntactically and/or semantically idiosyncratic in nature (See [Sag et al., 2001] for a survey). MWEs are used frequently in everyday language, usually to express precisely ideas and concepts that cannot be compressed into a single word. Due to their complexity and flexible nature, many NLP applications have ignored them. Although, in the past few years there has been a growing awareness of Multiword Expressions (MWEs) not only within large research projects specifically dedicated to MWEs (e.g. the Multiword

Expression Project⁶), Workshops *ACL-2004 Workshop on Multiword Expressions: Integrating Processing*, *ACL-2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*) but also in projects focused on other particular NLP tasks such as parsing (e.g. Robust Accurate Statistical Parsing, RASP⁷) and word sense disambiguation (e.g. MEANING⁸).

Whilst there has been considerable research on the extraction of MWEs [Schone and Jurafsky, 2001] or generating MWEs based on some knowledge source (cf. [Villavicencio, 2003a] and [Villavicencio, 2003b]), little work has been carried out in their identification. The traditional approach to deal with MWEs has been searching for the longest word-sequence match. An exception to the general use of the longest word-sequence match can be found for Question Answering [Litkowski, 2000] and Word-Sense Disambiguation ([Litkowski, 2001b], [Litkowski, 2001a], [Arranz et al., 2005]).

- **Named Entities:** Named Entities are phrases that contain the names of persons, organizations and locations as well as times and quantities. These tasks consist of recognising these phrases and classifying them according to a set of types (e.g. LOCATION, PERSON, ORGANISATION, MONEY, ...). This is usually known as Named Entity Recognition and Classification (NERC). Named Entity Recognition (NER) is a subtask of Information Extraction, thus different NER systems were evaluated as a part of the Sixth Message Understanding Conference in 1995 (MUC6). After 1995 NER systems have been developed for some European languages and a few Asian languages. In 2002, the shared task of the Conference on Computational Natural Language Learning (CoNLL-2002) also concerned language-independent named entity recognition.
- **Part of Speech Tagging:** This process consists in assigning a morphosyntactic label (e.g. noun, verb, adjective) to each word in a sentence (e.g. */the_AT/ /cat_NN1/ /eats_VVZ/ /fish_NN2/*). The set of possible morphosyntactic labels could vary (e.g. CLAWS tag set⁹, Penn treebank PoS tag set¹⁰).
- **Lemmatization:** This is the extraction of a canonical reference form from morphological variants, (e.g. the infinitive of a verb, as in *eat* for *eating*, or the masculine singular form for a noun, as in *cat* for *cats*). Lemmatization involves knowing not only the correct tokenization (i.e. having MWEs and NEs correctly identified) but also the correct PoS for each item.

⁶<http://mwe.stanford.edu>

⁷<http://www.informatics.susx.ac.uk/research/nlp/rasp>

⁸<http://www.lsi.upc.es/~nlp/meaning>

⁹CLAWS5 <http://www.comp.lancs.ac.uk/ucrel/claws5tags.html>

¹⁰http://www.ling.upenn.edu/courses/Fall_2003/ling001/penn.treebank_pos.html

- **Word Sense Disambiguation (WSD)**: This refers to determining automatically the sense of the content words of a sentence in context, usually according to a sense repository such as WordNet (e.g. */the/ /cat#n#1/ /eat#v#3/ /fish#n#2/*). The SENSEVAL¹¹ organization was created to evaluate the strengths and weaknesses of such algorithms with respect to different words, different varieties of language, and different languages within periodical competitions. Disambiguation algorithms usually take as starting point text that has been previously splitted into words (including NEs and MWEs) and PoS Tagged, and some of them also use some syntactic features.
- **Parsing**: This process determines the syntactic structure and relations inside a sentence. In NLP, parsing may be defined as the process of assigning structural descriptions to sequences of words. The input of most of the existing parsers consists of part-of-speech sequences. The kind of assigned structural description depends on the grammar¹² according to which the parser attempts to analyze the input.

Generally speaking, the motivation for parsing lies behind the belief that the grammatical structure contributes to meaning and that discovering the grammatical structure of a word sequence is a necessary step in determining the meaning of the sequence. In some parsers the construction of a meaning representation is carried out in parallel with the derivation of a structural analysis according to the grammar.

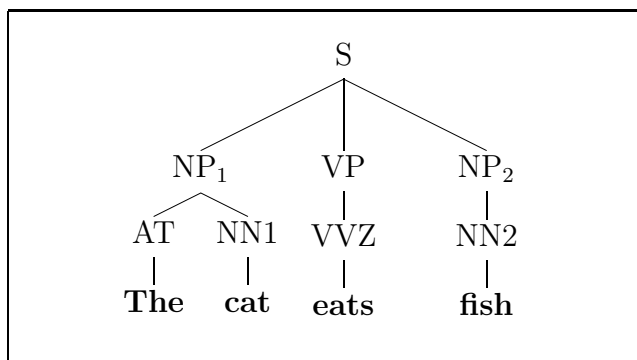


Figure II.1: Parsed Tree for “The cat eats fish”

Traditional parsers aim to recover exact and complete parses as the one shown in figure II.1. However, unrestricted text is noisy, both because of errors and because of the unavoidable incompleteness of lexicons and grammars¹³. When a full sentence parse is not possible, most of the “full” parsers, instead of rejecting the sentence as ungrammatical, attempt a parse covering the largest substring of the sentence. These global parsing considerations sometimes lead

¹¹<http://www.senseval.org>

¹²A description language plus a set of structural constraints.

¹³In restricted domains, it is difficult to integrate specialized rules (covering the sublanguage of the domain) from a broad-coverage linguistically motivated grammar.

to local errors. In many cases, a more local analyzer could perform better [Grishman, 1995].

In order to avoid these problems Abney proposes a *Chunk* approach to parsing in [Abney, 1991]. Chunking parsers build up small chunks using syntactic criteria and, then, assemble larger structures only if they are semantically licensed. Related to chunk parsing, there exists the notion of *head*¹⁴ of a chunk, which is crucial in many parsers, (e.g. Chunk Oriented Syntactic Analyzer (CHAOS) [Basili et al., 1998]). Chunk parsing is widely used in NLP to limit the analysis to the interesting context [Ciravegna and Cancedda, 1995] and for robustness. In fact, there was a general movement to use chunk parsing in the MUC systems [Grishman, 1995], on Speech Recognition Systems [Zechner and Waibel, 1998], and so on. Chunk parsing has also been extended/applied to control the chart parser strategy in order to obtain complete parsers [Ciravegna and Lavelli, 1997].

On the other side, bracketing taggers (which tag the boundaries of groups) (e.g. [Sang and Veenstra, 1999]) have evolved to tagging grammar functions [Voutilainen and Padró, 1997] and complex syntactic groups of a limited depth [Skut and Brants, 1998].

It is difficult to establish criteria for the evaluation of parsers. In 1991 the PARSEVAL system for syntactically evaluating broad-coverage English-language parsers was introduced. A new generation of parsing systems is emerging based on different underlying frameworks and covering other languages. PARSEVAL is not appropriate for many of these approaches (LREC 2002 Workshop Beyond PARSEVAL). The NLP community therefore needs to agree on a new set of parser evaluation standards [Carroll et al., 1999].

The most widely used evaluation method is based on constituent and it was proposed by the Grammar Evaluation Interest Group (PARSEVAL). However, Carroll in [Carroll et al., 1999] proposes and uses a more robust evaluation technique based on a dependency style analysis.

Beyond the parsing results, the need for a syntax-semantics interaction [Grishman, 1995; Yangarber and Grishman, 1998; Appelt et al., 1996] still remains an open issue. This is particularly so when dealing with free word order languages such as Spanish or Catalan. However, the integration of Syntax and Semantics in a Semantic Parser can vary significantly, going from *Full Integrated Systems* (where syntax and semantics interact at the same level¹⁵) to *Syntax-First Systems* (where syntax is resolved before any semantic analysis is carried out)¹⁶, or passing through *Tandem*¹⁷ Systems (where partial

¹⁴See the Shallow PARSing and Knowledge Extraction for Language Engineering (SPARKLE) page at <http://www.ilc.pi.cnr.it/sparkle.html> for an overview

¹⁵E.g., SAL [Jurafsky, 1992] (integrated) or COMPERE [Mahesh, 1995] (interactive).

¹⁶E.g., syntax driven rule-by-rule systems where each syntactic rule has a corresponding semantic interpretation rule, or those systems that are based on grammatical relations (such as Logical OBJECT and Logical SUBJECT).

¹⁷Also called interleaved systems.

syntax results are semantically validated (syntax driven)¹⁸ or where semantic attachments are proposed and then grammar rules relating the constituents are searched for to accomplish such attachments (semantic driven). An example of Tandem system is MOPTRANS/ULINK [Lytinen, 1986; Kirtner and Lytinen, 1991]).

- **Semantic Role Labeling (SRL):** A semantic role is the semantic relationship that a syntactic constituent has with a predicate. Typical semantic arguments include *Agent*, *Patient*, *Instrument*, etc. and also adjunctive arguments indicating Locative, Temporal, Manner, Cause, or other aspects. Recognizing and labeling semantic arguments is a key task for answering questions in Information Extraction, Question Answering, Summarization, and, in general, in all NLP tasks in which some kind of semantic interpretation is needed.

In two main evaluation conferences (CoNLL-2004 and SENSEVAL-III), semantic role labeling has been chosen as the shared task:

- The CoNLL-2004 shared task¹⁹ concerns the recognition of semantic roles for the English language. Given a sentence, the task consists of analyzing the propositions expressed by some target verbs in the sentence. In particular, for each target verb all the constituents in the sentence which fill a semantic role from the verb have to be recognized.
- The SENSEVAL-III task²⁰ calls for the development of SRL systems to meet the same objectives as those in Gildea and Jurafsky’s study [Gildea and Jurafsky, 2002]. The data for this task was a sample from the FrameNet hand-annotated data and the evaluation of systems followed the metrics established in Gildea and Jurafsky’s study.

- **Anaphora Resolution:** An anaphora is, roughly speaking, an abbreviated linguistic form whose full meaning can only be recovered by reference to the context. The reference is called Anaphora, and the mention of the entity to which anaphora refers is called the Antecedent (e.g. the problem of resolving what a pronoun refers to, like knowing what the referent of “*it*” is in the sentence “*the cat catches a mouse and eats it*”, see [Mitkov, 1999] for a survey). Anaphora resolution is a difficult task for a machine (and even for humans) and there is no doubt about the impact of anaphora resolution in other NLP tasks. For example refer to [Vicedo and Ferrández, 2000] for the implications of this task within Question Answering.

¹⁸e.g. ABSITY [Hirst, 1987]

¹⁹<http://www.lsi.upc.edu/~srlconll/>

²⁰<http://www.clres.com/SensSemRoles.html>

II.3.1 NLP Process Integration

This section will focus on the different ways in which processes could relate to each other in the NLU architectures. Basically, two main models of process interaction can be distinguished: *Sequential* and *Interactive*.

In **Sequential** models²¹, each level receives the output of the previous level and provides its output to the next one. Most of the current NLP architectures follow this *Sequential* model, mainly because it leads to the modularization of each task. However, it is not clear how this architecture could deal with overconstrained tasks, knowledge inconsistencies or the lack of necessary knowledge to solve a particular task.

The traditional pipeline approach works without underspecification. That is, a module can not postpone decisions and must give a single solution. Modules can not subsequently use information obtained through the operation of later modules to filter their set of solutions. In some systems, this problem is eased without breaking the sequential flow of information, by underspecifying the set of values (e.g. simplifying the set of PoS labels) or by allowing the modules' result to be a set of weighted values (instead of a unique value) (e.g. [Amtrup, 1998] or multitagging [Charniak et al., 1996]). However, in a sequential model, this set of solutions is passed on to the next module, but then the next modules are the ones in charge of reconsidering/filtering the set of results (and thus, breaking the modularity).

The interaction between syntax and semantics is a paradigmatic issue in NLU that points to a more integrated approach (e.g. WSD systems do not seem to have improved greatly between SENSEVAL-II and SENSEVAL-III), but nowadays other current challenges (e.g. WSD, SRL, MultiWord Expressions, etc.) also need new solutions.

Once we decide to allow the communication between different modules (**Interactive Model**), the integration of the different processes could be realized in different ways. [Smedt et al., 1996] differentiate three basic kinds of interactive control information flow for Natural Language Generation (NLG) (which also applies to NLP systems): **Feedback**, **Revision Based** and **Blackboard**.

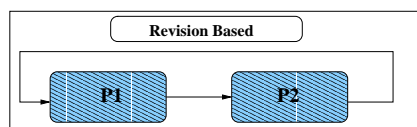


Figure II.2: Revision Based Architecture

Figure II.2 shows a **Revision Based** (or recursive Pipeline) architecture, which is basically a sequential process but the whole result can be revised/reprocessed at some point, if necessary (e.g. Hylite+ [Bontcheva and Wilks, 2001], a dynamic hypertext generation system).

²¹They are sometimes also referred to as *Stratified* models.

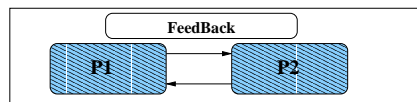


Figure II.3: Feedback Architecture

In a **Feedback** (or Interleaved) control flow, the modules work in turns, collaborating (see figure II.3). For example, in ABSITY (A Better Semantic Interpreter Than Yours) [Hirst, 1987], the syntactic and semantic modules work in tandem to build a semantic interpretation.

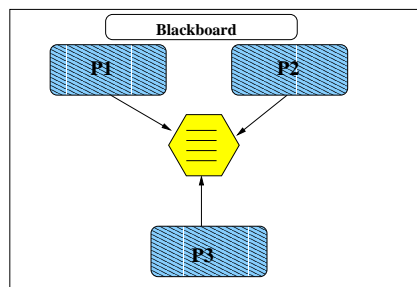


Figure II.4: Blackboard Architecture

The most general scheme of workflow is **Blackboard**, shown in figure II.4. A Blackboard System uses a shared data structure, referred to as the blackboard (BB), that contains the data of a problem to be solved and a number of different processes, referred to as modules²², that can access and modify the blackboard. Each module will post a partial solution whenever it can contribute to the overall solution of the problem. These partial solutions cause other modules to update their portions of the solution on the blackboard until eventually an answer is found.

Modules are production sets where each of them is specialized in a different type of knowledge (data-driven algorithms). Asynchronously, each module checks the BB and if it finds the appropriate input, the module processes it, and posts its results on the BB.

Meta-Control is the gateway to the BB for result posting and may rate these partial results to guide the search. It is also responsible for solution recognition. The blackboard architecture maps naturally onto multiprocessing systems or heterogeneous NLP components (e.g. Verbmobil-II architecture [Wahlster, 2000]). Different modules may be working on different parts or the same part of the problem (e.g. COMPERE [Mahesh, 1995] which integrates syntactic and semantic knowledge during processing, or the WHITEBOARD project which integrates deep and shallow parsing [Crysmann et al., 2002], and the EARSAY-II speech understanding system [Erman et al., 1980]).

However, BB also raises multiple issues [Boitet and Seligman, 1994], such as concurrence control or the communication overload, as well as the difficulties in debugging or finding an exact explanation for reaching a specific solution.

²²Sometimes, also known as *knowledge sources* (KS)

II.4 Knowledge for NLP

The previous section focused on the different ways in which processes could relate to each other in NLU architectures. This section focuses on *linguistic resources* used for NLU.

The term *linguistic resource* refers to large sets of linguistic data and descriptions in machine readable form to be used in building, improving, or evaluating natural language systems. Examples of linguistic resources are written and spoken corpora, lexical databases, grammars, and so on.

This section deals with some of these resources, mainly focusing on those lexical semantic resources related to SRL and WSD. There has been a number of initiatives to build real-world lexicons for semantic processing, most of them somehow related to Levin's Verb Classes [Levin, 1993] and WordNet [Miller et al., 1998]. Some examples are:

- **WordNet**: The Princeton WordNet [Miller et al., 1990; Fellbaum, 1998] is a lexical database which contains information about nouns, verbs, adjectives and adverbs in English and is organized around the notion of a *synset*. A synset is a set of words with the same part-of-speech that can be interchanged in a certain context. Synsets are related to each other by semantic relations, such as hyponymy (between specific and more general concepts), meronymy (between parts and wholes), cause, entailment, etc.
- **EuroWordNet (EuWn)**: EuroWordNet is a multilingual lexical database with wordnets for several European languages, which are structured as the Princeton WordNet. The EuroWordNet architecture includes the **Inter-Lingual-Index (ILI)**, a **Domain ontology** and a **Top Concept ontology** [Vossen, 1998]. The ILI consists of a list of records which interconnect word meanings in the local wordnets. During the EuroWordNet project, around 1,000 ILI-records were selected as **Base Concepts (BCs)** and consistently connected to the **Top Concept ontology**²³.
- **MultiWordNet Domains**: MultiWordNet Domains [Magnini and Cavaglia, 2000] were partially derived from the Dewey Decimal Classification²⁴. MultiWordNet Domains is a hierarchy of 165 Domain Labels associated to WordNet 1.6. Information brought by Domain Labels is complementary to what is already in WordNet. Domain Labels may include synsets from different syntactic categories: for instance, MEDICINE groups together senses from nouns, such as *doctor* and *hospital*, and from verbs such as *to operate*, and also from different WordNet subhierarchies (i.e. synsets deriving from different *unique beginners* or from different *lexicographer files*).
- **Suggested Upper Merged Ontology (SUMO)**: SUMO²⁵ [Niles and Pease,

²³<http://www.illc.uva.nl/EuroWordNet/corebcs/topont.html>

²⁴<http://www.oclc.org/dewey>

²⁵<http://ontology.teknowledge.com/>

2001] has been created as part of the IEEE Standard Upper Ontology Working Group. The goal of this Working Group is to develop a standard upper ontology that will promote data interoperability, information search and retrieval, automated inference, and natural language processing. SUMO provides definitions for general purpose terms and is the result of merging different free upper ontologies (e.g. Sowa's upper ontology, Allen's temporal axioms, Guarino's formal theory of parts and boundaries, etc.). SUMO consists of a set of concepts, relations, and axioms that formalize an upper ontology. There is a complete set of mappings from WordNet 1.6 synsets to SUMO.

- **University of Maryland's Lexical Conceptual Structures (LCS) Database**²⁶: This database [Dorr, 1993b] is based on the notion of LCS [Jackendoff, 1972]. LCS is a compositional abstraction with language independent properties. It has been used, for example, as the *interlingua* representation in several Machine Translation Systems (such as in UNITRAN [Dorr, 1993a] and MILT [Dorr, 1997]). Another related work for Spanish is that developed in **LEXPIR** [Fernández and Martí, 1996] and [Vázquez et al., 2000].
- **VerbNet**²⁷ [Kipper et al., 2000]: This is an enrichment of verb entries in WordNet that includes more specific syntactic information and verb class membership. It also draws heavily on the English Tree-Adjoining Grammar. Currently, the **PropBank** corpus²⁸ [Kingsbury and Palmer, 2002], [Kingsbury et al., 2002] and [Palmer et al., 2002] is being extended with VerbNet semantic predicates [Kipper et al., 2002]. A similar work is being carried out for nouns [Meyers et al., 2004], in the NomBank project²⁹.
- **Fernando Gomez's** work³⁰ enhanced WordNet with verbal predicates [Gomez, 1998] and designed and implemented an algorithm that uses these predicate definitions to solve verb meaning, thematic roles and prepositional attachments [Gomez, 2001].
- **FrameNet**³¹ [Baker et al., 1998]: This is based on frame semantics [Fillmore, 1968; Petruck, 1996]³². Currently, FrameNet's lexicon contains more than 4,000 lexical units (word senses) but the aim is to annotate 10,000 or more by the end of the FrameNet II project. FrameNet also provides a corpus of annotated examples (about 100,000 sentences).

These resources differ in their semantic representations (LCS vs Semantic Frames, arguments vs thematic roles) or semantic decomposition principles. However, some of these resources (e.g VerbNet, Framenet, PropBank and WordNet

²⁶<http://www.umiacs.umd.edu/~bonnie/LCS.Database.Documentation.html>

²⁷<http://www.cis.upenn.edu/verbnet/>

²⁸http://www.cis.upenn.edu/~mpalmer/project_pages/ACE.htm

²⁹<http://nlp.cs.nyu.edu/meyers/NomBank.html>

³⁰<http://www.cs.ucf.edu/~gomez>

³¹<http://www.icsi.berkeley.edu/~framenet>

³²See [F. Baker and Ruppenhofer, 2002] for a comparison between FrameNet's Frames and Levin's Verb Classes.

synsets) can be related through the **Unified index**³³. Nowadays, there is a real need for the integration of these different resources to face WSD or SRL. There are recent initiatives to build a real connection between these resources either manually (e.g. 3,094 entries integrating WordNet, FrameNet and VerbNet [Shi and Mihalcea, 2005]) or automatically (e.g. integrating FrameNet and PropBank [Giuglea and Moschitti, 2004]).

An alternative to the manual development of these resources is their automatic acquisition. This implies employing learning techniques to automatically extract linguistic knowledge from natural language corpora rather than require the system developer to manually encode the required knowledge. The following section provides an overview on these automatic methods.

II.4.1 Lexical Acquisition

Automatic lexical acquisition is an old open issue in NLP. A large battery of Machine Learning (e.g. Memory-Based Learning (MBL)), statistical (e.g. Minimum Description Length (MDL), Maximum Likelihood Estimation(MLE)/Estimation-Maximization Algorithm (EM algorithm)) or even heuristic methods have been used to obtain implicit information from structured and unstructured lexical resources.

Obtaining large explicit lexicons that are rich enough for NLP has proved difficult. Methods for automatic lexical acquisition have been developed for many topics and include collocations [Dunning, 1993; Justeson and Katz, 1995], word senses [Schütze, 1992; Lin and Pantel, 1994], prepositional phrase attachment ambiguity [Hindle and Rooth, 1993], selectional preferences [Resnik, 1993; Ribas, 1995; Li and Abe, 1998; McCarthy, 2001; Agirre and Martinez, 2001; Agirre and Martinez, 2002], subcategorization frames (SCFs) [Brent, 1991; Brent, 1993; Ushioda et al., 1993; Manning, 1993; Briscoe and Carroll, 1997; Carroll and Rooth, 1998; Gahl, 1998; Lapata, 1993; Zarkar and Zeman, 2000; Korhonen, 2002] and diathesis alternations [Lapata, 1993; Lapata, 2001; Schulte im Walde, 2000; McCarthy, 2001].

Being a multidimensional problem, predicate knowledge is one of the most complex types of information to acquire. Predicates (verbs and their corresponding nominalizations) are essential for the development of robust and accurate parsing technology that is capable of recovering predicate-argument relations and logical forms. Without predicate knowledge, resolving most structural ambiguities within a sentence is difficult, and understanding (representing at a semantic level) impossible. In that sense, the acquisition of predicate-argument associations is related to the automatic acquisition of patterns [Utsuro and Matsumoto, 1997],[Argamon et al., 1998], and IE-rule learning [Chai and Biermann, 1997], [Nobata and Sekine, 1999], [Turmo et al., 1999].

³³See the Verb Frame Search Tool at <http://www.cis.upenn.edu/~dgildea/Verbs/>

Predicate-argument knowledge has been shown to vary across corpus types (written vs. spoken), corpus genres (e.g. financial news vs. balanced text), and discourse types (single sentences vs. connected discourse) [Carroll and Rooth, 1998; Roland et al., 2000; Roland and Jurafsky, 1998]. [Roland and Jurafsky, 2002] have shown that much of this variation is caused by the effects of different corpus genres on the senses of a verb and by the effect of senses of a verb on predicate-argument associations.

Full account of predicate information requires specifying the number and type of arguments, the predicate sense, semantic representation of the particular predicate-argument component, mapping between the syntactic and semantic levels of representation, semantic selectional restrictions/preferences on participants, control of the omitted participants and possible diathesis alternations, etc. Unfortunately, all these kinds of knowledge are completely interdependent.

11.4.1.1 Traditional Approaches to Lexical Acquisition

Basically, the acquisition of predicate-argument associations has been merely syntax driven. Following a bottom-up approach, from syntax to semantics, if we identify specific associations between SCFs and predicates, we can gather information from corpus data about head lemmas which occur in argument slots of SCFs and use this information as input to selectional preference acquisition [McCarthy, 2001; Schulte im Walde, 2000]. Selectional preferences are an important part of predicate information, since they can be used to aid anaphora resolution [Ge et al., 1998], WSD [Ribas, 1995; Resnik, 1997; McCarthy et al., 2001; Grishman and Sterling, 1994] and automatic identification of diathesis alternations from corpus data [Schulte im Walde, 2000; Lapata, 1993; Stevenson and Merlo, 1998; McCarthy, 2001]. Most of these approaches are based on pre-existing syntactically annotated corpora which distinguish adjuncts from arguments. On the other hand, pure-syntactic SCFs can be acquired from corpora, or obtained using a parser and simple filtering techniques [Briscoe and Carroll, 1997] and improved with linguist cues [Lapata, 1993] or diathesis models [Korhonen, 1998]. Some semi-automatic approaches to obtain SCF (e.g. [Basili et al., 1996]) have been also proposed.

The methods used for automatic subcategorization acquisition can be divided into two groups: those based on Statistical Methods and those based on Machine Learning techniques.

11.4.1.2 Statistical Methods for Lexical Acquisition

Modeling the language in a statistical framework allows to apply different statistical techniques to obtain information from corpora with a different level of annotation (word form, lemmatized, PoS tagged, etc). In fact, most of the methods rely on the existence of syntactically analyzed corpora. Among all, Minimum Description Length (MDL) and Expectation Maximization (EM) are the most used statistical techniques.

MDL is a well-motivated and theoretically-sound principle of statistic estimation from information theory. MDL is based on the criteria that the best probability model for a given data is that which requires the least code length bits for encoding the model itself (model description length) and the given data observed through it (data description length). MDL (using a Tree-Cut-Model representation against Wordnet [Li and Abe, 1995]) has been applied to the generalization of case-frames [Li and Abe, 1995; McCarthy, 1997] and also to detect diathesis alternations [McCarthy and Korhonen, 1997; McCarthy, 2000].

EM algorithm performs Maximum Likelihood Estimation (MLE) for data in which some variables are unobserved. [Rooth et al., 1999] applies EM algorithm (representing the verbal classes as hidden variables) to classify the English verbs according to its alterations, obtaining a clustering of verbs similar to the linguistically motivated classification of Levin's [Levin, 1993]. In [Miguel et al., 1999; Miguel et al., 1998], Portuguese verbs are clustered according to its subcategorization behaviour using MLE on a Log Linear Model.

II.4.1.3 Machine Learning Methods for Lexical Acquisition

On the other hand, Machine Learning (ML) techniques are evolving rapidly. Firstly, these techniques were applied to simple tasks, e.g. MBL was used to distinguish arguments from adjuncts so that instances of the different subcategorization frames could be retrieved [Buchholz, 1998] or to learn local syntactic patterns [Argamon et al., 1998]. MBL³⁴ is a supervised Machine Learning technique for clustering which is closely related to the k nearest neighbours classifiers. The basic idea is to store all the instances in memory without making any abstraction or explicit rule. Given an example, a similarity measure is used to find the most similar instances stored.

Lately, there has been a great effort to perform more complex tasks, such as assigning semantic roles, using more sophisticated ML techniques such as Support Vector Machines (SVM). Unfortunately, most of these ML techniques are based on building black-box classifiers. This makes it basically impossible to extract any explicit knowledge that is to be extended or combined with other resources.

II.4.1.4 Future Directions

These methods are still under development and need further research before they can be successfully applied to large scale acquisition. However, [Korhonen, 2002] showed that in terms of SCF distributions, individual verbs correlate more closely with syntactically similar verbs and clearly more closely with semantically similar verbs, than with all verbs in general. Moreover, her results show that verb semantic generalisations can successfully be used to guide and structure the acquisition of SCFs from corpus data.

³⁴Also known as Instance Base or Case Base Learning.

Thus, it is possible to devise alternative acquisition schemes going top-down from semantics to syntax. If we identify specific associations between participants and predicates (selectional preferences), we can also gather information from corpus data about their particular syntactic behaviour with respect to a predicate, thus helping the acquisition of SCFs, diathesis alternations, etc. However, this new approach requires working directly at a sense level, having predicates and associations to participants semantically disambiguated.

Language diversity is not usually addressed on these works and there have been a few works related to the acquisition of subcategorization frames for languages other than English: Portuguese [Miguel et al., 1998; Miguel et al., 1999], German (B7 project)³⁵ and Spanish [Esteve Ferrer, 2004].

Furthermore, in a multilingual semantic scenario, it seems possible to devise ways to acquire some predicate-argument knowledge from a particular language and using a bottom-up approach, and then, following a top-down fashion, to acquire or validate the acquired knowledge in another language.

11.4.2 NLP Knowledge Integration

Building appropriate resources for broad-coverage processing is a hard and expensive task, involving large research groups during long periods of development. The manual creation of these resources, specially when semantics is involved, usually implies several difficulties, e.g. consistency, coverage, completeness. Moreover, sometimes it is difficult to annotate or make explicit all the information, e.g. in FrameNet, there is no explicit modelling of the syntactic realization of the frame elements. Currently, this relation is being made explicit in the annotation of the FrameNet corpus. Thus, they allow the automatic learning of these models by the application of Machine Learning/Statistical techniques (See [Gildea and Jurafsky, 2000; Gildea and Jurafsky, 2002]).

Beyond the complexity of obtaining (manually or automatically) these kind of resources, it is unrealistic to expect that a single comprehensive theory or resource can account for all the phenomena in NLP. There are many resources, sense repositories (WordNet, dictionaries), Ontologies (CYC, SUMO), verbal subcategorization and selectional preference information (LEXPIR, LCS, VerbNet, PropBank, FrameNet). All these resources vary considerably not only in the kind of knowledge they hold but also in the paradigms they are built on.

All these resources bring different pieces of information that could be the touchstone to understand the meaning of the whole sentence. Although all these resources could give partial, and sometimes contradictory information, NLP applications will need to use different NLP resources together to accomplish their particular task. Thus, some kind of integration of these resources is needed.

³⁵<http://www.sfs.nphil.uni-tuebingen.de/~abney/b7home.html>

The integration of already existing knowledge is also an open issue in many fields (information fusion [Menzel, 2002], ontologies, speech recognition, image recognition). It presents multiple difficulties given that either the knowledge components come from different data sources (e.g. enrichment (integration/fusion) of manual resources with automatically acquired knowledge), or the different knowledge components have been developed based on completely different paradigms or the different components have been designed to keep a separate representation of different types of knowledge (e.g. to improve performance, portability).

Furthermore, the current consensus that all NLP systems that need to represent and manipulate meanings require an ontology raises a hard integration issue: the use of ontologies and their integration with other traditional resources in NLP (e.g. WordNet). Ontologies, although being meaningful constructs, can not be straightforwardly used for NLP unless they are associated to linguistic units and structures.

II.4.2.1 *The Integration of Ontologies in NLU*

A straightforward way to associate an ontology to linguistic units is integrating it with the *de facto* standard for WSD, that is, WordNet (e.g. that is the case of the Suggested Upper Merged Ontology (SUMO) [Niles and Pease, 2001]). This approach has two main advantages: firstly, WordNet offers a wide coverage; secondly, wordnets and ontologies are both graphs connecting concepts, thus, it seems easier to integrate or to build a map between them than to integrate ontologies with other less structured/rich resources (such as thesauri).

However, Ontologies and WordNet are different in nature: while wordnets build concepts upon lexical units of a particular language, nodes in ontologies are claimed to be language-independent concepts.

Moreover, different ontologies are usually designed based on different theoretical grounds; e.g. while SUMO incorporates previous ontologies and insights by Sowa, Pierce, Russell and Norvig, and others, the EuroWordNet Top Concept Ontology is based on more linguistic grounds: Lyons, Vendler, Verkuyl and Pustejovsky. Therefore, although different ontologies can be comparable, it would take a great theoretical effort to merge all of them in a unique standard and comprehensive construct to be consistently associated to WordNet.

In order to show the complexity of this integration, Appendix B contains the different pieces of information that could be associated to the sentence “*The cat eats fish*” on some of the most broadly used resources in NLP, WordNet, VerbNet, FrameNet, SUMO and MultiWordNet Domains.

II.5 Integrating NLP Processes and Knowledge

While sections II.4.2 and II.3.1 have shown the issues in integrating Knowledge and Processes independently. This section is related to approaches which aim to integrate both simultaneously.

Constraints can help us to integrate both, processes and knowledge, in the same framework. On the one hand, many forms of ambiguity arising in computational linguistics can be represented compactly and elegantly, and be processed efficiently with constraints. On the other hand, many NLP process (e.g. many WSD techniques) could also be represented as constraints.

It was during the 70s that the first works on Constraint Satisfaction Problems (CSPs) appeared. The Constraint Satisfaction framework allows to express properties of a problem by means of constraints and search for a solution using specialized algorithms. CSP has been applied to solve a wide range of problems, e.g. scheduling [Agnese et al., 1995], hardware verification, graph matching [Rudolf, 1999], machine vision, etc.

Finding a solution that holds all the constraints of a CSP is NP-complete. However, finding the “best” possible solution, even if we violate some constraint³⁶ is NP-hard. Thus, it is believed that any algorithm to solve these kinds of problems will present exponential worst-case behaviour.

However, in most real applications, and NLP is not an exception but a clear example, we need to express fuzziness, possibilities, preferences, costs, that is, soft constraints, and then the problem to be solved becomes over-constrained. Despite the advances in the area of solving efficiently these kinds of CSPs with soft constraints (or preferences) [Beale, 1996], [Rudova, 2001], it still remains an open issue.

A natural way to model Constraint Satisfaction Problems is the *Consistent Labeling Problems* (CLPs) [Messeguer and Larossa, 1995]. A *Consistent Labeling Problem* basically stands as the problem of finding the most consistent value assignments for a set of variables, given a set of constraints.

Both CLP and CSP are being successfully used in several NLP tasks, such as Part of Speech tagging ([Pelillo, 1991], [Pelillo and Refice, 1994], [Padró, 1998]), for parsing using *Constraint Grammars* [Voutilainen and Padró, 1997] and *Weighted Constraint Dependency Grammars* (WCDG) ([Schröder, 2002], [Daum et al., 2002], [Foth et al., 2003], [Daum, 2004] which uses constraint optimization techniques to integrate deep and shallow parsing techniques for German). Such techniques are also being applied to more complex tasks, such as Machine Translation (Mikrokosmos [Beale, 1996]), Text planning (ICONOCLAST³⁷ [Kibble and Power, 2000]) or taxonomy mapping [Daudé, 2005].

³⁶Often known as *partial constraint satisfaction*.

³⁷<http://www.itri.brighton.ac.uk/projects/iconoclast>

Consistent Labeling Problems (CLP) can be solved via Relaxation Labeling. Relaxation labeling is a generic name for a family of iterative algorithms which perform function optimization, based on local information (see Appendix A for a more formal introduction to CLP and relaxation labeling algorithms).

The main advantages of the relaxation labeling algorithm are:

- Its highly local character (each variable can compute its new label weights given only the state at previous time step). This makes the algorithm highly parallelizable (we could have a processor to compute the new label weights for each variable, or even a processor to compute the weight for each label of each variable).
- Its expressivity: The problem is stated in terms of constraints between variable labels.
- Its flexibility: We do not have to check absolute consistency of constraints.
- Its robustness: It can give an answer to problems without an exact solution (incompatible constraints, insufficient data, ...)

The main drawbacks of the relaxation labeling algorithm are:

- Its cost. Being n the number of variables, v the average number of possible labels per variable, c the average number of constraints per label, and I the average number of iterations until convergence, the average cost is $n \times v \times c \times I$, that is, it depends linearly on n , but for a problem with many labels and constraints, or if convergence is not quickly achieved, the multiplying terms might be much bigger than n .
- Since it acts as an approximation of gradient step algorithms, it has their typical convergence problems: Found optima are local, and convergence is not guaranteed, since the chosen step might be too large for the function to optimize.

CHAPTER III.

Knowledge Integration for NLU

It was six men of Indostan
To learning much inclined,
Who went to see the Elephant
(Though all of them were blind),
That each by observation
Might satisfy his mind.

The First approached the Elephant,
And happening to fall
Against his broad and sturdy side,
At once began to bawl:
"God bless me! but the Elephant
Is very like a wall!"

The Second, feeling of the tusk
Cried, "Ho! what have we here,
So very round and smooth and sharp?
To me 'tis mighty clear
This wonder of an Elephant
Is very like a spear!"

The Third approached the animal,
And happening to take
The squirming trunk within his hands,
Thus boldly up he spake:
"I see," quoth he, "the Elephant
Is very like a snake!"

The Fourth reached out an eager hand,
And felt about the knee:
"What most this wondrous beast is like
Is mighty plain," quoth he;
"'Tis clear enough the Elephant
Is very like a tree!"

The Fifth, who chanced to touch the ear,
Said: "Even the blindest man
Can tell what this resembles most;
Deny the fact who can,
This marvel of an Elephant
Is very like a fan!"

The Sixth no sooner had begun
About the beast to grope,
Than, seizing on the swinging tail
That fell within his scope.
"I see," quoth he, "the Elephant
Is very like a rope!"

And so these men of Indostan
Disputed loud and long,
Each in his own opinion
Exceeding stiff and strong,
*Though each was partly in the right,
And all were in the wrong!*

John Godfrey Saxe's (1816-1887) version of the **The Blind Men and the Elephant**

As mentioned before, it is unrealistic to expect that a single comprehensive resource can account for all the linguistic phenomena in NLP. There are many resources available, sense repositories (WordNet, dictionaries), ontologies (EuroWordNet Top Concept Ontology, SUMO), verbal subcategorization and selectional preferences information (VerbNet, PropBank, FrameNet) which vary significantly not only in the kind of knowledge they hold but also in the paradigms they are built on. Moreover, all these resources seem to capture some piece of knowledge that the others do not and which could be crucial to solve a particular NLP task.

Furthermore, nobody guarantees that even if the integration is possible, the resulting resource will be either consistent, coherent or complete.

Any application which aims to be able to deal with natural language (e.g. understand, generate or translate) needs to have access to some representation of what words mean and what the application domain looks like.

Obviously, integrating these different kinds of conceptual resources is not an easy task, mainly due to their possible inconsistencies but also to the difficulty in finding a balance between robustness and applicability. Since our test tasks (SRL and

WSD) are related to semantics, we will build our knowledge base (named Multilingual Central Repository) around the *de facto* standard sense repository, that is, WordNet [Miller et al., 1990; Fellbaum, 1998]. In order to maintain compatibility among all the heterogeneous resources uploaded into the Multilingual Central Repository¹, it is fundamental to have a robust and advanced ontological support. We studied the mapping of the main sources of ontological meaning (e.g. SUMO, MultiWordNet Domains, EWN Top Concept Ontology (TCO), etc.) onto the Multilingual Central Repository. We also presented a preliminary study on the utility of the TCO to support advanced ontological inference. It should also be pointed out that, since we plan to work on an open domain, we did not integrate any domain-specific knowledge base into the Multilingual Central Repository.

The following section describes the Multilingual Central Repository (hereafter MCR) and the three different processes involved in the building of the MCR: the Upload process, the Integration Process and the Porting process. The **Upload process** relates the different resources and checks local consistence, the **Integration process** cross-checks and infers new knowledge, while the **Porting process** is related to the mechanism for porting knowledge across different languages. In this thesis we will focus mainly on the Upload and Integration Processes, but a detailed description of the Porting Process and its results can be found in [Atserias et al., 2004b].

III.1 MCR Overview

As we are dealing with semantics, we will integrate some existing resources based on word senses. The Multilingual Central Repository is the result of the integration of many different resources (different wordnet versions, Ontologies, SCFs lexicons) using the *de facto* standard sense repository, WordNet. The resulting MCR is going to constitute a natural multilingual large-scale linguistic resource for a number of semantic processes that need large amounts of linguistic knowledge in order to be effective tools (e.g. semantic web ontologies). The fact that word senses are linked to concepts in MCR will allow for the appropriate representation and storage of the acquired knowledge.

III.1.1 MCR Structure

MCR follows the model proposed by the EuroWordNet project². EuroWordNet is a multilingual lexical database with wordnets for several European languages, which are structured as the Princeton WordNet [Fellbaum, 1998].

The Princeton WordNet contains information about nouns, verbs, adjectives and adverbs in English and is organized around the notion of a *synset*. A synset is a set of words with the same part-of-speech that can be interchanged in a certain context.

¹This resource was developed in the framework of the MEANING project (IST-2001-34660) <http://www.lsi.up.edu/~nlp/meaning>

²<http://www.illc.uva.nl/EuroWordNet>

Synsets are related to each other by semantic relations, such as hyponymy (between specific and more general concepts), meronymy (between parts and wholes), cause, entailment, etc.

The EuroWordNet architecture includes the **Inter-Lingual-Index (ILI)**, a **Domain ontology** and a **Top (Concept) ontology** [Vossen, 1998]. The ILI consists of a list of records which interconnect word meanings in the local wordnets. During the EuroWordNet project, around 1,000 ILI-records were selected as **Base Concepts (BCs)** and consistently connected to the **Top Concept ontology**³.

Figure III.1 gives a schematic presentation of the EuroWordNet architecture. The language-independent structures are given in the middle: the ILI, a Domain Ontology and a Top Concept Ontology. The ILI consists of a list of so-called ILI-records (ILIRs) which are related to word-meanings in the local wordnets, (possibly) to one or more Top Concepts and (possibly) to domains.

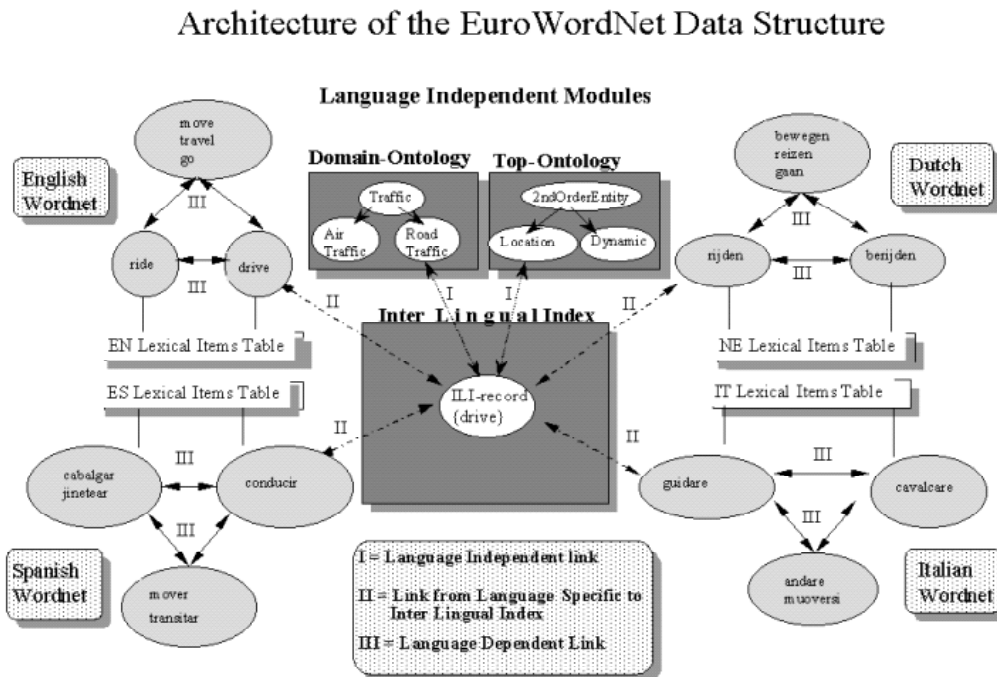


Figure III.1: EuroWordNet architecture

Using the **Inter-Lingual-Index**, wordnets are interconnected which allows to go from the words in one language to similar words in the other languages. The ILI of EuroWordNet was aligned to WordNet1.5, while the ILI of MCR has been aligned to WordNet 1.6.

³<http://www.illc.uva.nl/EuroWordNet/corebcs/topont.html>

The overall design of the EuroWordNet database made it possible to develop the local wordnets relatively independent while guaranteeing a high level of compatibility. Among the specific measures taken to enlarge the compatibility of the different resources was the definition of a common set of so-called **Base Concepts**. The Base Concepts were used as a starting point by all the sites so as to develop the cores of the different wordnets. Base Concepts are meanings that play a major role in the wordnets.

The ILI is enhanced, enriched and structured by two separate ontologies:

- The **Top Concept ontology**, which is a hierarchy of language-independent concepts (see Figure III.2) reflecting important semantic distinctions, e.g. Object and Substance, Location, Dynamic.
- The **Domain ontology**, which is a hierarchy of domain labels. The domain labels are knowledge structures grouping meanings in terms of topics or scripts, e.g. Transport, Sports, Medicine, Gastronomy.

Top ⁰	
IstOrderEntity ¹	2ndOrderEntity ⁰
Origin ⁰	SituationType ¹²
Natural ³²	Dynamic ⁴⁴⁹
Living ³⁰	BoundedEvent ²⁶⁹
Plant ¹⁸	UnboundedEvent ⁵³
Human ¹⁰⁷	Static ⁷⁵
Creature ²	Property ⁷⁰
Animal ²³	Relation ⁸⁹
Artifact ¹⁴³	SituationComponent ⁰
Form ⁰	Cause ¹¹⁰
Substance ³²	Agentive ²⁶⁶
Solid ⁶³	Phenomenal ²⁶
Liquid ¹³	Stimulating ³⁵
Gas ¹	Communication ¹⁰¹
Object ¹⁷³	Condition ¹⁰¹
Composition ⁰	Existence ⁴¹
Part ⁸⁶	Experience ¹²³
Group ⁶³	Location ¹⁷⁶
Function ⁵⁴	Manner ²¹
Vehicle ⁸	Mental ¹⁸⁴
Representation ¹²	Modal ²²
MoneyRepresentation ¹⁰	Physical ²⁹⁰
LanguageRepresentation ³⁴	Possession ³⁹
ImageRepresentation ⁹	Purpose ¹⁵⁹
Software ⁴	Quantity ⁴¹
Place ⁴⁵	Social ²²²
Occupation ²³	Time ³⁵
Instrument ¹⁸	Usage ²⁸
Garment ³	
Furniture ⁶	
Covering ⁸	
Container ¹²	
Comestible ³²	
Building ¹³	
3rdOrderEntity ³²	

Figure III.2: The EuroWordNet Top-Ontology

The main purpose of the **Top Concept ontology** is to provide a common framework for all the wordnets. However, in the EuroWordNet project, only the **Base Concepts** were classified according to the **Top Ontology**. The superindex in figure III.2 is the number of BCs associated to each Top Concept Ontology property.

On the other hand, the **Domain ontology** groups concepts in a different way, based on scripts rather than classification. The information brought in by Domain Labels is complements that already existing in WordNet.

This multilingual structure allows to port the knowledge from one WordNet to the other languages via the EuroWordNet ILI, maintaining the compatibility among them. In that way, the ILI structure (including the **Top Concept ontology** and the **Domain ontology**) will act as a natural backbone to transfer the different knowledge acquired from each local wordnet to the other wordnets, balancing resources and technological advances across languages. In the same way, all the different resources (e.g. different ontologies) could be related through the ILI, and thus cross-checked.

III.1.2 MCR Content

The MCR includes only conceptual knowledge. This means that only semantic relations between synsets will be acquired, uploaded and ported across local wordnets. However, when necessary, the relations acquired can be underspecified.

In that way, they will be uploaded and ported and will be ready to be used by other acquisition processes and languages. For instance, consider the following relation *<gain>* INVOLVED *<money>* captured as typical object. Although this relation may be further refined into *<gain>* INVOLVED-PATIENT *<money>* at a later stage, other processes can profit immediately from a ported relation, such as *<ganar>* INVOLVED *<dinero>* for Spanish.

Currently, the MCR integrates:

- ILI aligned to WordNet 1.6 [Fellbaum, 1998]:
 - EuroWordNet Base Concepts [Vossen, 1998]
 - BalkaNet Base Concepts [Cristea et al., 2003]
 - EuroWordNet Top Concept Ontology [Vossen, 1998]
 - MultiWordNet Domains [Magnini and Cavaglia, 2000]
 - Suggested Upper Merged Ontology (SUMO) [Niles and Pease, 2001]
- Local wordnets:
 - Princeton English WordNet 1.5, 1.6, 1.7, 1.7.1, 2.0 [Fellbaum, 1998]
 - eXtended WordNet (XWN) [Mihalcea and Moldovan, 2001]
 - Basque wordnet [Agirre et al., 2002]
 - Catalan wordnet [Benítez et al., 1998]
 - Italian wordnet [Pianta et al., 2002]
 - Spanish wordnet [Atserias et al., 1997]

- Large collections of semantic preferences:
 - Direct dependencies from Parsed SemCor [Agirre and Martinez, 2001]
 - Semantic preferences acquired from SemCor [Agirre and Martinez, 2001; Agirre and Martinez, 2002]
 - Semantic preferences acquired from BNC [McCarthy, 2001]
- Large collections of Sense Examples:
 - SemCor [Miller et al., 1993]
- Instances:
 - Named Entities from IRST [Pianta et al., 2002]
 - Instances from SUMO [Niles and Pease, 2001]
 - Named Entities from the work of Alfonseca [Alfonseca and Manandhar, 2002]
- Verb Lexicon
 - VerbNet [Kipper et al., 2000]

A full description of the MCR and its contents can be found in [Atserias et al., 2004f]. For simplicity, this chapter does not describe all the resources uploaded into the MCR, but only the resources that are more relevant to the experiments described in this thesis: Base Concepts, local wordnets, MultiWordNet Domains, SUMO, Top Concept Ontology, Instances, LCS and VerbNet .

III.1.2.1 Base Concepts

The main characteristic of the Base Concepts is their importance. We uploaded two main groups of **Base Concepts** in the MCR: EuroWordNet's **Base Concepts** and Balkanet's **Base Concepts**.

Although their basic aim is similar, they are built based on different criteria. EuroWordNet's **Base Concepts** were used as a starting point by all the sites in order to develop the local wordnet cores and they were classified according to the **Top Ontology**. The EuroWordNet Base Concepts were selected manually so as to cover the most important concepts of the languages involved in the project [Vossen, 1998]. The Balkanet Project⁴ also defined three different subsets of Base Concepts over WordNet1.7 [Tufis et al., 2004]. The first BalkaNet subset of BCs is the EWN BCs subset 1 (approximately 1,300 concepts). The second BalKanet subset of BCs are approximately 5,000 concepts common across all Balkan languages with high frequency occurrences. The third BalKaNet subset of BCs (about 2,500) comprises concepts selected in order to enrich the coverage of the local wordnets and fill in potential gaps in the monolingual taxonomies.

⁴<http://www.ceid.upatras.gr/Balkanet/>

III.1.2.2 WN Lexicographer Files

Synsets are organized into forty-five lexicographer files based on syntactic category and logical groupings. These lexicographer files can be also seen as a coarse-grained sense distinctions or subject codes [Rigau et al., 1997]. Table III.1 presents the synset distribution across the Lexicographer Files in WN1.6. From left to right, Lexicographer File number (LF), Frequency (#synsets) and the Lexicographer File name comprising its respective Part-of-Speech (POS).

LF	#synsets	LF	LF	#synsets	LF
00	14,734	adj.all	23	1,104	noun.quantity
01	3,099	adj.pert	24	371	noun.relation
02	3,575	adv.all	25	300	noun.shape
03	13	noun.Tops	26	2,550	noun.state
04	5,373	noun.act	27	2,392	noun.substance
05	7,295	noun.animal	28	875	noun.time
06	9811	noun.artifact	29	495	verb.body
07	2,634	noun.attribute	30	2,006	verb.change
08	1,592	noun.body	31	635	verb.cognition
09	2,261	noun.cognition	32	1,388	verb.communication
10	4,548	noun.communication	33	411	verb.competition
11	851	noun.event	34	229	verb.consumption
12	394	noun.feeling	35	1,953	verb.contact
13	2,378	noun.food	36	606	verb.creation
14	1,832	noun.group	37	303	verb.emotion
15	2,124	noun.location	38	1247	verb.motion
16	41	noun.motive	39	410	verb.perception
17	1,050	noun.object	40	688	verb.possession
18	6410	noun.person	41	1,007	verb.social
19	524	noun.phenomenon	42	671	verb.stative
20	7,873	noun.plant	43	78	verb.weather
21	908	noun.possession	44	82	adj.ppl
22	521	noun.process			

Table III.1: Semantic File distribution in WN1.6

III.1.2.3 Local WordNets

Spanish [Atserias et al., 1997], Catalan [Benítez et al., 1998] and Basque [Agirre et al., 2002] wordnets are the result of ten years of combined effort of several research centers involved in different national and international projects. Their first versions were built during the EuroWordNet project following the *expand model* [Vossen, 1998]. That is, following an automatic method and exploiting several Spanish/Catalan/Basque-English bilingual dictionaries, WordNet synsets were mapped into equivalent synsets in the local language. In that way, an aligned version of WordNet 1.5 was built.

On the other hand, the Italian WordNet [Pianta et al., 2002], developed within

the MultiWordNet⁵ project, is strictly aligned to WN1.6. In the semi-automatic construction of the Italian WordNet, the construction of the corresponding Italian synsets relies on various sources, such as Princeton WordNet and the Collins English/Italian bilingual dictionary.

Finally, newer versions of Princeton WordNet are also enriching the MCR. For instance, WordNet 2.0 comprises more than 42,000 new links between morphologically related nouns and verbs, a topical organization for many areas that classifies synsets by category, region, or usage, as well as gloss and synset corrections, and new terminology, mostly in the terrorism domain.

In this version, the Princeton team has added links for derivational morphology between nouns and verbs. Furthermore, some synsets have also been organized into topical domains. Although Princeton domains are always noun synsets, synsets from every syntactic category can be connected. Each domain is further classified as *category*, *region*, or *usage*.

III.1.2.4 MultiWordNet Domains

The initial EuroWordNet design included a Domain ontology. However, only the *Computer Domain* was included into the EuroWordNet database. Thus, instead of using the original EuroWordNet Domain ontology we uploaded the MultiWordNet Domains.

MultiWordNet Domains [Magnini and Cavaglia, 2000] were partially derived from the Dewey Decimal Classification⁶. WordNet Domains is a hierarchy of 165 Domain Labels associated to WordNet 1.6.

The information contained in these by Domain Labels complements that already available in WordNet. First of all, Domain Labels may include synsets of different syntactic categories: for instance MEDICINE groups together senses from nouns, such as *doctor* and *hospital*, and from verbs such as *to operate*.

Secondly, a Domain Label may also contain senses from different WordNet sub-hierarchies (i.e. sense that derives from different *unique beginners* or from different *lexicographer files*). For example, SPORT contains senses such as <athlete>, deriving from <life form>, <game equipment> from <physical object>, <sport> from <act>, and <playing field> from <location>.

III.1.2.5 Suggested Upper Merged Ontology (SUMO)

SUMO⁷ [Niles and Pease, 2001] has been created as part of the IEEE Standard Upper Ontology Working Group. The goal of this Working Group is to develop a standard upper ontology that will promote data interoperability, information search and retrieval, automated inference, and natural language processing. SUMO provides definitions for general purpose terms and is the result of merging different free upper ontologies (e.g. Sowa's upper ontology, Allen's temporal axioms, Guarino's formal

⁵<http://multiwordnet.itc.it>

⁶<http://www.oclc.org/dewey>

⁷<http://ontology.teknowledge.com/>

theory of parts and boundaries, etc.). There is also a complete set of mappings from WordNet 1.6 synsets to SUMO: nouns, verbs, adjectives, and adverbs.

SUMO consists of a set of concepts, relations, and axioms that formalize an upper ontology. An upper ontology is limited to concepts that are meta, generic, abstract or philosophical, and hence are general enough to address (at a high level) a broad range of domain areas. Concepts specific to particular domains are not included in the upper ontology, but such an ontology does provide a structure upon which ontologies for specific domains (e.g. MEDICINE, FINANCE, ENGINEERING, etc.) can be constructed.

The SUMO version uploaded into the MCR consists of 1,019 terms (all of them connected to WordNet 1.6 synsets), 4,181 axioms and 822 rules.

We believe that further investigation is needed to compare SUMO and the other ontological sources. For instance, the process typology in SUMO was inspired by Beth Levin's verb classes [Levin, 1993]. Among other things, this work attempts to classify over 3,000 English verbs into 48 "semantically coherent verb classes". Some of the verb classes relate to static predicates in the ontology rather than to processes, and some classes are syntactically motivated, e.g. the verb class that takes predicative complements.

Currently, only the SUMO labels and the SUMO ontology subclass relations are loaded into the MCR.

III.1.2.6 Instances

Regarding name entities and Instances, MCR integrates three different resources, from IRST [Pianta et al., 2002], the work of Alfonseca and Manandhar [Alfonseca and Manandhar, 2002], and the instance information contained in the Suggested Upper Merged Ontology (SUMO) [Niles and Pease, 2001]. The current figures for these resources are:

- 6,961 Named Entities from the work of [Alfonseca and Manandhar, 2002]
- 5,561 Named Entities from SUMO [Niles and Pease, 2001]
- 4,097 Named Entities from MultiWordNet [Pianta et al., 2002]

Although they share the basics of what an instance is, they use different criteria and granularity to classify them. Once uploaded, A new ontology of Named Entities can be built in order to support and cover the formal criteria followed by the three approaches. This initiative would be also very useful when comparing Named Entities derived using different language processors.

III.1.2.7 LCS

The **University of Maryland's Lexical Conceptual Structures (LCS) Database**⁸ [Dorr, 1993b] is based on the notion of LCS [Jackendoff, 1972]. LCS

⁸http://www.umiacs.umd.edu/~bonnie/LCS_Database_Documentation.html

is a compositional abstraction with language independent properties. It has been used, for example, as the *interlingua* representation in several Machine Translation Systems (such as UNITRAN [Dorr, 1993a] and MILT [Dorr, 1997]). Another related work for Spanish is that developed in **LEXPIR** [Fernández and Martí, 1996] and [Vázquez et al., 2000].

III.1.2.8 VerbNet

VerbNet⁹ [Kipper et al., 2000] is an enrichment of verb entries in WordNet that includes more specific syntactic information and verb class membership. It also draws heavily on the English Tree-Adjoining Grammar. Currently, the **PropBank** corpus¹⁰ ([Kingsbury and Palmer, 2002], [Kingsbury et al., 2002] and [Palmer et al., 2002]) is being extended with VerbNet semantic predicate information [Kipper et al., 2002].

III.2 The Uploading Process

This section focuses on the issues that have arisen during the integration of the above described resources into the MCR. Most of the resources uploaded into the MCR have been derived from data linked to WN1.6. However, the Basque, Catalan and Spanish WNs were aligned to WN1.5. Moreover, new and richer versions of Princeton WordNet have appeared (e.g. WordNet 2.0) together with other resources aligned to these new versions (e.g. eXtended WordNet).

Although the technology to provide compatibility across wordnets exists [Daudé et al., 1999; Daudé et al., 2000; Daudé et al., 2001]¹¹ uploading resources linked to wordnets not based on WordNet 1.6 to the MCR is a complex process.

To deal with the gaps between versions we used a set of accurate mappings between all involved English WNs so as to maintain the compatibility across wordnets. These mappings are used to build the mapping between each particular wordnet version (or the version to which a resource was aligned) and the MCR ILI (which is aligned to WN1.6). Through a particular mapping between WordNet versions, synsets can be split (1:N), joined (N:1), added (0:1) or deleted (1:0). For instance table III.2 shows, for each different case, the number of links between WN1.5 and WN1.6 synsets and the number of different synsets for each version involved.

	1:1		1:N		M:1		M:N		1:0		0:1	
	#1.5	#1.6	#1.5	#1.6	#1.5	#1.6	#1.5	#1.6	#1.5	#1.6	#1.5	#1.6
Noun	65,740	65,740	34	69	683	338	4	4	530	-	-	4,994
Verb	10,841	10,841	21	42	212	106	2	1	160	-	-	964
Adj	7,824	17,824	83	171	1,374	665	-	-	243	-	-	2,440
Adv	2,854	2,854	5	30	168	81	8	7	33	-	-	448
Total	97,259	97,259	153	312	2,437	1,190	8	7	966	-	-	8,846

Table III.2: Mapping WN1.5 → WN1.6 for Princeton WordNet version

⁹<http://www.cis.upenn.edu/verbnet/>

¹⁰<http://www.cis.upenn.edu/~mpalmer/project/pages/ACE.htm>

¹¹<http://www.lsi.upc.es/~nlp>

Before describing the particular problems of uploading each different resource in the MCR, we will explore the common issue of evaluating the impact of the alignment to a new WordNet version.

III.2.1 *Uploading Base Concepts*

The procedure for selecting the EuroWordNet Base Concepts and the Top Ontology is discussed in [Vossen, 1998]. The final set of common Base Concepts totaled 1030 WordNet 1.5 synsets.

The EuroWordNet's Base Concepts from WN1.5 have been mapped to WN1.6. After a manual revision and expansion to all WN1.6 top beginners, the resulting BC for WN1.6 totaled 1,601 ILI-records. In that way, the new version of BC covers the complete hierarchy of ILI-records. The BalKanet's and Princeton's Base Concepts were also aligned to WN1.6 from WN1.7 and WN2.0 respectively.

III.2.2 *Uploading Top Concept Ontology*

The original purpose of the EuroWordNet **Top Concept ontology** was to enforce more uniformity and compatibility of the different wordnet developments.

The uploading of the **Top Concept ontology** was performed in two steps:

1. The properties were assigned automatically to WordNet 1.6 synsets through the WN1.5-WN1.6 mapping.
2. The properties related to its semantic file were assigned to the WordNet 1.6 Tops (those synsets that has no father) which doesn't have any property assigned through the mapping.
3. Fixing 38 synsets whose set of properties assigned by hand appeared to be incompatible among them. For instance, the synset 4950638n *subject _1 topic_1 theme_1* is assigned simultaneously to *3rdOrderEntity* and *MENTAL*, which are incompatible.

III.2.3 *Evaluating the Uploading Process*

In order to illustrate the evaluation of resource re-alignment from a version of WordNet to a newer one, we will present an exhaustive analysis of the alignment process of the Spanish WN from WN1.5 to WN1.6. Similar analysis can be performed for the rest of the wordnets uploaded and could also help other WN developers to keep their local WNs up to date with respect to the latest Princeton wordnet.

Uploading wordnets not based on WordNet1.6 to the MCR is a not a simple process, because even if we perform manual checking of these connections, for those remaining cases of splitting or joining synsets the information inside the synsets should be modified accordingly to avoid inconsistencies. For instance table III.3 shows a summary of the different cases when uploading the Spanish WordNet (aligned to WN1.5), which results in the *losing* of 449 Spanish synsets.

	1:1		1:N		M:1		M:N	
	#1.5	#1.6	#1.5	#1.6	#1.5	#1.6	#1.5	#1.6
Noun	37,704	37,704	28	57	468	284	3	4
Verb	8,722	8,722	14	28	185	101	1	2
Adj	13,970	13,970	81	167	1,311	656	2	1
Total	60,396	60,396	123	252	1,964	1,041	6	7

Table III.3: Mapping synsets WN1.5 \rightarrow WN1.6 figures for SpWN

The whole process of re-aligning wordnets (synsets and relations) not aligned to WordNet1.6 to the new ILI based on WordNet1.6 consists of:

- **Synsets:** While “*local*” synsets (e.g. those created in the Spanish Wordnet) do not vary, the synsets coming from WN1.5 were mapped to WN1.6 as follows:
 1. For all splitted synsets, all information of synset 1.5, including variants, is copied to each of the equivalent synsets in 1.6
 2. For all joined synsets, all information of the original synsets including variants, is copied to the equivalent synset in 1.6
 3. Manual revision is performed to validate the split and joined synsets for the Basque/Catalan/Spanish WordNets.
- **Relations:** Since Basque, Catalan and Spanish wordnets were build semi-automatically from WN1.5, we consider that we should remove all the relations imported from WN1.5, replacing them with the relations coming from WN1.6. For the rest of Princeton WordNets the relations were not ported.

Only those relations added with respect to WN1.5 were uploaded through the mapping. Thus, even if we perform manual checking of these connections, for those remaining cases of splitting or joining synsets the information inside the synsets should be modified accordingly.

We will focus on the mapping of the synsets, because the impact for re-aligning the relations is minimum in the WordNets.

Since a manual checking of the whole mapping of the resulting Spanish wordnet will be too time consuming, we will measure the quality of the mapping, measuring how much a synset in the source-wordnet differs from their equivalents in the target-wordnet. That is, how much their contents and their relations with other synsets differ.

The mapping divides the synsets in four categories according to the cardinality of the mapping relation (1:1) (1:N) (M:1) (M:N). On the one hand, we must perform a revision of all the cases where the mappings are not one-to-one (1:1). Split synsets (*:N) must be revised because not all the information which is copied to each new resulting synset (i.e. variants/glosses/relations) will be correct. Similarly, for joined synsets (M:*), because the resulting content information will be repeated or will not be consistent.

On the other hand, the relaxation labeling algorithm used to build the mapping tries to converge to the solution (mapping) that best holds a whole set of restrictions. Although the best mapping is the most consistent, sometimes changes have to be done in order to suit the new synset. In fact, even in those cases where there is a 1:1 mapping, we need to check the quality/consistency of the equivalences between English wordnet versions.

The quality of the mappings regarding its content can be measured by comparing the original synset and their equivalent in the target versions. That is, their synonym set (exactly equal, extended, etc.) and glosses (empty, equal, extended, overlap, ...). Similarly, we also measured the changes in the relations of the synset through the mapping by measuring the changes in the WN relations.

We choose a very simple way of combining the different measures by just calculating the mean of their values. First, the quality measure for each mapping between a WN1.5 synsets an WN1.6 synsets is calculated. Then, the quality of the resulting WN1.6 synset (confidence score) is defined as the minimum of the quality of all its mappings.

The following subsections explains in details these different measures over each component of the synset: the Variant Based Measure (*QVariant*), the Lexicographer File Measure (*QSemf*), the Gloss Based Measure (*QGloss*) and Relationship Based Measure (*Qrel*)

III.2.3.1 Variant Based Measure (QVariant)

This measure is based in the overlapping between the set of variants of the source and target synset. Due to the mapping construction both sets share at least a variant. Thus, comparing the two set of variants we can find the following cases: **(EQ)** both set of variants are equal, WN1.6 variants includes WN1.5 variants (**EXTENDED**)(see figure III.3), or viceversa that is WN1.5 variants include WN1.6 variants (**REDUCED**)(see figure III.4), or in the worst case none of the sets is included in the other (**OVERLAP**) (see figure III.5). For these cases, we calculate the score as: twice the number of common variants divided by the number of variants in both synsets.

WN1.5	00003128-r	just#1	merely#1	only#1	simply#1	
WN1.6	00003737-r	but#1	just#1	merely#1	only#1	simply#1

Figure III.3: Wn1.6 Extends the Wn1.5 variants

WN1.5	00003345-v	hiccough#1	hiccup#1	make_a_hiccup#1
WN1.6	00002841-v	hiccough#1	hiccup#1	

Figure III.4: Wn1.6 Reduces the Wn1.5 variants

WN1.5	00022594-r	almost#2	close_to#1		
WN1.6	00006065-r	about#1	approximately#1	around#5	
		close_to#1	just_about#2	more_or_less#1	
		or_so#1	roughly#1	some#1	

Figure III.5: Wn1.6 and Wn1.5 variant set are not subsets of each other

III.2.3.2 Lexicographer File Measure (QSemf)

This measure is based in the overlapping between the lexicographer file of the synset in WN1.5 and the mapped WN1.6 synset. This measure scores 1 if equal and -1 otherwise (there are only 431 synsets whose Lexicographer File differs through the mapping).

III.2.3.3 Gloss Based Measure (QGloss)

This measure is based in the overlapping between the words of the glosses. Before comparing glosses, the possible examples included in the gloss are removed. Obviously these measures can be considerably improved by stemming, lemmatising or parsing the glosses.

In a similar way than with the variants, both glosses could be equal (**EQ**), equal except for the text inside parenthesis (**NEAREQ**), the WN1.5 gloss could be a part of WN1.6 gloss (**EXTENDED**) or viceversa (**REDUCED**) or simple share some common words (**OVERLAP**) as in Figure III.6. When mapping to WN1.5. there is another special case, (**NULL**) when the method can not be applied as not all WN1.5 synsets have a gloss.

WN1.5	00005659-a	being the most complete of its class
WN1.6	00005386-a	being the most comprehensive of its class

Figure III.6: Overlap Glosses

For these cases, we calculate the score as: twice the number of common words divided by the number of words in both glosses or zero if the method is not applicable (i.e. WN1.5 synset has not gloss).

III.2.3.4 Relationship Based Measure (Qrel)

Once all the WN1.5 synsets are mapped to WN1.6, the relations can be also mapped accordingly. This measure is based in the overlapping between the set of relations of one synset in WN1.5 and their equivalent/s in WN1.6.

- **EQ**: All the WN1.5 relations have a corresponding WN1.6 relation.
- **CHANGED**: When some WN1.5 relations do not have a corresponding WN1.6 relation. Then, the quality is calculated as the relation kept in WN1.6 divided by the number of relations from WN1.5.
- **NONE**: None of the WN1.5 relations has the corresponding relation in WN1.6.

III.2.3.5 Results

Combining¹² all these measures we can evaluate the impact of re-aligning a resource to a new wordnet version. Following the SpWn example, we should point out that using the mapping between WN1.5 and WN1.6, almost half of the mapped synsets (42,161) have exactly the same variants and glosses. Table III.4 shows the quality per POS of the 1:1 mapping. A global quality measure of 0.88 means that the impact in the Spanish WordNet will be minimum. However, verb glosses seems to be more difficult than for the rest of POS and the relation measure is quite low in adjectives (0.66), maybe because there are few connections among them.

POS	QVar	QGloss	QSem	QRel	Quality
<i>noun</i>	0.85	0.92	0.99	0.76	0.88
<i>verb</i>	0.91	0.77	0.99	0.92	0.90
<i>adj</i>	0.94	0.81	0.98	0.66	0.84
total	0.88	0.87	0.99	0.76	0.88

Table III.4: Quality measure for 1:1 WN1.5 to WN1.6 for Spanish wordnet.

Table III.5 shows the figures for SpWn1.5, the number of synsets which come from Princeton WN1.5, the number of “*local*” synsets (i.e. those synsets not appearing in the English WordNet), the resulting synsets aligned to Princeton WN1.6 after the mapping and the final figures for SpWn1.6. As we can observe there is no much change in coverage.

pos	Spwn1.5	syn1.5	local	syn1.6	Spwn1.6
<i>noun</i>	43,652	38,308	5,344	38,023	43,367
<i>verb</i>	9,258	9,045	213	8,830	9,043
<i>adj</i>	15,859	15,585	274	14,667	14,941
total	68,769	62,938	5,831	61,520	67,351

Table III.5: Figures for Spanish WordNet aligned to wn1.6

For simplicity, in the following sections we will mainly focus on those resources which will be used in the experiments on the following chapters, that is, local wordnets, Top Concept Ontology, SUMO. For the rest of the resources (extended WordNet, selectional preferences, semcor examples, etc) an exhaustive description of their uploading can be found in [Atserias et al., 2004f], [Atserias et al., 2004e], [Atserias et al., 2004d] and [Atserias et al., 2004c].

¹²As explained before to combine these measures we calculate the mean.

III.3 The Integration Process

Once all the different resources are correctly uploaded into the MCR, three different subprocesses can be devised inside the integration process: realisation, generalization and cross-checking. By *realisation* we mean the process of making explicit all the knowledge contained into the MCR (e.g. expanding by inheritance, top-down through the hierarchy, relations or properties), while by *generalization* we mean a bottom-up mechanism to collapse or generalize on a particular Base Concepts and ontological nodes different knowledge from the MCR. Both processes, realization and generalization, can take advantage of other resources, that is cross-checking the different knowledge sources.

III.3.1 Realisation

Realisation is the process of making explicit by inference (in particular, inheritance) all the knowledge contained into the MCR. That is, expanding top-down relations or properties, or making inferring knowledge using other properties e.g. transitivity.

For instance, once all this data is uploaded into the MCR, it is possible to perform a full expansion process of the Top Ontology properties associated to the Base Concepts through the nominal and verbal hierarchies.

Some of the selectional preferences acquired from SemCor and BNC in the MEANING project and uploaded in the MCR could also be inherited through the nominal part of the hierarchy. This process involves a huge computational effort.

III.3.1.1 Realization of the Top Concept Ontology

The EuroWordNet project only performed a complete validation of the consistency of the **Top Concept ontology** of the Base Concepts. However, the classification of WordNet is not always consistent with the **Top Concept ontology**.

In order to make explicit the **Top Concept ontology** properties for all the synsets, we should propagate top-down the **Top Concept ontology** properties assigned to the Base Concepts. That is, we will enrich the complete ILI structure with features coming from the BC by inheriting the Top Concept features following the hyponymy relationship.

This way, once the ontological properties are exported to the ILI and inherited through the complete WN hierarchy, all WN concepts will have associated a set of semantic features as in the example shown in table III.6.

lentil_1	
DOMAIN	gastronomy
LF	food
SUMO	FruitOrVegetable
TCO	Comestible ; Plant

Table III.6: lentil_1

We use the following modifiers associated to the TCO properties to state whether the property has been, directly assigned (=), inherited using the top-down propagation using the wordnet structure (+) or assigned using the WordNet Lexicographer File (#).

In order to provide consistency to the inheritance process we used the following basic incompatibilities among TCO properties (furtherly expanded to their daughter concepts) which were defined inside the EWN project:

- substance - object
- plant - animal - human - creature
- natural - artifact
- solid - liquid - gas

These incompatibilities impeded a fully automatic top-down propagation of the TCO properties. That is, when any of the current **Top Concept ontology** properties of a synset is incompatible with the property currently expanded, this property is not assigned to the synset and the propagation to the synset's descendants stops.

That full-automatic process resulted in a number of synsets showing non-compatible information. Specifically:

- Sticking to TCO and according to the set of incompatibilities, some TCO properties assigned by hand appeared to be incompatible with either (a) inherited information, (b) information assigned via equivalence to LF
- TCO properties, either original or inherited, are suspicious to be incompatible with other Sources of Ontological Meaning (henceafter SOM).

By manual examination of a subset of synsets, we realised that there are at least the following main sources of errors: erroneous hand-made TCO mappings, erroneous statements of equivalence between TCO properties and LFs, erroneous ISA links in WN -which causes erroneous inheritance [Guarino and Welty, 2000]- and also multiple inheritance within WN. which can cause incompatibilities in inheritance of properties

The example shown in table III.7 has incompatible information. 3rdOrderEntity can not coexist with properties only attributable to *Events* (e.g. *Cause*).

00660718-v process_1	
DOMAIN	factotum
LF	act
SUMO	IntentionalProcess
TCO	<i>3rdOrderEntity Cause Mental Purpose</i>

Table III.7: 00660718-v process_1

Multiple inheritance could also bring up a new type of problems. Figure III.8 shows another example where multiple inheritance will lead to inherited incompatible attributes: *Artifact* from and *Natural* from *organic_compound_1*.

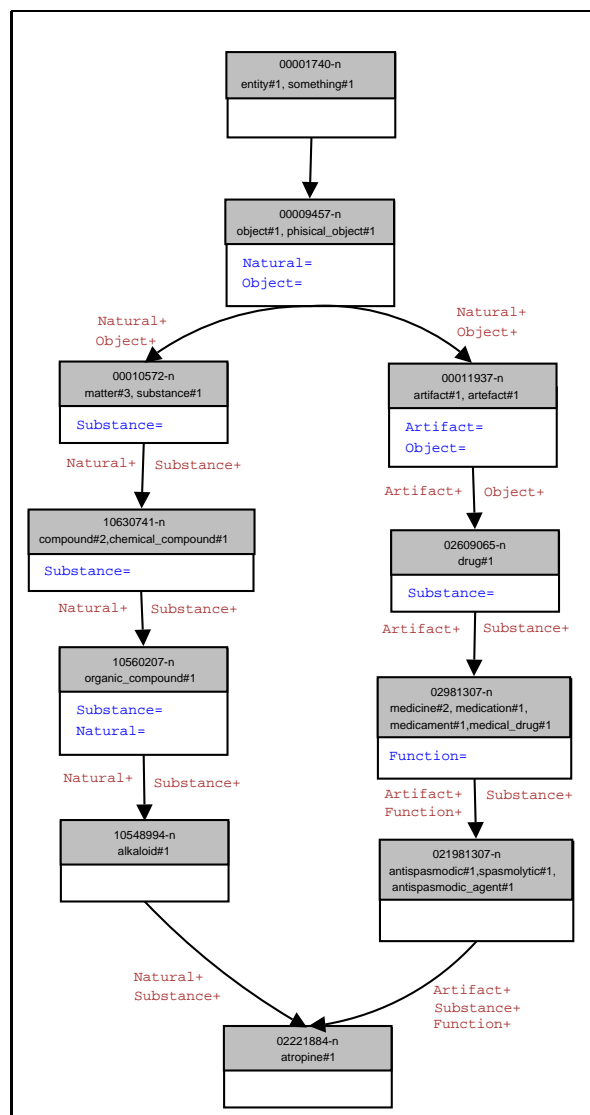


Figure III.8: Multiple inheritance for *atropine#1*

In this fully-automatic expansion the blocking points are established when incompatible properties are assigned to the same synset. Although we can stop the top-down propagation of the feature, it is not possible to automatically determine the correct assignments.

Thus, in order to solve the TCO incompatibilities, we proposed a semi-automatic top-down procedure based on resolving these conflicts either by means of removing erroneous properties or by establishing blocking-points where there is an erroneous hypernym relation.

The methodology is as follows:

1. Hand-fixing TCO mappings when appearing incompatible properties
2. Setting inheritance-blocking-points and hand-fixing TCO mappings around these points (i.e. all involved hypernyms and hyponyms)
3. Recalculating the inheritance according to the information obtained in (1) and (2)
4. Reexamining the involved subtrees to check whether re-calculation of the inheritance produces new incompatibilities
5. Exporting the mappings and blocking-point information to the ILI.

It should be noticed that it is important to export also blocking-point information to the ILI in order to ease future correct exportation of TCO's information to other wordnets, i.e. to prevent incorrect expansion of properties by inheritance.

Inside a particular wordnet, when reaching a blocking point, a subsumption link can be considered as broken for ontological purposes –therefore, it will be assumed that the conceptual chain only proceeds upwards consistently to the TCO (not to the hypernym synsets), via the ILI-records.

This process can be applied iteratively looking for suspicious synsets in WN. In that way, 38 synsets showing incompatibility between hand-assigned TCO properties were fixed.

The next step will be to check the set of WN top beginners which only bear information mapped via the TCO-LF table of equivalence (see appendix D) and finally to check synsets showing incompatibility between information directly mapped via TCO and information mapped via the TCO-LF table of equivalence. Finally, we will check the remaining cases of incompatibility between TCO manual and inherited information.

This realisation process was started inside the MEANING project and currently it is still under development:

1. Fixing the properties of those synsets having contradictory TCO properties. In that way, TCO assignments are fixed in the synset and its immediate relatives (mainly hypernym and hyponyms). All these synsets are marked as "hand-checked". The result is a correct TCO information assigned to several synsets as in the following example where, originally, non-agentive and non-intentional **00661612-v stiffening_1** was inheriting all of the **00660718-v process_1** properties as shown in table III.8

00660718 process_1	
<i>TCO</i>	Dynamic Agentive Purpose
00661612 stiffening_1	
<i>TCO</i>	Dynamic Cause

Table III.8: 00660718 process_1 and 00661612 stiffening_1

2. For those synsets having false WN subsumptions, we will introduce a blocking point between a pair of synsets. The result will be a list of blocking points, e.g.: between synsets <stiffening> and <process>.
3. We keep record of TCO-LF erroneous equivalences, since they will be useful in the future to detect more synsets with erroneous mappings. The result will be a list of suspicious TCO-LF equivalences, e.g.: [TCO:Agentive-LF:ACT]
4. To study TCO-SUMO equivalences in such synsets. As in the previous step, they can be useful in the future to detect more synsets with mistaken mappings. The result will be a list of incompatible TCO-SUMO concepts, e.g.: [TCO:3rdOrderEntity-SUMO:Physical]
5. To inspect as well WN Domain assignments. The result will be a list of doubtful WN Domain assignments, e.g. 00364173-n#play_3:ENTERPRISE

Following an iterative and incremental approach, the inheritance has been recalculated, the resulting data has been re-examined, and the eventual correct information has been again uploaded into the MCR thus overwriting the pre-existent one.

Although such hand-checking is extremely complex and delicate, we expect the task to be affordable since critical conflicts seem to concentrate in a workable layer of synsets close to the higher part of the WN hierarchy [Atserias et al., 2005].

III.3.1.2 Realization of the Meronymy relation

Figure III.9 presents a partial view of WN1.6 where solid lines represent direct connections between synsets and dotted lines represent indirect or inferred connections. Using the WN browser provided by Princeton we can ask for the direct and inherited *PART-OF* relations of a particular sense. A direct *PART-OF* relation occurs between *plant_2* and *plant_part_1*, and inherited *PART-OF* relation occurs between *apple_2* and *plant_part_1*. However, while for *succulent_1*, the browser provides an inherited *PART-OF* relation to *plant_part_1*, for *cactus_1* the browser does not provide any inherited relation at all. Consider now the following simple questions:

1. Does a cactus have leaves?
2. Does an orchard apple tree have leaves?
3. Does an orchard apple tree have fruits?

Obviously, this simple questions only can be answered applying a systematic inference mechanism on WN [Harabagiu and Moldovan, 1998]. What should be the correct behaviour of the mechanism for inheriting correctly the *PART-OF* relation through the entire hierarchy of WN?

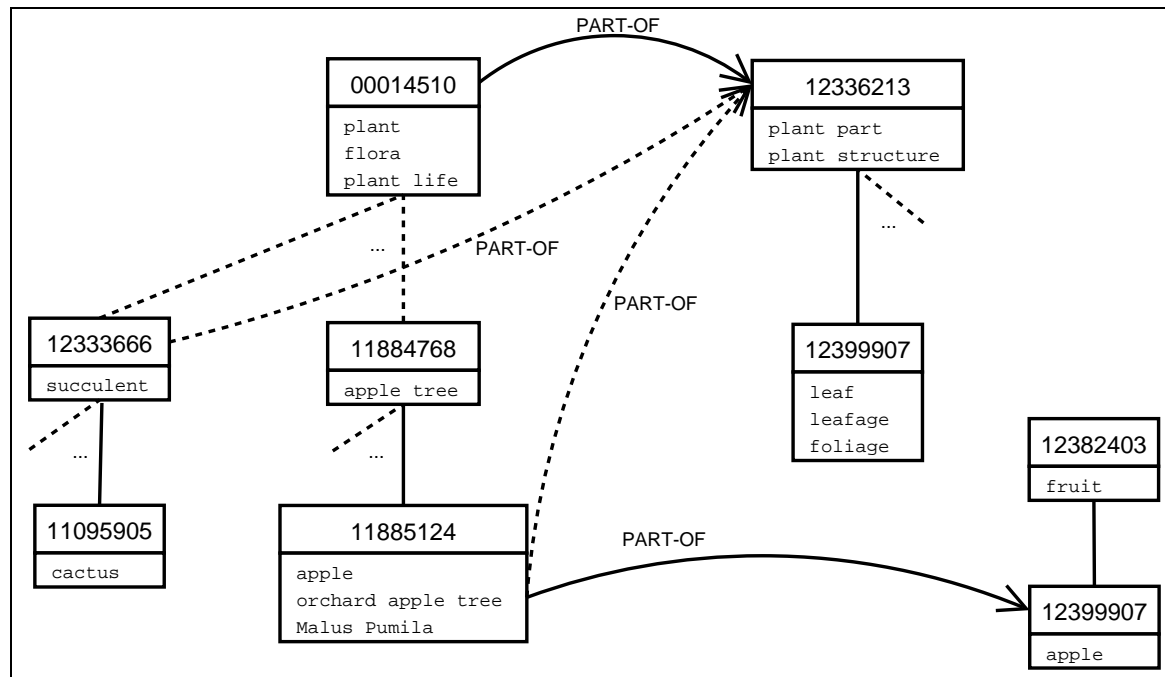


Figure III.9: Partial view of WN1.6

In order to test an inference mechanism on the *PART-OF* relation we implemented the following inference rules:

$$\begin{aligned}
 (\mathbf{A} \text{ has_hyperonym } \mathbf{B}) \wedge (\mathbf{B} \text{ has_hyperonym } \mathbf{C}) &\Rightarrow (\mathbf{A} \text{ has_hyperonym } \mathbf{C}) \\
 (\mathbf{A} \text{ has_hyponym } \mathbf{B}) \wedge (\mathbf{B} \text{ has_hyponym } \mathbf{C}) &\Rightarrow (\mathbf{A} \text{ has_hyponym } \mathbf{C}) \\
 (\mathbf{A} \text{ has_mero_part } \mathbf{B}) \wedge (\mathbf{B} \text{ has_mero_part } \mathbf{C}) &\Rightarrow (\mathbf{A} \text{ has_mero_part } \mathbf{C}) \\
 (\mathbf{A} \text{ has_hyperonym } \mathbf{B}) \wedge (\mathbf{B} \text{ has_mero_part } \mathbf{C}) &\Rightarrow (\mathbf{A} \text{ has_mero_part } \mathbf{C}) \\
 (\mathbf{A} \text{ has_mero_part } \mathbf{B}) \wedge (\mathbf{B} \text{ has_hyponym } \mathbf{C}) &\Rightarrow (\mathbf{A} \text{ has_mero_part } \mathbf{C})
 \end{aligned}$$

The first two inference rules only represent the transitivity of the *IS-A* relation. The same holds for the third with respect *PART-OF* relation. The fourth inference rule will allow to inherit the *PART-OF* relation through an hypernym chain. For example, the relation (*cactus_1 has_mero_part plant_part_1*). The last inference rule will allow to propagate a *PART-OF* relation through an hyponymy chain. For example, the relation (*plant_2 has_mero_part leaf_1*). The resulting inferences derived by this rule are not precise. In a sense, these are abductions. For example, a *tree_1* do not have as *PART-OF* all possible hyponyms of *fruit_1*. Moreover, the combination of the last two inference rules will also allow to produce relations such as (*cactus_1 has_mero_part leaf_1*).

Notice that when applying these five simple inference rules we will obtain direct answers for the first three questions mentioned above.

tree_1	
DOMAIN	botany
LF	plant
SUMO	FloweringPlant+
TCO	Group+
TCO	Living+
TCO	Object=
TCO	Plant=

Table III.9: tree_1 synset

When applying systematically these inference rules using an inference mechanism on a particular synset we obtain large collections of new explicit and inferred *PART-OF* relations. However, most of them are completely erroneous relations. For instance, for *tree_1* (table III.9 presents their main characteristics uploaded into the MCR) we obtain 2,423 new *PART-OF* relations. However, most of them are erroneous because they are violating ontological properties. Taking a random sample of 100 proposed *PART-OF* relations only 37 were correct.

For instance, we are obtaining in that way *PART-OF* relations for *tree_1* corresponding to all *body_part_1* hierarchy (e.g. *finger_1*). This inference is produced because of the following inference chain:

tree_1 —ISA→ life_form_1 —PART-OF→ body_part_1 ←ISA— finger_1

The problem now is how to solve this unwanted phenomena produced by the structure of WN. Table III.10 and III.11 presents, respectively, the main characteristics uploaded into the MCR of *finger_1* and *apple_1*.

finger_1	
DOMAIN	anatomy
LF	body
SUMO	BodyPart+
TCO	Part+
TCO	Living+

Table III.10: finger_1 synset

apple_1	
DOMAIN	botany
DOMAIN	gastronomy
LF	food
SUMO	FruitOrVegetable+
TCO	Part+
TCO	Living+
TCO	Comestible+
TCO	Function+
TCO	Natural+
TCO	Object+
TCO	Plant+

Table III.11: apple_1 synset

While an *apple_1* can be part of *tree_1*, a *finger_1* can not. Thus, we suggest to use the TCO properties associated to a particular synset as blocking marks to impede further inference propagation beyond this synset. Both *tree_1* and *apple_1* share *Living* and *Plant* TCO properties¹³. Moreover, when applying the inference rules to propagate *PART-OF* relations we can also include the TCO *Part* property as a constraint.

¹³In a corrected version of the TCO, *tree_1* should have also the *Natural* property too

In order to demonstrate the feasibility of this approach, we select those *PART-OF* relations whose target has the TCO property *Plant* from a total number of 2,423 new *PART-OF* relations for <tree_1>, obtaining 583 possible *PART-OF* relations. Taking a random sample of 100 proposed *PART-OF* relations all of them were possible *PART-OF* of a *tree*.

Finally, as a validation methodology, we also suggest performing a cycling process TCO revision/enrichment of the selected BC by means of this powerful inference mechanism. Obviously, as a side effect we can also obtain an enriched version of the MCR having thousands of new validated relations.

III.3.2 Generalisation

A similar process can be devised in order to expand the knowledge into the MCR. In this case, rather than expanding top-down the knowledge and properties represented into the MCR, a bottom-up generalisation mechanism can be performed. In this case, different knowledge and properties can collapse on particular Base Concepts and ontological nodes.

III.3.3 Cross-Checking

The integration of all these resources into a single platform both demands and allows for cross-checking. For instance, we can improve SUMO labels and WordNet Domains mappings by merging and comparing them.

Synset	Word	SUMO	Domain
00003142-v	exhale	Breathing	medicine
00899001-a	exhaled	Breathing	factotum
00263355-a	exhaling	Breathing	factotum
00536039-n	expiration	Breathing	anatomy
02849508-a	expiratory	Breathing	anatomy
00003142-v	expire	Breathing	medicine

Table III.12: SUMO vs. Domain labels

To illustrate how we can detect errors and inconsistencies between different types of SOM, we can see in the example in table III.12 that the nouns corresponding to the SUMO process *Breathing* has been labelled with ANATOMY domain, some verbs with MEDICINE and some adjectives with FACTOTUM, when in fact, all these senses correspond to different Part-of-Speech of the same concept.

On the other hand, once all the TCO properties have been fully expanded as shown in the previous section, the resulting Top Concept Ontology can be cross-checked against the Lexicographer Files from WN or against SUMO and the realization can take advantage if this cross-checking.

Cross-checking can also show up differences in criteria. For instance, **Animal vs Plant** for the synset 00911639-n *phytoplankton_1* (SUMO Plant+) and its direct descendant 00911809-n *em planktonic_algae_1* (SUMO Alga). It can also show inconsistencies due to the different granularity, as for **Human vs Animal**: i.e. all the Hominids are considered *animal* by the Lexicographer File, but they are labelled as *Human* by the Top Concept ontology (SUMO Hominid+) or **Human vs Creature**: all the creatures (mainly the descendants of *imaginary_being_1* *imaginary_creature_1*) are classified as *person* by the Lexicographer File, but as CREATURE by the TCO.

Cross-checking could also be used to enrich each one of the resources. For instance, there are synsets whose complete set of attributes of the TCO can not be inferred top-down from the hand-made assignments. Another way to automatically enrich wordnet with more TCO attributes is using the Lexicographer File. For example, the synset 10960967-n *first_half* only has the attribute *Part*. But its Lexicographer File is **noun.time** thus the associated TCO property **Time** could be added. Similar methodologies could be applied using SUMO, e.g. *first_half*, the SUMO label is *TimeInterval+*, which could also be mapped to the TCO property TIME.

Next two subsections will describe a basic cross-checking between resources on Instances and Base Concepts.

III.3.3.1 Instance-Name Entities Cross-Checking

MCR contains instance information provided by IRST [Pianta et al., 2002] and SUMO [Niles and Pease, 2001] which was already related to WN1.6, but also from [Alfonseca and Manandhar, 2002] 6,961 instances in WN1.7 automatically identified (7,033 WN16 synsets).

Table III.13 shows the intersection between each pair of these resources. These three resources together identify about 10,000 synsets as instances. However, only 1,994 synsets are identified as instances by the three resources. Merging all the information about instances not only can help to complete (e.g. adding the class of to the instance) and to correct each resource but also can help to establish a criteria about what is an instance or help to build a richer NE classification.

	IRST	SUMO	Alfonseca
IRST	4,097	2,048	2,063
SUMO	-	5,561	3,177
Alfonseca	-	-	7,033

Table III.13: Instances overlapping for wn1.6 ILIs

III.3.3.2 BCs

The Balkanet Project enlarged the set of BCs defined in EuroWordNet adding first, about 5,000 concepts BCs common across all Balkan languages with high frequency occurrences, and second about 2,500 BCs to enrich the coverage of the wordnets in order to fill potential gaps in the monolingual taxonomies.

Similarly, according to our pragmatic point of view, a concept is important if it is widely used, either directly or as a reference for other widely used concepts.

Importance is thus reflected in the ability of a concept to function as an anchor to attach other concepts. This anchoring capability was defined in terms of three operational criteria that can be automatically applied to the available resources:

1. the number of relations (general or limited to hyponymy).
2. being widely used by several languages
3. high position of the concept in a hierarchy

However, the definition of Base Concepts in EuroWordNet could not use the sense frequency information currently available in the Princeton WordNet¹⁴. It is possible to devise a simple and fully automatic method to derive the Base Concepts from the information inside the MCR following the operational criteria defined above. However, we consider that the BCs should be general enough (being in the high part of the hierarchy) but also particular enough (being in the lower part of the hierarchy) to represent the main characteristics of each concept represented in the MCR.

#occur.	#rel.	offset	synset
2338	18	00017954-n	group_1,grouping_1
0	19	05962976-n	social_group_1
729	37	05997592-n	organisation_2,organization_1
30	10	06002286-n	establishment_2,institution_1
15	12	06023733-n	faith_3,religion_2
62	5	06024357-n	Christianity_2, church_1 ,Christian_church_1
11	14	00001740-n	entity_1,something_1
51	29	00009457-n	object_1,physical_object_1
1	39	00011937-n	artifact_1,artefact_1
68	63	03431817-n	construction_3,structure_1
50	79	02347413-n	building_1,edifice_1
0	11	03135441-n	place_of_worship_1,house_of_prayer_1,house_of_God_1
59	19	02438778-n	church_2 ,church_building_1
25	20	00017487-n	act_2,human_action_1,human_activity_1
611	69	00261466-n	activity_1
2	5	00662816-n	ceremony_3
0	11	00663517-n	religious_ceremony_1,religious_ritual_1
243	7	00666638-n	service_3,religious_service_1,divine_service_1
11	1	00666912-n	church_3 ,church_service_1

Table III.14: Hypernym chain for all senses of the noun church in WN1.6

Table III.14 presents the hypernym chain for all the senses of the noun *church* in WN1.6. For each synset we show the result of summing up all the sense frequency counts appearing in SemCor (*#occur*)¹⁵ and the total number of direct relations (*#rel*). Having calculated these two numbers for each synset (both representing the

¹⁴WordNet started to contain sense frequency information derived from SemCor and other materials in version 1.6

¹⁵For the rest of languages there is not available a sense tagged corpora for all words.

first two criteria defined above), a very simple arithmetic operations can be devised to obtain automatically a set of BC for these particular synsets.

We suggest to study the following bottom-up approach to derive by automatic means the whole set of BCs. Following bottom-up the hypernym chain, we can obtain for both, the number of occurrences and the number of relations, the first local maxima of each synset. For instance, for **church_1** the first local maximum for #occur corresponds to *organization_2* (with 729 occurrences), and for the #rel corresponds to *faith_3* (with 12 relations). For **church_2** the first local maximum for the #occur corresponds to *construction_3* (with 68 occurrences), and for the #rel corresponds to *building_1* (with 79 relations). Finally, for **church_3** the first local maximum for the #occur corresponds to *service_3* (with 243 occurrences), and for the #rel corresponds to *religious_ceremony_1* (with 11 relations). Obviously, both criteria can also be combined. Furthermore, we suggest to collect all local maxima for each leaf of the WN hierarchies. All these local maxima can constitute the new Base Concepts of the MCR.

III.4 Porting Process

Having all this types of different knowledge and properties completely expanded and covering the whole MCR, a new set of inference mechanism can be devised in order to further infer new relations and knowledge. For instance, new relations can be generated when detecting particular *semantic patterns* occurring for some synsets having certain ontological properties, for a particular Domains, etc. That is, new relations can be generated when combining different methods and knowledge. For instance, when several relations derived in the integration process have particular confidence scores greater than certain thresholds. Moreover, without having inferred extra knowledge in the porting process all the knowledge integrated into the MCR can be ported (distributed) to the local wordnets.

All wordnets can gain some kind of new knowledge coming from other wordnets by means of the porting process. A direct result of the upload/integration/porting effort is that all information associated to the ILLs is automatically ported to the other wordnets. Thus, MultiWordNet Domains are now available to the rest of local wordnets, EuroWordNet Top Concept Ontology is also available for Italian MultiWordnet and for English WordNet 1.6. Moreover, local relations can be ported to the rest of wordnets. Thus, Italian and English Wordnet can be enriched with all the new set of relations coming from EuroWordNet. In turn, Basque, Catalan, Italian and Spanish wordnets can be extensively enriched with the large amounts of selectional preferences acquired from English.

However, the Porting issues are out of the scope of this thesis. Although, the detailed figures of the Porting Process as well as the issues rised by the three rounds of this process during the MEANING project are available at [Atserias et al., 2004b].

The Appendix C shows two examples (for all the senses for the Spanish words *vaso* and *pasta*) of the knowledge uploaded into the MCR and how this knowledge could characterize all the different sense of those words.

CHAPTER IV.

Process Integration in PARDON

As the problems are new, we must disentrall ourselves from the past.

Abraham Lincoln

IV.1 Introduction

As seen in the first chapter, NLU architectures are basically determined by both Process and Knowledge integration. The architecture presented in this work, PARDON, aims to give a general framework in which different NLP tasks can be easily formalized. So that, these different tasks can be tested separately or carried out simultaneously.

PARDON aims to explore the limits of the current NLP technologies, without wrongly filtering partial solutions, or over-constraining the interaction between processes/knowledge. We use Consistent Labeling Problem (CLP) as the framework to integrate different NLP processes and to apply any kind of knowledge (syntactic, semantic, linguistic, statistical) at the earliest opportunity, while retaining an independent representation for every kind of knowledge.

In previous chapters, regarding knowledge integration, we have adopted an hybrid and simple approach. No claim of completion have been made. Different resources and knowledge repositories are different views of the language and the world. None of them can claim to cover completely the richness of the language. All these sources of knowledge do not need to be equivalent nor even compatible as they will stand as independent information. Even when different knowledge/views could become incompatible or contradictory, CLP will also give us a natural way to integrate them.

Regarding process integration, CLP is a framework that allows to fusionate also processes, as a set of constraints. Thus, as long as we can relate each different source of knowledge, and processes can be viewed as satisfying a certain set of constraints, CLP will allow to integrate both, knowledge and processes.

Then, NLP tasks will be regarded as an optimization problem, by means of transforming the appropriate pieces of knowledge and processes in a set of constraints and trying to find a solution that satisfies them to the maximum possible degree.

In order to integrate processes, since we aim that this architecture could be applied to different NLP tasks, we need a knowledge representation as neutral as possible, but at the same time, powerful enough to deal with complex NLP tasks such as Semantic Interpretation.

A basic principle in semantics which guides most theories is compositionally. A compositional theory of meaning will have a representation of the contribution of each word and sub-phrase towards the meaning of the whole. The meaning of a component is its contribution to the meaning of any complete sentence of which it could be a part.

A flexible way to represent Semantic Interpretations is an **Object Oriented Semantics** approach (e.g. ABSITY [Hirst, 1987]), which can be roughly seen as an equivalent view of Object Oriented Programming paradigm but for semantics¹. Figure IV.1 shows a frame representation for the semantics objects *cat* and *fish*.

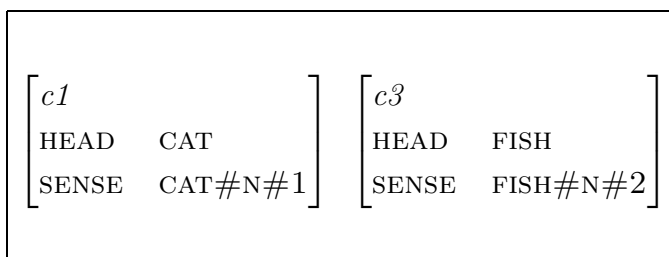


Figure IV.1: Semantic Representation for *cat* and *fish*

In an **Object Oriented Semantics** approach, semantic objects are build *Compositionally*. That is, a syntactically well formed component of a sentence corresponds to a semantic object.

The semantic object retains its identity even when it is part of a larger semantic object. For example, a semantic representation for the sentence “*The cat eats fish*” could be composed by three objects, the *cat* and *fish* objects of type *animal* and the whole sentence object of the type *eat-event* which includes the two previous ones.

¹Recently the Object Oriented Programming nomenclature has broken through NLU researchers, (e.g. Dialog Objects in the RoBoDiMa Speech Dialog System toolkit [Quast et al., 2003])

Figure IV.2 shows how the main wellformed syntactic structures of the sentence relate to each part of the resulting semantic representation. In order to represent the semantic interpretation of a sentence we have to represent the semantic objects and the relations between those objects. A *lexicon* maps the input (e.g. words or phrases) to their semantic objects which permits accessing any knowledge related to the word.

Inside PARDON we will adopt this “neutral” compositional object-oriented approach. PARDON uses a particular frame-like representation as the knowledge structure to represent objects. PARDON represents the relationships between objects in a dependency-like style, through *models* and *roles*. An object could have several models associated. However, an object can use only one of these models to combine itself with other objects (composition). We will say that, those objects, which the model combine, are *playing a role* in the model. Objects are triggered by the input (e.g. words) and are in charge of allowing access to all their related knowledge.

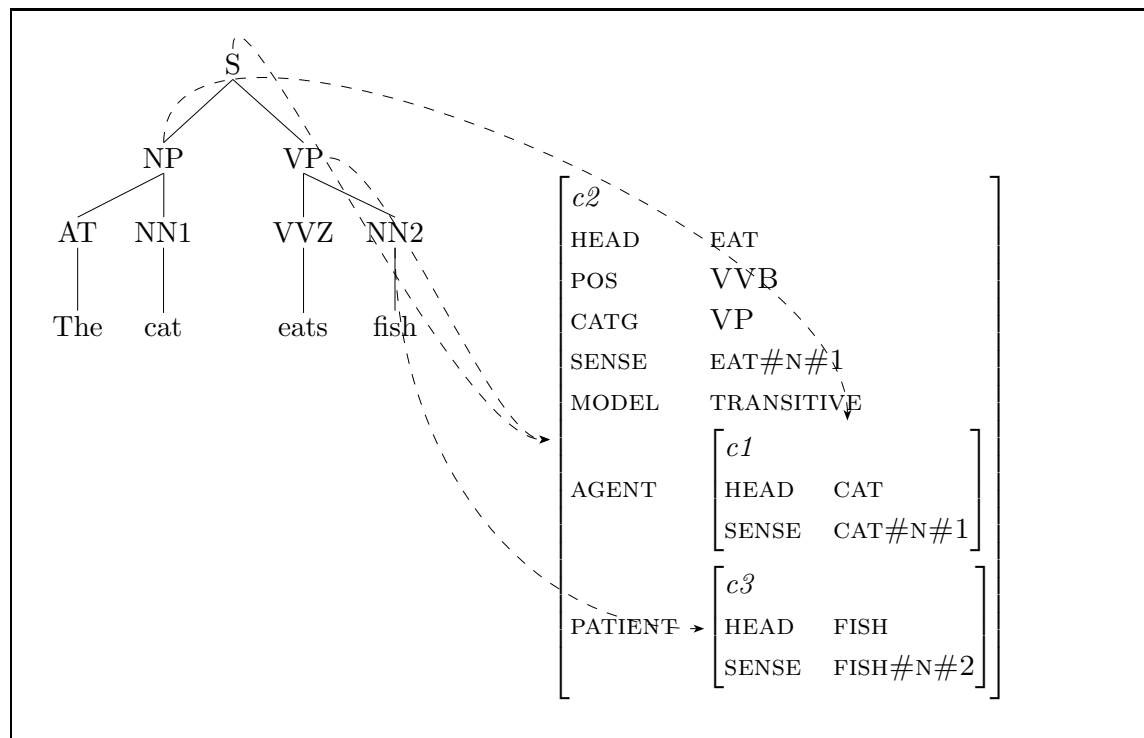


Figure IV.2: Example of compositionally

IV.2 PARDON's Architecture

PARDON's architecture is based on the idea of *compositionally*. An element combines itself with other elements to build a new element. In most cases the new element shares or contains the representation of the combined elements. Elements can not be freely combined. The correct combinations of elements are determined by models and these models are associated to the initial elements.

Thus, the compositional system is formed by a set of combinatorial patterns or rules (*models*) associated to initial elements (hencefore *initial objects*). The system has to establish not only which combination of objects (derivational sequence) can be correctly performed to cover the whole sentence, but also which ones are more plausible than the others. We have restricted the combinational process to best suit the kind of models we applied in the two test tasks, but other kinds of model matching and combination criteria could be established and formalized in different ways inside PARDON's architecture.

A frame-like semantic representation, as well as the compositional and pattern matching process of PARDON's architecture, could be formalized as a CLP. This formalization can be done in many different ways. Different formalization could lead to different performance or even to converge to different solutions when using algorithms that do not assure the global optimization. Unfortunately, there are few works on the impact of the different possible modelizations in the performance [Borrett and Tsang, 1996] and besides for empirical results, it is not clear which general properties must hold a good formalization.

Thus, PARDON's Architecture is similar to a rule-based system and has three main components:

- **Knowledge Representation:** How the information, either for partial analysis or the whole sentence, are represented.
- **Model Application:** In which conditions and how a model is applied. In most cases, the application (or learning) of models involves the definition of a similarity function, a distance or some kind of unification process or pattern matching. These mechanisms allow to compare the models and the input. So that, several parts of the model/pattern could be identified in the input.
- **Inference Engine:** How and when it is decided to apply a model.

Next sections will present this three components and how they can be formalized in the CLP framework. In order to illustrate the nature of the architecture we will use a simple example simulating the behaviour of a well-known rule-base system, the application of a Context Free Grammar (CFG). In this example, the input of the system will be words and the output a parse tree.

IV.3 Knowledge Representation in PARDON

A frame-like representation can be straightforwardly formalized in a CLP by representing each slot-value as a pair of variable-value. When the attribute contains a complex structure, we will use a reference. Figure IV.3 shows the equivalent CSP representation for the frame *cat* in figure IV.1).

Variable	Values
C1.POS	{ NN1 }
C1.HEAD	{ cat }
C1.SENSE	{ cat#n#1 }

Figure IV.3: Variables associated with the frame-like representation of *cat*

However, most of the problems which are naturally modelled as a CLP don't have and implicit structure. We will use a kind-of dependency representation between objects, 'flattening' our problem. 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, the object does not take part in any model).

Variable	Values
C1.POS	{ NN1 }
C1.HEAD	{ cat }
C1.SENSE	{ cat#n#1 }
C2.POS	{ VVZ }
C2.HEAD	{ eat }
C2.SENSE	{ eat#n#2 }
C2.MODEL	{ transitive }
C2.AGENT	{ c1 }
C2.PATIENT	{ c3 }
C3.POS	{ NN1 }
C3.HEAD	{ fish }
C3.SENSE	{ fish#n#2 }
C3.MODEL	{ NONE }
C3.ROLE	{ TOP }

Variable	Values
C1.POS	{ NN1 }
C1.HEAD	{ cat }
C1.SENSE	{ cat#n#1 }
C1.MODEL	{ NONE }
C1.ROLE	{ agent.transitive.c2 }
C2.POS	{ VVZ }
C2.HEAD	{ eat }
C2.SENSE	{ eat#n#2 }
C2.MODEL	{ transitive }
C3.ROLE	{ TOP }
C3.POS	{ NN1 }
C3.HEAD	{ fish }
C3.SENSE	{ fish#n#2 }
C3.MODEL	{ NONE }
C3.ROLE	{ patient.transitive.c2 }

Figure IV.4: Two different CLP formalization of *The cat eats fish*

Figure IV.4 shows two different CLP representations for the "The cat eats fish", on the left using references and on the right using two special variables *model* and *role* for each object.

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

Since a CLP always assigns a label to each variable; we will use the two null-labels defined previously: NONE for the model variables (objects which do not 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).

In the CFG example, first a “lexicon” maps the input to our initial object. For this simple task, our initial objects will be simple PoS tags. For the current example, we will use a simple lexicon (shown in figure IV.5), establishing that *cat* and *fish* could be both nouns (N) and verbs (V) and also that the only valid PoS for *eat* and *the* are verb (V) and determiner (D) respectively.

Each one of these initial objects could have different models (in this case we will associate CFG rule to PoS). For the current example, we will use a simple grammar (shown in figure IV.5), with two rules. The first one (named MNP) establishing that a Noun acting as head of this model could be combined with a determiner (D) to build a noun phrase (NP). The second one (named MS) establishing that a verb acting as head of this model could be combined with two different noun phrases to construct a sentence (S).

Lexicon		CFG Grammar		
Word	PoS	Head	Id	CFG Rule
cat	N, V	N	MNP	D, N \implies NP
eat	V	V	MS	NP ₁ , V, NP ₂ \implies S
fish	N, V			

Figure IV.5: A simple Context Free Grammar

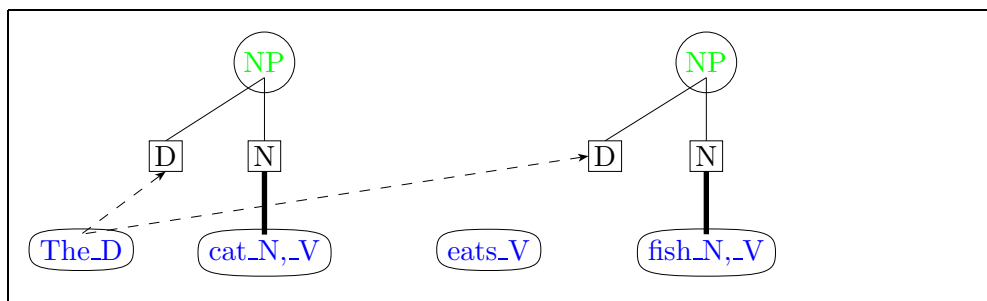


Figure IV.6: Representation of the possible instantiations of the rule $D, NP \implies NP$

Being a CFG, in order to fill a role, we will impose to the object to have the same PoS than the role. For instance, following the current example, figure IV.6 shows graphically the possible applications of the rules for the example sentence. That is, the object associated to *The* could fill the role *D* (dash line) of the model $D, N \implies$

NP anchored (thicker line) in *cat* but also the *role D* of the model anchored in *fish*. Figure IV.7 shows the final CLP representation.

Variable	Values
C1.POS	{ D }
C1.HEAD	{ The }
C1.MODEL	{ NONE }
C1.ROLE	{ TOP, D.MNP.cat, D.MNP.fish }
Variable	Values
C2.POS	{ N, V }
C2.HEAD	{ cat }
C2.MODEL	{ MNP, MS }
C2.ROLE	{ TOP, NP ₁ .MS.eat, NP ₂ .MS.eat }

Variable	Values
C3.POS	{ V }
C3.HEAD	{ eat }
C3.MODEL	{ MS }
C3.ROLE	{ TOP }
Variable	Values
C4.POS	{ N, V }
C4.HEAD	{ fish }
C4.MODEL	{ MNP, MS }
C4.ROLE	{ TOP, NP ₁ .MS.eat, NP ₂ .MS.eat, NP ₁ .MS.cat, NP ₁ .MS.cat }

Figure IV.7: CLP representation for CFG parsing of “The cat eats fish”

IV.4 Role and Model Application

In order to see whether a model can be applied or not, we should determine which combination of objects could be used to fill the model’s roles (henceforth instantiate). First we will establish which roles an object can play in isolation, that is, regardless which objects fulfil the other roles of the model. For instance if our model needs a number agreement between two roles we will initially oversee this constraint since it involves knowing which object is instantiating the other role.

Regarding a role and the possible object that could fill it, we distinguish three different kinds of attributes:

- **Compulsory:** The object attribute must match the role attribute.
- **Optional:** The object will be considered as a possible filler of the role, even though, the object attribute do not match the role attribute. The matching function will penalise it.
- **Ignore:** The object could contain information that the match function must not take into account (e.g. an attribute containing the description of the role or its name). We do not consider these attributes at all.

We define the function $match(object, role)$ to determine whether an object fills a role in isolation. That is, if an object matches all the compulsory attributes of the role without considering the objects that could fill the other roles of the model.

In order to choose between different possible fillers for a role, we need a finest function to measure how well an object fills a role, and not only whether an object can fill a role or not, as the $match(object, role)$ function does. We named this function $sim(object, role)$ and range the similarity to $[-1, 1]$, that is, from incompatible objects to full compatible objects. Although other properties are desirable (such as to be a distance), we will not make any additional restriction on the $sim(object, role)$ function.

Complex multiple slot match functions could be formalized in this framework as a sim measure. For simplicity, we will use a matching/similarity measure between a role and an object as the normalized sum of the similarity between the values of all the attributes:

$$sim(object, role) = \frac{\sum_{a \in Atts} sim(object.a, role.a)}{|Atts|}$$

For instance, in the current example of a CFG where the syntactic category is the only attribute, we allow an object to fill a role if both have the same category (Compulsory). Thus, we define the matching function as

$$match(object, role) = \begin{cases} 1 & \text{if } object.catg = role.catg \\ -1 & \text{otherwise} \end{cases}$$

In this simple example, as the category is the only attribute, the sim function will be the same than the $match$ function.

Using this matching function we obtain the set of possible role–objects instantiation shown in Table IV.1.

Role	Object
D.MNP.cat	{ The }
D.MNP.fish	{ The }
NP_1 .MS.eat	{ cat, fish }
NP_2 .MS.eat	{ cat, fish }
NP_1 .MS.cat	{ cat, fish }
NP_2 .MS.cat	{ cat, fish }

Table IV.1: Possible CLP Assignments using the $match$ function

Once the possible fillers for each role are determined, we should choose which ones could be used together to instantiate the full model. For instance, which pairs of objects hold the number agreement, that is, both objects filling the roles simultaneously have the same number. In our example, a typical CFG will force the element to be contiguous and in a determined sequential order. Thus, in the current CFG example *The* could not fill the role *D* from the *fish*'s *MNP* model.

Obviously, the correspondence between the input sentence and the models is not usually perfect. The applicability conditions of the models could vary greatly, e.g. we could relax that conditions and allow the non-contiguity of the elements, or even allow changes in their order.

Moreover, the application of a model does not only need to formalize all the possible combination of objects that can instantiate a model but also to establish which is the best instantiation among all the possibilities. This measure mostly depends on the kind of pattern matching implicit in models we are considering for a particular task. For instance, a particular instantiation of a model can be penalized according to different criteria, e.g. the number of gaps, the unordered fillers, the number of optimal roles that are not instantiated, etc.

Approximate pattern matching techniques based on edit operations (e.g. [Wang et al., 1994], [Shasha et al., 1994]) are the most commonly used to deal with inexact or error-tolerant methods. One of the main drawbacks of the tree-edit matching approaches is the difficulty to integrate them with other types of knowledge. However, [Torsello and Hancock, 2003] prove that it is possible to approximate a tree edit distance matching using a more general framework, that is, CLP. Thus, CLP will allow us to modelize different kinds of model application (pattern matching), e.g. unordered, gaps, optional roles, and also integrate it with any other processes or knowledge we can formalize as a set of constraints.

In the current example we will extend our CFG formalism to allow optional roles in a model, e.g. $D^*, N \implies NP$ will stand for allowing an optional role D .

IV.5 Model Application Constraints

We should establish a set of constraints to ensure the right application of roles and models in isolation (**model instantiation Constraints**).

In order to formalize this framework we will use the following conventions: *Objects* is the set of all possible objects, *Roles* is the set of all possible roles and *Models* is the set of all possible models. Regarding a particular model, $Roles(m)$ where $m \in Models$, stands for the set of all the roles of a particular models. Similarly, regarding a particular object, $Roles(x)$ where $x \in Objects$, stands for the set of all the models of a particular object.

Constraints are represented as follows: $[A = x] \sim^w [B = y]$ denotes a constraint stating a compatibility degree w when variable A has label x and variable B has label y . The compatibility degree w may be positive (stating compatibility) or negative (stating incompatibility). For simplicity we will also use the symbol \approx to denote incompatibility.

According to the particular nature of the models used, constraints should be added for:

- **Role Support:** Establishing how likely is a particular instantiation of a role given the current context, i.e. taking into account due the static and dynamic properties of the possible filler or how likely is its model.
- **Model Support:** Establishing how likely is a model due to the possible (or the lack of) instantiation of its roles.
- **Model Inconsistence:** Establishing when a model is inconsistent due to the possible (or the lack of) instantiation of its compulsory roles.

IV.6 Inference Engine

In the previous section, we have seen how the application of a model can be formalized as a CLP. However, a model is not only applied in isolation, the object resulting of the application of a model could also be used by other models.

For instance, in the current example, the model $NP_1, VP, NP_2 \implies S$ can only be applied if we have previously applied the models associated to *cat* and *fish* to build two *NP* objects.

Thus, a production rule system (like CFG) requires some kind of inference engine to manipulate the rules (models) and decide which ones are ready to apply. That is, which ones have a set of objects that correctly match their roles.

Some constraints must be included to ensure the right application of the models, that is, the correct identification of each element in a model, but also to ensure the correct compositional process. However, it is not an easy task to model a rule-based system inside a CLP. A CLP is mostly based on knowing in advance the search space, that is the whole set of variables and their possible values (domain).

In a production rule system, each time we apply a model a new object is generated. Then, this new object could be used for other rules to generate new objects and so on. Thus, to explicitate all the possible objects a model can use, we would have to explicitate all the possible instantiations of the models. In a general case, it is neither practical nor possible to calculate this closure.

The standard approach to implement a production rule system is to store the partial set of objects in a temporal memory and design strategies to decide which rule to apply next. These strategies try to avoid backtracking and the generation of partial results which are not present in the final solution. A similar strategy has been applied to dependency parsing [Menzel, 1998], [Schröder, 2002].

There are some other alternatives, for instance, adding new variables and values to our CLP (that kind of problem are called dynamic CSP/CLP) as new objects are generated in the working memory. Another possibility would be to restrict the architecture so that the closure of the possible models that could be applied on a particular sentence can be calculated.

We have chosen the last possibility. That is, to reduce the computational cost of this closure by restricting our architecture to: associate models only to the initial set of objects, and allow the application of only one model per object.

When several objects combine themselves using a model, a new objects representing the result should be created. We constraint these new objects to not have models, avoiding the recursive application of models. Taking into account that our main task will be Semantic Parsing, it seems more practical to adopt this kind of “lexicalized”-models, and allow models only for the objects associated to the input sentence. On the other hand, regarding the number of models per object, if only one model per object is allowed, only one of the next-level objects will be present in the final solution. Adding these two restrictions, the number of objects (partial solutions) than can be part of the final representation is $2 * n$ being n the number of initial object.

Figure IV.8 shows the closure of all possible instantiations of the models and their recursive application (derivational sequences). Each initial object (in blue) has several models associated (thicker lines). These models have several roles (boxes) which could be filled by different objects (dashed lines) in order to generate a new object (circles). These new generated object could also be applied to fill the roles of other models.

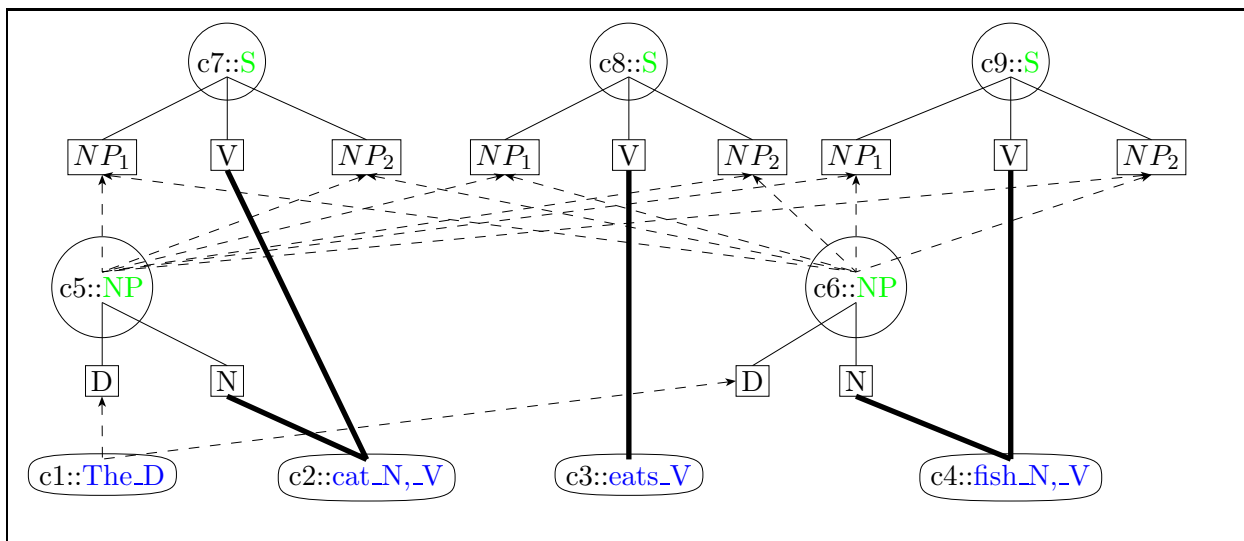


Figure IV.8: A complete scheme of all possible derivations

IV.7 Derivational Sequences

Although, we have ensured the correct application of a model in isolation, we also need to ensure the correct combination of the models. That is, we should ensure that the possible application of the models is a consistent derivational sequence. We will establish a set of constraints (**model combination constraints**) to ensure the correct combination of the models.

IV.7.1 Model Combination Constraints

The combination of models in PARDON is ruled by the following four axioms:

- **Object Instantiation Uniqueness:**

The first axiom constraints an object not to fill more than one role, otherwise we could reach a derivational structure like the one shown in figure IV.9, where the NP derived from *the cat* is instantiating simultaneously to NP_1 and NP_2 of the model anchored in *eats*.

$$[c_x.role = a] \approx [c_x.role = b] \quad \forall x \in Objects \quad \forall a, b \in Roles(x) \mid a \neq b$$

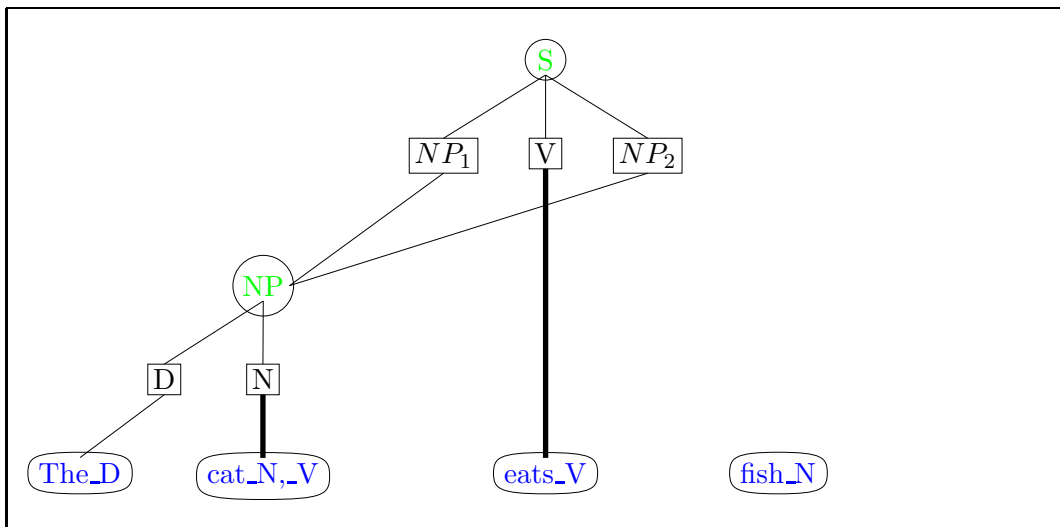


Figure IV.9: Violation of the Uniqueness Object Instantiation

- **Role Uniqueness:** Only one object can instantiate a role. This constraint avoids situations like the one shown in figure IV.10, where the object associated to *the cat* and to *fish* are filling the same role. This axiom does not mean that a semantic object can play a unique semantic role (which is not necessary true). This restriction enforces that the model should establish this co-indexing and has a unique element (role) in the model.

$$[c_x.role = a] \approx [c_y.role = a] \quad \forall x, y \in Objects \quad \forall a \in Roles \quad | \quad x \neq y$$

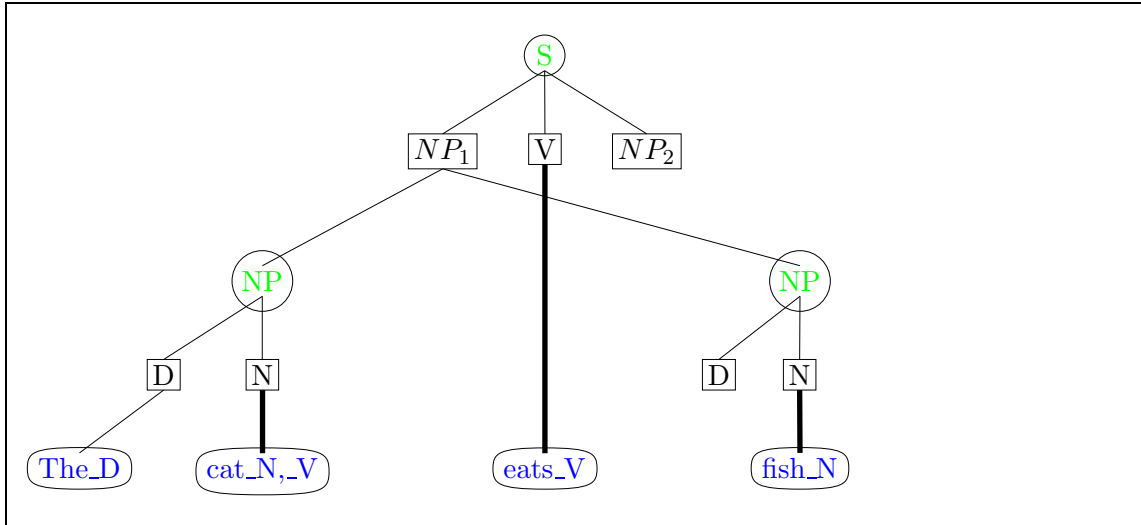


Figure IV.10: Violation of Role Uniqueness

- **Model Uniqueness:** We restrict the models associated to an object to be incompatible among them. For instance, we avoid derivational sequences such as the one shown in figure IV.11, where the object *cat* is using two of its models $NP, V, NP \implies S$ and $D, N \implies NP$ simultaneously. This constraint ensures that an object only applies one of its models. For instance, that either the object *cat* is using the model $NP, V, NP \implies S$ or $D, N \implies NP$ but not both.

$$[c_x.model = a] \approx [c_x.model = b] \quad \forall x \in Objects \quad \forall a, b \in Models(x)$$

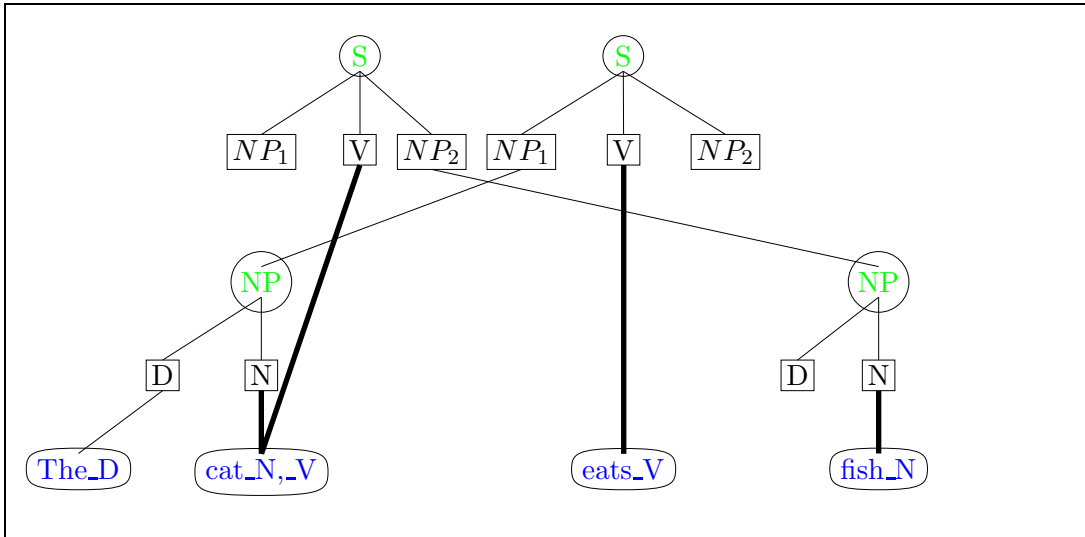


Figure IV.11: Violation of the Model Uniqueness

- Role Inconsistence:** A role can not be filled if the object which the model is anchored to is using another model. For instance, in figure IV.12 where the object *fish* is filling the role NP_2 of the model MS anchored in *cat* (dash line) while the object *cat* is applying another model (MNP) to combine itself with the object *The*.

$$\forall x, y \in Objects \ (r, x, m_a) \in Roles(y) \ m_b \in Models(x) \ | \ m_a \neq m_b \ [c_y.role = (r, x, m_a)] \approx [c_x.model = m_b]$$

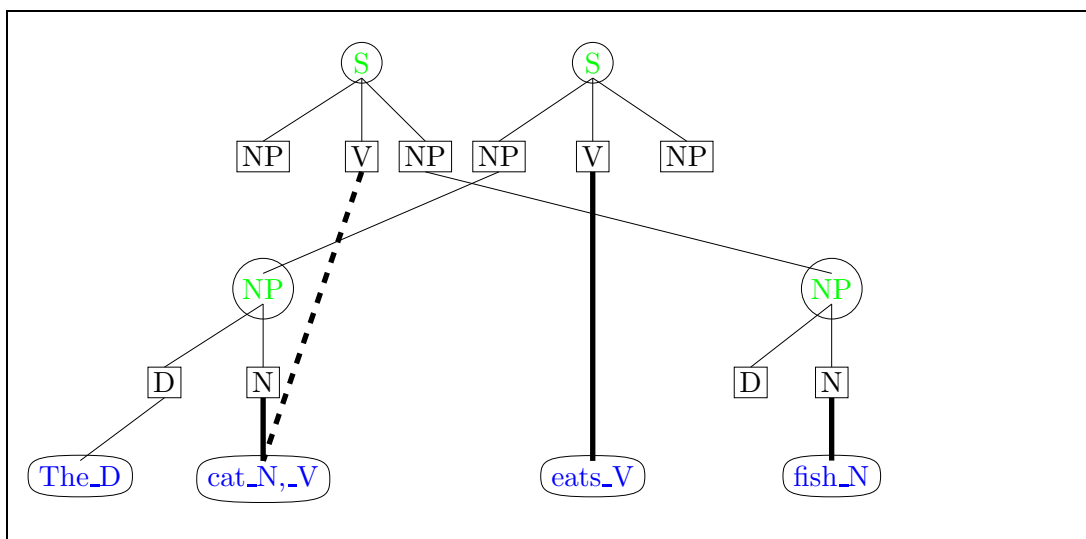


Figure IV.12: Role Inconsistence

These four axioms ensure the right combination of the models, but the architecture also needs to explicitate a measure of the goodness of the resulting combination. Thus, in order to evaluate the goodness of the application of a model, different properties can be measured, for instance, how well each object fits the role, the **Model consistence**, that is how the goodness of a particular instantiations of the roles of a model affect the good application of the model, or the **Role Support**, that is how the goodness of the whole instantiation of a model affect the goodness of the particular instantiation of a role.

IV.7.2 Amalgamating the Search Space

In real NLP applications, reducing the search space is an important issue, for instance, Stephen Beale [Beale, 1996] shows that the number of possible semantic analysis for an *average* sentence in the Mikrokosmos Machine Translation system is about 56 millions. It becomes even harder when we have to deal not only with hard constraints (staying yes or not) but also with soft Constraints (preferences or heuristics) or even with inconsistencies in our knowledge. For instance, assuming we consider all free combinations of objects taking only into account the PoS, we can obtain 196 different parse trees for our example sentence.

The proposed architecture could be formalized as a Constraint Satisfaction Problem and solved using optimization techniques (in a similar way than [Beale, 1996]). The main problems that we still need to face are the reduction of the search space and how to deal with hard Constraints (constraint that must be satisfied) and soft Constraints (preferences or heuristics).

Even with these simplifications, the amount of objects that can be generated is large, either due to the application of different models, or due to the different instantiations of the same model (the use of non-exact pattern matching techniques could multiply greatly the number of possible instantiations of a model).

Each of these possible applications of a model will create a different resulting object, which has to be taken into account when applying the models from other objects. The more we have and the looser our model application is, the more different objects could be generated. The exponential enlargement of our search space could make our approach inviable in practice.

id	Object	Gaps	Un.	Opt.
c5.1	D:(NONE) N:(cat)	N	N	Y
c5.2	D:(The) N:(cat)	N	N	N
c6.1	D:(NONE) N:(fish)	N	N	Y
c6.2	D:(The) N:(fish)	Y	N	N
c7.1	NP_1 : NONE V:eat NP_2 : NONE	N	N	Y
c7.2	NP_1 : (D:(NONE) N:(cat)) V:eat NP_2 : NONE	N	N	Y
c7.3	NP_1 : (D:(The) N:(cat)) V:eat NP_2 : NONE	N	N	Y
c7.4	NP_1 : NONE V:eat NP_1 : (D:(NONE) N:(fish))	N	N	Y
c7.5	NP_1 : NONE V:eat NP_1 : (D:(The) N:(fish))	N	N	Y
c7.6	NP_1 : (D:(NONE) N:(cat)) V:eat NP_2 : (D:(NONE) N:(fish))	N	N	Y
c7.7	NP_1 : (D:(NONE) N:(cat)) V:eat NP_2 : (D:(The) N:(fish))	Y	Y	Y
c7.8	NP_1 : (D:(The) N:(cat)) V:eat NP_2 : (D:(NONE) N:(fish))	N	N	Y
c8.1	NP_1 : NONE V:fish NP_2 : NONE	N	N	Y
c8.2	NP_1 : (D:(NONE) N:(cat)) V:fish NP_2 : NONE	N	N	Y
c8.3	NP_1 : (D:(The) N:(cat)) V:fish NP_2 : NONE	Y	N	Y

Figure IV.13: Consistent Partial Objects generated from “The cat eats fish”

For instance, for the object resulting of the application of the model MNP (that is, D^* , $N \implies NP$) for the object $c5$ (that is, cat) we can generate two different objects $c5.1$ (D:The N:cat) and $c5.2$ (D:NONE N:cat), doubling the number of possible objects that can be generated in the second level (that is the models using NPs).

Figure IV.13 shows all the correct objects that can be correctly generated from the current example, the **Gaps** column indicates whether stands the models are discontinuous, **Un.** indicates whether there are unordered elements, and **opt** whether optional roles are allowed. That is, without taking into account all of the inconsistent combinations that can be tried.

Even if we use robust pattern matching techniques, we can not expect to be always able to generate an object that covers the whole sentence. Thus, PARDON should incorporate a mechanism to combine all the generated objects plus the initial ones. Figure IV.14 shows some of the object combinations that can be tried.

Object Combinations
c1 c2 c3 c4
c1 c5.1 c3 c4
c5.2 c3 c4
...
c8.2 c3
c8.3

Figure IV.14: Some of the different possible solutions for the “The cat eats fish”

IV.7.2.1 Structural Constraints

A set of axioms is needed to ensure the correctness of the partial object combination (**structural constraints**). This is done through the NONE model and the TOP role, and allows us to obtain a solution for the whole sentence even if this solution is a combination of various objects (in a similar way to return a partial parsing instead of nothing when a full parse tree can not be obtained):

- **TOP Uniqueness:** There is only a Top. That is different assignments of the label TOP are incompatible.

$$[c_x.model = TOP] \approx [c_y.model = TOP] \forall x, y \in Objects, x \neq y$$

- **TOP Existence:** There is at least one TOP.

$$\exists x \in Objects [c_x.model = TOP]$$

- **No Cycles:** Two assignments forming a cycle are incompatible. This ensures that an object can not take part of its own model. Only direct cycles are checked.

$$[c_x.role = (r, y, m_y)] \approx [c_y.model = m_x] \forall x, y \in Objects \ m_x \in models(y) \ m_y \in models(x)$$

- **NONE Support:** The NONE model is compatible with the inexistence of any role assignment of the semantic object models.

$$[c_y.model = NONE] \sim \nexists [c_y.role = a] \forall y \in Objects$$

IV.7.2.2 The Amalgamated Representation

In order to soften this combinatorial explosion, given an initial object, we will amalgamate the representation of all the possible objects which could be generated using the models associated to the same initial object. Thus, meanwhile an object uses its models to combine itself with other objects, some of the resulting object values are determined (in a similar way of Hearst's *Polaroid Words* [Hirst, 1987]). Roughly speaking, PARDON combines objects from one level in order to build the objects corresponding to the next level of the task under consideration but the resulting object is calculated simultaneously to the task of determining which models are to be applied to find the best solution.

The formalization proposed will relax this sequential application of models by means of calculating which roles can play the generated object regardless the particular instantiation of the model's roles. That is, calculating the closure of the possible values of the slots of the object that can be generated applying the model.

It is our believe that since the object resulting from an application of a model is a function of all the objects involved, many of this possible combinations (and the different resulting objects) share the same properties. That is, we can follow on the application of models without knowing the exact application of the model (that is, for instance without taking all the PP attachments decisions). For example in the sentence, "John saw Mary on the hill with a telescope", the different resulting

semantic object covering the whole sentence for the four different possibilities of attaching “*on the hill*” or “*with a telescope*” will share the event $see(John, Mary)$. Thus, in a more complex sentence like “*Do you mean that John saw Mary on the hill with a telescope*” we can overcome the impact of these decisions and see how the common resulting object $see(john, mary)$ fits a role in another model (e.g. *person - mean - statement*). Then, as we take decision of these local attachments, we can reconsider how well this new object fits on a model.

The calculation of the closure of the object’s slots over the set of models will be easier to calculate restricting our formalism with types as in Attribute-Logic Engine (ALE)² [Carpenter, 1992]. Although it could also be roughly calculated as in the following algorithm 1:

Algorithm 1 Pseudo code of the algorithm to calculate the closure of the object’s Attributes

```

for each initial object do
   $AO_i.Att \leftarrow O_i.Att$ 
  while a rule can be applied do
    for each new instantiation of rule  $R_i$  anchored in  $O_i$  do
       $AO_i.Att \leftarrow AO_i.Att \cup applyRule(R_i).Att$ 
    end for
  end while
end for

```

In a general case, the calculation of the slot’s closure could be as computationally hard as calculating all the possible combinations of models. However, in most NLP tasks, the relevant properties of the generated object, that is, the properties used to know if this object can be used in another model, do not depend on the instantiation of the roles. Moreover, it is not necessary to calculate the exact closure, any superset of the closure can be used, although in the limit we will explore the whole search space of combinations in which a object can play any role, because nothing is known in advance about their slots.

In the current example of a CFG, restricting the closure of the all the objects the set of model can generate is not computationally hard to calculate. The amalgamated objects will have only one attribute *catg*. For instance, in the current example, *cat* has only two models and the only relevant slot to fill other roles is *catg*. Regarding the slot *catg*, all the possible objects that can be generated can be either *NP* or *S*. In order to calculate the closure it has also to be taken into account that an object may not use any of their models, thus the closure regarding the slot *catg* must include the initial values; which will result in $\{ N, V, NP, S \}$.

²<http://www.cs.toronto.edu/gpenn/ale.html>

The figure IV.15 shows the CLP after applying the algorithm described above to calculate the closure over the models associated to the initial objects.

Variable	Values
c1.CATG	{ D }
c1.ROLE	{ TOP, D.MNP.c2, D.MNP.c4 }
c1.MODEL	{ NONE }
c2.CATG	{ N, V, NP, S }
c2.ROLE	{ TOP, NP ₁ .MS.c2, NP ₂ .MS.c2, NP ₁ .MS.c3, NP ₂ .MS.c3 }
c2.MODEL	{ NONE, MNP, MS }
c3.CATG	{ V, S }
c3.ROLE	{ TOP }
c3.MODEL	{ NONE, MNP, MS }
c4.CATG	{ N, V, NP, S }
c4.ROLE	{ TOP, NP ₁ .MS.c2, NP ₂ .MS.c2, NP ₁ .MS.c3, NP ₂ .MS.c3 }
c4.MODEL	{ NONE, MNP, MS }

Figure IV.15: CLP representation

Once we have restricted which roles could play any of the possible objects a model can generate (regardless the particular combination of object that instantiates the roles), a set of constraint should be added to ensure that the set of selected values is consistent with the application and instantiation of the models.

For example, the following constraint will assure that if the model selected for $c1$ is *MNLP* the attribute *catg* will be *NP* $[c1.model = MNP] \sim^+ [c1.catg = NP]$. Similar constraints will be added for the rest of the models. In that example, we will need another constraint supporting the selection of the *catg* *S* in case the model *MS* is selected for $c1$: $[c1.model = MNS] \sim^+ [c1.catg = NS]$

IV.7.2.3 Attribute Propagation/Percolation

The closure in a CFG, although depends on the complexity of the grammar, is easy to calculate because all the resulting objects share the relevant slot (the syntactic category). However, having more complex models, a relevant slot could depend on the object that fills a role. This not only increases the complexity of the closure but also the CLP formalization as we have to establish constraints to propagate these attributes.

More complex models, as the model in figure IV.16, which includes a re-entrancy in the attribute *Sem*, need a more complex representation inside our CLP formulation.

We can distinguish two types of attributes in the object resulting from the application of a model, *Static attributes* which are known before applying the model, such as the attribute *catg* in the model figure IV.16, and *Dynamic attributes* which can only be calculated when all of at least some of the roles that instantiated the model

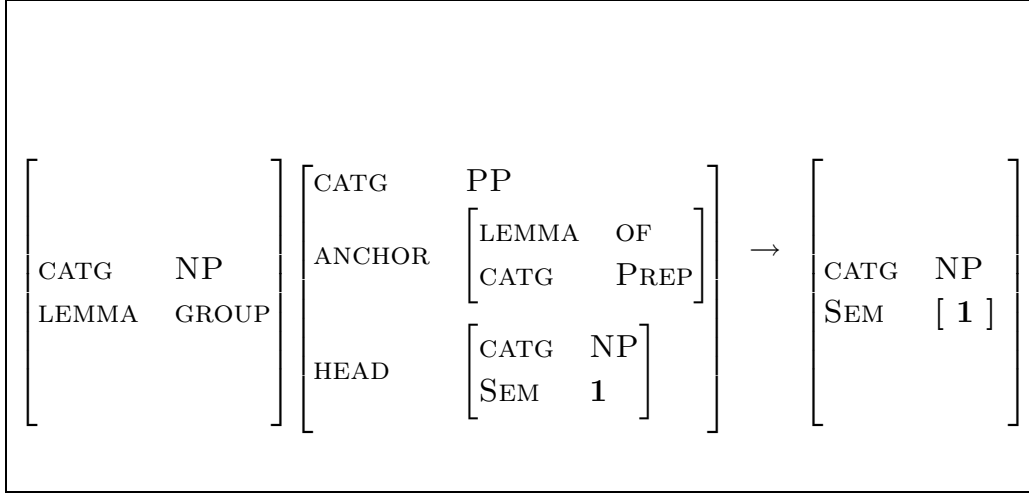


Figure IV.16: A Complex Model with Propagation of Attributes

are known, as the attribute *Sem* in figure IV.16 which can only be determined when the slot *Sem* of the role HEAD filler is known. The reentrancy implied in *Dynamic attributes* needs additional mechanisms to be added to our CLP formulation.

These dynamic attributes have to be consistently propagated through the compositional process in order to update how similar is the current common representation of exclusive objects and a role (that is, $sim(obj, role)$).

The mechanism to ensure this consistency consists of adding variables to represent the dynamic attributes (Obj.attribute) and restrictions to *propagate* the labels consistently through the instantiation of the model. Bellow we show the constraints which propagate a dynamic attribute f defined in a model m of an object O as $[f = X.f]$.

- **attribute from a complex object**

$$[O.f = l] \sim^+ \exists [X.role = (r, O, m)] \wedge [X.f = l] \quad \forall l \in X.f,$$

e.g:

$$[group.sem = Animal] \sim^+ [cat.role = (head, groupof, group)] \wedge [cat.sem = Animal]$$

- **attribute from the basic object:**

$$[O.f = l] \sim^+ \exists [O.model = NONE] \vee [BO_A.f = l] \quad \forall l \in A.f,$$

e.g:

$$[Group.sem = Group] \sim^+ [Group.model = NONE] \vee [BO_{Group}.sem = Group]$$

IV.7.2.4 Attribute Representation

Each attribute in the semantic object could be represented as a variable in the CLP. However, most of the attributes have a unique value or do not have any constraint which could select a value for this attribute among their possible values. That is, most of the attributes/variables that determine how well an object will fit a role will not change (that is, they are *Static*). Those attributes of the amalgamate representation of an Object which can change or select a value conform the *Dynamic* part. Then, the object attributes can be split into *static* and *dynamic*:

$$O.Att \triangleq O.StaticAtt \cup O.DynAtt$$

Thus, also a great part of the constraints about how likely is that an object plays a role can be calculated only once, at the beginning of the process.

For instance, to calculate how likely is that the object *c2* (cat) play the role *NP₁* of the model *MS* for the object *c3* (eat), $sim(c2, NP_1.MS.c3)$ will return how suitable are the amalgamated objects (that is considering all the possible values for each attribute of the amalgamated objects).

The function *sim* measures the similarity between an Object and a Role. *sim* could be also split in two, a static sim_{static} and a dynamic part sim_{dyn} :

$$sim(R, O) \triangleq sim_{dyn}(O.DynAtt) \otimes sim_{static}(O.StaticAtt)$$

Since *O.StaticAtt* does not change, the function $sim_{static}(O.StaticAtt)$ could be calculate only once (e.g. when the CLP is built).

In a similar way in the amalgamated object representation we can split the similarity measure in two parts. The one involving only static attributes, that do not change, (sim_{static}) and the dynamic part (sim_{dyn}) which has to take into account the current state of the amalgamate objects (represented as the weight of the different assignments in the CLP). In order to calculate this dynamic part we can establish a set of constraints which establishes the similarity between the current state of the amalgamate object and the role:

$$\begin{aligned} [AO_i = R] &\sim^{sim_{static}(AO,R)} \\ [AO_i = R] &\sim^{sim_{dyn}(AO,R)} [AO_i.Att_1 = v_1] \dots [AO_i.Att_n = v_1] \\ \dots & \\ [AO_i = R] &\sim^{sim_{dyn}(AO,R)} [AO_i.Att_1 = v_k] \dots [AO_i.Att_n = v_m] \end{aligned}$$

For instance, in the current example, the constraint:

$$[cat.role = NP_1.MS.Eat] \sim^{+sim(\dots)} [cat.Sem = Animal]$$

will ensure that the similarity measure takes into account whether we are selecting the Animal sense though the *Group of Cats* or not.

Only *Dynamic* attributes need to be represented in the CLP, as all the functions regarding the *Static* ones could be transformed into constant expressions in order to optimize the calculations.

IV.8 Formalization as a CLP

As seen in the previous subsections, PARDON's framework can be formalized as a CLP. Once a NLP task is modelled as a CLP using PARDON, it can be solved using well known optimization methods (e.g. the relaxation labeling algorithm) to find the most consistent solution. A CLP with weighted constraints does not distinguish between *hard* and *soft* constraints. Some hard-constraints are implicit in the formalization and thus can not be violated (e.g. role unicity), some part of the *hard* constraints can be applied during the CLP formalization to filter out labels (e.g. matching constraints between the role and objects), and the remaining are relaxed to *soft* constraints giving to them an arbitrary large (infinite) weight to force the system to satisfy them on convergence.

However, as we will see later on the experiments, using a relaxation labeling technique, if the final state do not hold all these constraints (because we have converged to a local maxima or there is no state which could satisfy all the constraints) a mixed partial or multiple models could be combined in the solution.

We are not concerned about the well formedness of the input and the models. We are dealing with a communication event, where there is no doubt that, even when the utterance is not well formed and contains any kind of error, there is an intended meaning. Thus, from our point of view, robustness in NLP means to find always the more reliable solution, even if the input is not well formed or the models are incomplete or inconsistent. Thus, we allow to violate hard constraints if there is no other way to find a possible solution, codifying them as soft constraints with a high weight.

On CLP the different assignments of a variable are incompatible. Thus, using the formalization proposed in this chapter the *Object Instantiation Uniqueness* and *Model Uniqueness* constraints are ensured by the algorithm itself (as the labels of a variable are incompatible among them).

The next step in the formalization of PARDON as a CLP is to establish the possible assignments, that is, to determine which the possible models are, and which roles of these models can be played by the initial objects. Thus, we have to determine which of the restrictions expressed by the model must hold (*Hard Constraints*) and which constraints can be softened in order to find a solution (*Soft Constraints*), (e.g. selectional preferences, heuristics or knowledge that we know could be inconsistent or incomplete).

When formalizing the problem, the models that can not fill any of their compulsory roles should not be taken into account and neither should all their associated roles (and their possible assignments). The *hard-constraints* involved in the identification of roles are applied in the function **match**. Using this function, we would determine whether an object could play a role (represented as a possible assignment) or not.

The algorithm 2 describes in pseudo-code the general procedure for building the CLP once the initial objects are created:

Algorithm 2 Algorithm for determining the set of possible roles for an object

```

for each model  $M$  associated to a initial object do
  add  $\langle M, A \rangle$  to the activeModels list
end for
for each  $\langle M, A \rangle$  in the activeModels do
  if all the compulsory roles of  $M$  have at least one match then
    for each role  $R$  of model  $M$  do
      for each object  $SO$  do
        if  $\text{match}(SO, R)$  then
          add  $SO$  as possible player of role  $R$ 
        end if
      end for
    end for
  end if
end for

```

One of the main advantages of a CLP is that it can be enriched with arbitrary sets of task-specific constraints (e.g. statistical information, selectional preferences) which enforce the application of the models or assure other preferences or desirable characteristics of the solution (e.g. non crossing of syntactic dependencies).

Once the CLP is build, The relaxation labeling algorithm can be applied to find a local maxima that satisfies all the constraints to a maximum degree.

IV.9 Conclusions

In this chapter we have presented the PARDON's architecture, which similarly to a rule-based systems is based on the idea of *compositionally*. An element combines itself with other elements to build a new element. We have formalized as a CLP the three major components of PARDON's Architecture, that is the **Knowledge Representation**, the **Model Application** and the **Inference Engine**.

In order to demonstrate the flexibility and robustness of this novel architecture, chapters 5 and 6 will apply the PARDON architecture to two different tasks, Semantic Parsing and Word Sense Disambiguation respectively, adapting the formalization to the concrete task.

CHAPTER V.

PARDON Semantic Role Labeler

“We are all full of weakness and errors; let us mutually pardon each other our follies it is the first law of nature.”

Voltaire

This chapter presents the application of the general PARDON’s architecture detailed in the previous chapter to a particular NLP tasks, Semantic Role Labeling (SRL). We describe how PARDON can be applied to SRL task and the concrete knowledge sources used to solve this task.

The aim of this chapter is also to show the development of a robust (able to work on unrestricted text) and flexible (portable and extensible) approach to SRL. We will do so by means of the application of the PARDON’s architecture. That is, formalizing SRL as a Consistent Labeling Problem (CLP).

As shown in chapter II, SRL consist in the production of a case-role analysis in which the semantic roles –such as *agent* or *instrument*– played by each entity are identified [Brill and Mooney, 1997]. This task, is crucial in any application which involves some level of Semantic interpretation or Natural Language Understanding.

SRL is a particular interesting task to apply the PARDON architecture because we will need to focus on the interaction between syntax and semantics, as well as on verbs, as the head sentence components.

Recent works on computational linguistics has attempted to construct representations which integrate both syntactic and semantic information about a word, e.g. Head-Driven Phrase Structure Grammars (HPSG), Lexical Functional Grammars (LFG).

Regarding the interaction of syntax and semantics, a crucial issue is the required level of syntactic analysis. As shown in chapter II, chunk parsing [Abney, 1991] has been widely used in several fields (e.g. Information Extraction) as an alternative to deal with the lack of robustness presented by traditional full parsing approaches. Chunk parsing softens problems with close world assumption (full coverage grammar and lexicon), local errors produced by global parsing considerations [Grishman, 1995]

and the selection of the best full parse tree among a forest of possible candidates.

Given that the verb is the core component of a sentence, there is no doubt that subcategorization information may not only improve parsing —e.g. taking into account probabilistic subcategorization on a statistical parser [Carroll et al., 1998]— but also provide the basic information to assemble those chunks into more complex structures.

Despite the lack of robustness of full parsing (specially for a free word-order language like Spanish), it provides useful information for the identification of roles that a simple chunk analysis is unable to capture [Gildea and Palmer, 2002]. PARDON’s architecture integrates the subcategorization information used by statistical full parsers with the information used to identify roles, thus, using this knowledge simultaneously in a collaborative and integrated manner.

V.1 *Different Approaches to Semantic Interpretation*

Semantic Interpretation and in particular Semantic Role Labeling (SRL) have been an old challenge in NLP. In this section, we will mainly focus on two systems with integrated architectures: ABSITY and *Hunter-Gatherer*. Emphasizing their processes and knowledge integration. We will also present other approaches (Fernando Gomez’s system and the Compansion project) and the recent works from Machine Learning community on SRL.

V.1.1 *ABSIITY*

Back in 1987, Hirst developed ABSITY (A Better Semantic Interpreter Than Yours) [Hirst, 1987]. ABSITY was a *tandem* semantic interpreter which combines different kinds of knowledge in a sequential order. ABSITY is based on two components: Marker Passing and Polaroid Words (PW). Marker Passing is one of the earliest techniques to find relations between components based on paths on a semantic nets. Polaroid Words (PW) are fake semantic objects, a-kind-of ambiguous representation of all the possible semantic objects associated to a word. A PW like Polaroid photograph, is a partly developed picture, but viewable and usable in its underspecified/degraded/amalgamated form. As the interpretation takes place PWs become more developed/disambiguated. ABSITY was pioneer in combining several knowledge sources. However, ABSITY applies its different knowledge in a pre-established order. Alternative knowledge is just taken into account when the previous knowledge is unable to decide.

This could drive ABSITY into an early pruning of the right interpretation. As Hirst himself pointed on [Hirst, 1987] in the example in figure V.1, where PW choose the wrong sense for *start* (Astronomical object). That is because PW disambiguates *star* based on the simple relation between *astronomer* and *astronomical_object* without taken any other consideration (such as selectional preferences for the verb marry). In Hirst's words "*The error will only be discovered after the sentence is fully interpreted and the Frail¹ attempts to evaluate the erroneous frame statement that was build*".

<i>The astronomer</i>	<i>married</i>	<i>the star</i>
Human	event	Human
		Astronomical_object

Figure V.1: Example of what PW can not do.

V.1.2 Hunter-Gatherer

On middle 90', *Hunter-Gatherer* (HG) [Beale and Nirenburg, 1995; Beale, 1996; Beale et al., 1996] was developed inside the Mikrokosmos project. Mikrokosmos² is a knowledge-based machine translation system. Mikrokosmos semantic representation is based on Text Meaning Representation (TMR) and its different knowledge sources are keep independent in different "*microtheories*".

Regarding process integration, *Hunter-Gatherer* is also a tandem³ semantic interpreter which combines different knowledge sources inside a Constraint Satisfaction Problem frame. Thus, the knowledge is not applied on a pre-established order. In order to avoid the combinatorial explosion of possibles partial interpretations, *Hunter-Gatherer* uses the parse tree to divide the problem into a-kind-of pseudo-independent sub-problems (named circles). Thus, further than efficiency criteria, the main drawback of *HG* approach is its reliance on having the correct full parse tree of the sentence.

V.1.3 Fernando Gomez's Semantic Parser

More recently Fernando Gomez⁴ has developed an algorithm for semantic interpretation [Gomez, 2001] based on extending WordNet with predicates [Gomez, 1998]. His work is centred in the determination of the meaning of the verb. WordNet does no classifies the verbs based on semantic decomposition (see for instance [Zickus, 1994] for a detailed comparison between WordNet senses and Levin Classes for some

¹Frail is the ABSITY's frame language which incorporates first-order predicate calculus

²See <http://crl.nmsu.edu/Research/Projects/mikro/index.html>

³Even though in HG, parsing is carried out alone before any semantic validation takes place in a sequential model, it can be considered as a tandem semantic interpreter in the sense that all the possible parse trees in the resulting forest are checked semantically. Thus, the overall results obtained should be the same.

⁴<http://www.cs.ucf.edu/~gomez>

verbs). Although, the WordNet troponym relation covers a diverse class of semantic relations between verbs, including the intention of the agent, the way the action is carried out, the instrument, etc. Thus, they have taken a top-down approach that defines generic abstract predicates subsuming semantically and syntactically a large class of verbs.

The abstract semantic predicates (see figure V.2)⁵ contain selectional preferences/restrictions (e.g. *substance*) and syntactic relations (e.g. *obj* with a preposition *with*) for the semantic roles defined (e.g. *theme*).

```
(fill-or-load
  (is-a (cause-to-change-of-location))
  (wn-map (fill1) (fill2))
  (theme
    (substance physical-thing)
    (obj (prep with))
  )
  (goal
    (instrumentally physical-thing)
    (obj obj-if-with (prep into on onto in))
  )
)
```

Figure V.2: Example of Fernando Gomez's Semantic Predicates

The set of semantic roles used is not defined independently of the meaning of the verb. Thus, it differs greatly from Dowty [Dowty, 1991] because it makes no distinction between adjuncts or thematic roles (e.g. he establishes a *distance* role for *change-of-location* verbs).

These entries in the predicates will be used by the semantic interpreter to attach modifiers and to link syntactic relations to semantic ones. The input of the semantic interpreter is the result of a parsing process which recognizes clausal, NP complements and relative clauses, but do not solve structural ambiguity. Then the algorithm for the semantic interpretation, roughly speaking, looks up the predicates corresponding to verbs (or nominalizations) in the sentence, establishes the possible matches for filling the predicates and applies in a fixed order a set of heuristic rules for disambiguation, PP attachment, detection of clause boundaries [Gomez et al., 1997], [Gomez, 2001]. Although two corpus of fully tagged sentences of about 1,000 sentences has been released, the whole system and the predicates associated to WordNet are not yet available.

⁵I want to thank Fernando Gomez for the example of his system

V.1.4 *Compansion*

A different approach to Semantic Interpretation for NLU is the *Compansion Project*⁶. The goal of *Compansion* is to improve the communication of people with severe disabilities via natural language processing techniques. It expands a compressed (telegraphic) sequence of words input by the user into a semantically and syntactically well-formed utterance. The input is a sequence of roots of the content words of the desired utterance; thus, many function words including determiners (e.g., *the*, *a*) and prepositions (e.g., *of*, *in*) will normally be left out. The system is responsible for filling in missing words as well as correctly conjugating the verb and forming a syntactically correct utterance. The system attempts to form an utterance whose word order most closely reflects the word order given in the original input string. For example, if the system receives “*Apple eat John*”, we would like the system to produce the sentence, “*The apple is eaten by John*”. Although with different objectives in mind, the *Compansion* projects also seeks robustness for Semantic Parsing, and can be seen as another approach for the improvement of the Natural Language Understanding components.

V.1.5 *Machine Learning approaches to SRL*

More recently, the development of resources such as FrameNet, PropBank has lead some works on the task of automatic labelling of thematic roles, using statistical and machine learning techniques [Gildea and Jurafsky, 2000], [Gildea and Jurafsky, 2002] and Combinatory Categorical Grammars [Gildea and Hockenmaier, 2003], etc.

The great interest in the community has draw to new tasks in the main evaluation competitions on NLP (CoNLL and Senseval). For instance, the Conference on Computational Natural Language Learning (CoNLL-2004 and CoNLL-2005) included semantic role labeling as a shared task. The test data consist of the sections of the Wall Street Journal part of the Penn Treebank used in past editions of the CoNLL shared tasks. The role labeling information have been derived from the Penn TreeBank II project for the syntactic information, and from the PropBank project for the propositional analysis. Also a new SENSEVAL-III task has been set up about the *Automatic Labeling of Semantic Roles* based on a FrameNet approach. This task consists in, given a sentence, a target word and its frame, identify the frame elements within that sentence and tag them with the appropriate frame element name.

Following this brief introduction, sections V.2, V.3 and V.4 explain, respectively, the basic ideas behind our system, its architecture, as well as the different sources of knowledge and the way in which they are integrated. Section V.5 describes the experiments carried out and reports the results obtained. Finally, section V.6 draws some conclusions and outlines further research lines in the application of the PARDON architecture to Semantic Parsing.

⁶<http://www.ase1.udel.edu/nli/nlp/compansion.html>

V.2 Applying the PARDON's approach

Most of the current linguistic theories assume that the syntactic structure of a sentence depends to a large extent on the lexical properties of the verbs. The meaning of the verb becomes, then, a central element whose argument structure determines the overall syntactic shape of the clause.

Our view of semantic parsing is based on compositional semantics and lexicalized models (i.e. the meaning of a sentence is the result of combining the meaning of its words and the possible combinations are determined by models associated to these words).

Bearing that in mind, using PARDON architecture, the semantic objects associated to syntactic chunks that appear in a sentence are combined in order to build a case-role representation of the whole sentence. This combination is carried out using syntactic and semantic knowledge obtained from a linguistic approach (subcategorization frames). In order not only to complement the modelization of the task but also to show the flexibility of the architecture, we enriched the system with a statistical model of lexical attraction.

For instance, starting with the chunks in the sentence “*Este año en el congreso del partido se habló de las pensiones*”⁷ shown in Figure V.3, we will obtain the case-role representation shown in Figure V.4 by combining:

- The initial semantic objects associated to those chunks.
- The impersonal model of the verb “*hablar*” (to talk) shown in Table V.2.
- The noun modifier model shown in Table V.3.

Este año	en el congreso	del partido	se	habló	de las pensiones
[C1	[C2	[C3	[C4	[C5	[C6
HEAD AÑO	HEAD CONGRESO	HEAD PARTIDO	HEAD SE	HEAD HABLAR	HEAD PENSIÓN
HDLE	HDLE EN	HDLE DE	HDLE	HDLE HABLAR	HDLE DE
CATG NP	CATG PP	CATG PP	CATG SE	CATG VP	CATG PP
NUM SG	NUM SG	NUM SG	NUM	NUM SG	NUM PL
GEN M	GEN M	GEN M	PER	PER 3	PER 3
SEM TIME	SEM TOP	SEM GROUP	SEM TOP	SEM COMMUN.	SEM TOP

Figure V.3: Chunks for “*Este año en el congreso del partido se habló de las pensiones*”

Applying the PARDON's architecture, decisions related to high level syntax and semantics will be fully integrated. Moreover, we will be able to easily integrate two completely different kinds of knowledge, the subcategorization model from LEXPIR (manually developed) and a lexical attraction model (statistical).

As the input of the PARDON Semantic Parser are chunks, the first step is to preprocess the sentences, performing PoS-tagging, chunking and semantic annotation. Then, the LEXPIR model and lexical attraction model for the elements in the sentence will be retrieved and formalized as a Consistent Labeling Problem in

⁷Literal Translation: *This year in the meeting of the political party [someone] talked about the pensions*

One of our main goals in this chapter are to explore and exploit the relations between syntax and semantics. Therefore, we need a model that makes this mapping explicit because we focus on free-word order languages, such as Spanish or Catalan, which are our big challenge. However, at present, the availability of this type of resources for languages other than English is poor. Dorr's LCS has just become recently available for Spanish and Chinese. Also, a Spanish version of FrameNet is currently under early development, although, FrameNet is not directly suitable for our purposes because there is no explicit modelling of the syntactic realization of the frame elements. This relation is being made explicit in the annotation of the FrameNet corpus to allow the automatic learning of these mapping/models by the application of Machine Learning/Statistical techniques (See [Gildea and Jurafsky, 2000; Gildea and Jurafsky, 2002]).

We chose LEXPIR ([Fernández and Martí, 1996], [Fernández et al., 1999] and [Morante et al., 1998]) for our experiments, because it provides an explicit mapping between syntax and semantics and it focuses on a free-word order language (Spanish). But our approximation to semantic parsing could be applied to other lexicons by modelling appropriately the constraints associated with their models.

Next subsection gives a detailed description of the LEXPIR lexicon and its components. This description is needed in order to better understand the way we have adapted the PARDON architecture to Semantic Parsing.

V.3.1 *LEXPIR*

PIRAPIDES [Vázquez et al., 2000] is a project centred on the study of the English, Spanish and Catalan verbal predicates. PIRAPIDES has several goals: On the one hand, from a theoretical point of view, a deep study is being carried out on the units that the model of a verbal entry has produced. This syntactic component focuses on the representation of the interaction between the syntactic and semantic components. On the other hand, from an application-oriented point of view, a lexicon (LEXPIR) based on this theoretical model is being developed, in order to perform corpus analysis.

Capturing the argumental structure or even the syntactic functions of a Spanish sentence may be a hard task, given the optionality of some constituents (such as the subject) and the free-order syntax structure of Spanish.

The classification is based on verb meaning components as well as their diathetic alternations [Vázquez et al., 2000; Fernández et al., 1999; Morante et al., 1998]. The diathetic information allows to infer the number of semantic components which can be explicitly realized or implicitly understood. For the work presented here, information about the prepositional value of arguments is also included⁹.

Verb classes are organized in a hierarchy which enables the use of default monotonic inheritance to describe verb properties. That is, each verb inherits the elements from its group and each group from its class. However, the inherited information can be overwritten by the information already associated to the specific verb entry.

⁹The information of LEXPIR has been augmented with prepositions for 61 trajectory verbs.

<i>basic model for Trajectory verbs</i>					
Catg.	Handle	Comp.	Sem.	Agree.	Opt.
NP	p_inic	starter	Human	yes	yes
x	x	entity	Top	no	yes
PP	p_ruta	route	Top	no	yes
PP	p_orig	source	Top	no	yes
PP	x	destination	Top	no	yes

Table V.1: Basic Model for trajectory verbs

Table V.1 presents the different elements that appear on the *basic* model for any *Trajectory* class verb:

- *Catg*: Syntactic realization of the semantic component. For the second component this information is unspecified (x) as the syntactic realization depends on the subclass. Moreover, this element, which is usually the Direct Object, has other restrictions: if its semantics indicates that it is [+human/animate] it should be a PP, while if it is [-human/animate] it has to be realized as an NP.
- *Handle*: List of prepositions that may introduce the component according to their meanings and occurrences.
- *Component*: Describes the meaning component, determined by the class.
- *Semantics*: Describes the semantics of the component. This is an argument-specific feature.
- *Agreement*: States whether person and number agreement with the verb is required.
- *Optionality*: States whether the component is optional.

Dealing with the optionality of the meaning components within the model itself allows us to reduce the number of possible alternations which have been established at a theoretical level (PIRAPIDES takes the underspecification of a component as an alternation). Only information which is different to the one associated to the class is actually marked. For instance, in the **periphrastic passive** model associated to the communication class, the entity element (defined as {**NP**;entity;x;Top:yes;no}), has to be realized as an NP and also has to agree with the verb.

Table V.2 shows the resulting expansion of the *basic* and *impersonal* models for a concrete verb: “*hablar*” (*to talk*) which is an instance of the *Trajectory* class. Unspecified entries are inherited from the class model definition. The upper part of the table presents the basic model inherited from the *Trajectory* class in Table V.1 but using the specific preposition of the verb “*hablar*”. The lower part shows an alternative model for impersonal uses (i.e. ‘se habla’ – people talk –) which is not present either in the general class.

<i>basic model for “hablar”</i>					
Catg.	Handle	Comp.	Sem.	Agree.	Opt.
NP	x	starter	Human	yes	yes
PP	de, sobre	entity	Top	no	yes
PP	con	destination	Top	no	yes

<i>impersonal model for “hablar”</i>					
Catg.	Handle	Comp.	Sem.	Agree.	Opt.
SE	x	se	Top	no	no
PP	de, sobre	entity	Top	no	yes
PP	con	destination	Top	no	yes

Table V.2: Models for the verb “*hablar*”

V.4 PARDON’s *Formalization for Semantic Role Labeling*

This section formalizes the PARDON approach to Semantic Role Labeling by setting up the case-role interpretation problem as a *Consistent Labeling Problem* (CLP), where the different kinds of knowledge are applied as weighted constraints.

A CLP basically consists in finding the most consistent label assignment for a set of variables, given a set of constraints. Once the sentence and its knowledge is represented in terms of a CLP, a relaxation labeling algorithm is used to obtain the most consistent interpretation¹⁰.

As we shown previously, this formulation allows us to naturally integrate different kinds of knowledge coming from different sources (linguistic and statistical), which may be partial, partially incorrect or even inconsistent.

¹⁰See chapter I and appendix A for a detailed description of CLP and the relaxation labeling algorithm.

V.4.1 Knowledge Representation

	Variable Name	Values
este año	c1.model	N_{de} NONE
	c1.role	(<i>starter, basic, c5</i>) TOP
en el congreso	c2.model	N_{de} NONE
	c2.role	TOP
del partido	c3.model	N_{de} NONE
	c3.role	(<i>entity, basic, c5</i>) (<i>entity, impersonal, c5</i>) (<i>modif, N_{de}, c1</i>) (<i>modif, N_{de}, c2</i>) (<i>modif, N_{de}, c3</i>) TOP
se	c4.model	NONE
	c4.role	(<i>se, impersonal, c5</i>) TOP
habló	c5.model	basic impersonal NONE
	c5.role	TOP
de las pensiones	c6.model	N_{de} NONE
	c6.role	(<i>entity, basic, c5</i>) (<i>entity, impersonal, c5</i>) (<i>modif, N_{de}, c1</i>) (<i>modif, N_{de}, c2</i>) (<i>modif, N_{de}, c3</i>) TOP

Figure V.5: CLP associated to the objects in Figure V.3

As described in chapter IV, PARDON represents the meaning of a sentence in terms of relationships between semantic objects, using two variables for each semantic object: the *model* (*object.model*) and *role* (*object.role*) variables. For instance, the semantic object associated to a chunk headed by “hablar” (to talk) can use a *basic* model (someone talks about something with someone: [*c5.model* = *basic*]) or

an *impersonal* model (*one* talks about something [*c5.model = impersonal*]).

The *role* variable represents the role that a semantic object plays inside the model of another semantic object. For instance, the semantic object “pensiones” (the pensions) can play the role *entity* for both models of “hablar” (to talk) (e.g. [*c6.role = (entity, basic, c5)*]).

In order to identify a role from a model label we need a triplet (*role, model, semantic object*). For instance, the role *starter* of the *basic* model for “hablar” is represented as (*starter, basic, c5*).

In our current model, the features of the initial Semantic Objects associated to the chunks do not change¹¹. Thus, only the *role* and *model* variables are needed in the CLP. Figure V.5 shows the variables and labels associated to the semantic objects in Figure V.3. As described in chapter IV, there are two *null* labels: NONE and TOP.

On one hand, the label NONE is associated to the model variables representing semantic objects that do not have/use a model (usually the semantic objects with no sub-constituents), for instance most of the semantic objects associated to Prepositional Phrases.

On the other hand, the label TOP is associated to the role variables to represent semantic objects not playing a role in the model of a higher constituent. In this application of PARDON to SRL, it usually represents the sentence head, which will probably be the main verb of the sentence.

The next step in the formalization of the SRL following PARDON’s architecture is to establish the possible assignments. That is, to determine which the possible models are and which roles of these models can be played by the Semantic Objects associated to the initial chunks.

Thus, we have to determine which restrictions expressed by the lexicalized model must hold (*Hard Constraints*) and which constraints can be softened in order to find a solution (*Soft Constraints*), (e.g. selectional Preferences, heuristics or knowledge that we know could be inconsistent or incomplete).

When formalizing the problem, the models that can not fill any of their compulsory roles should not be taken into account and neither should all their associated roles (and their possible assignments). The function *match(Object, Role)* we would determine whether a Semantic Object could play a role (represented as a possible assignment) or not.

¹¹Except role and model

Algorithm 3 describes in pseudo-code the procedure for building the CLP:

Algorithm 3 Pseudo code of the algorithm for building the CLP

```

for each chunk  $C$  in the sentence do
  create the Semantic Object  $A$  for the chunk  $C$ 
  for each model  $M$  associated to the head of chunk  $C$  do
    add  $\langle M, A \rangle$  to the activeModels list
  end for
end for
for  $\langle M, A \rangle$  in the activeModels do
  if all the compulsory roles of  $M$  have at least one match then
    for each role  $R$  of model  $M$  do
      for each Semantic Object  $SO$  do
        if match Role  $SO$  then add  $SO$  as possible player of role  $R$ 
        end if
      end for
    end for
  end if
end for

```

V.4.1.1 Attribute Representation

The constraints that establish how likely it is that a Semantic Object plays a role can be calculated only once, at the beginning of the process. This is possible because in our current model, the features of the initial Semantic Object associated to the chunks do not change¹².

V.4.2 Role and Model Application

In order to apply the $sim(Object, Role)$ measure, we established a particular similarity measure for each feature. The value returned is normalized into $[-1, 1]$. For the compulsory attributes we use the strict equality while for the optional attributes, this measure is inversely proportional to the number of relabeling operations needed to transform one feature structure into the other. Currently, only person, number semantics are considered.

As seen in the general description of PARDON's architecture, regarding the matching between an object and a role, there are three different types of attributes, **compulsory**, **optional** and those attributes to be **ignored**. In this particular formalization, we will consider **compulsory** CATG and HANDLE attributes while we will consider **optional** the SEM attribute.

As CATG, HANDLE, PERSON and NUMBER are *static*, the **hard constraints** associated to these attributes in the $match(Object, Role)$ can be calculated outside the CLP. In a similar way the agreement constraint (agreement between the verb

¹²Except role and model

and one of more roles) can be also calculated as the verb of the model is known and thus it can be calculated in the static function *sim_{static}*. Thus, in this formalization we will not add any constraint for **attribute propagation/percolation**.

After formalizing Semantic Parsing as a Consistent Labeling Problem, a set of constraints stating valid/invalid assignments is required to find a possible solution. PARDON uses three kinds of constraints: The first group contains the constraints that encode the linguistic information obtained from verb subcategorization models. The second group are additional constraints added to force a tree-like structure for the solution. Finally, a third set of constraints encoding statistical information about word co-occurrences was added in order to complement the subcategorization information available.

Following the PARDON’s framework, next sections will describe how the general modelization is adapted to the specific models from LEXPIR, that is the specific formulation of the *model application constraints*, *model combination constraints* and *structural constraints*. *Structural constraints* do not change but are included to show a more complete view of the formalization.

Moreover, extra specific constraints modelizing PP-attachment and Lexical Attraction are added to help the models application and to show the flexibility of the architecture (PP-attachment and Lexical Attraction).

V.4.3 Model Application Constraints

Two different kinds of subcategorization models have been used: one about verbal subcategorization and another one about noun modifiers.

For each chunk labelled as verb phrase (VP), all possible subcategorization models for the verb heading the chunk are retrieved from LEXPIR. For prepositional phrases (PP) and noun phrases (NP) we use the simple nominal modifier model N_{de} presented in table V.3.

N_{de} model for nouns					
Catg.	Handle	Comp.	Sem.	Agree.	Opt.
PP	de	modifier	Top	no	no

Table V.3: Model for noun modifiers

Due to the richness and complexity of natural language, the prototypical subcategorization patterns defined in LEXPIR do not reflect exactly the complex patterns to be found in real data. Thus, a measure of the “goodness” of the possible model instantiation is defined in a similar way to the tree-edit based pattern matching used in [Atserias et al., 1999; Atserias et al., 2000].

In order to ensure the global applicability (minimal disorder, agreement, maximum similarity between the role and semantic object and maximal number of roles) and the consistence of the model (a unique instantiation per role and the instantiation of compulsory roles) the following constraints are automatically instantiated from the models:

- **Model Support:** A model assignment is compatible with its optional roles, e.g.: $[hablar.model = basic] \sim [c6.role = (entity, basic, c5)]$
- **Model Inconsistence:** A model assignment is incompatible with the inexistence of any of its compulsory roles, e.g.:

$$[c5.model = impersonal] \approx \neg [c4.role = (se, impersonal, c5)]$$

- **Role Support:** A role assignment is compatible with the assignment of its model, e.g.:

$$[pension.role = (entity, basic, c5)] \sim^{+sim(\dots)} [c5.model = basic]$$

The weight for this constraint is the *similarity* between the feature structures of both the Semantic Object and the Role (e.g in the constraint example, the similarity between the Semantic Object associated to “*pension*” and the Role *entity* of the *basic* model of the verb “*hablar*”).

V.4.4 Model Combination constraints

On CLP the different assignments of a variable are incompatible. Thus, using the formalization proposed in this chapter the *Object Instantiation Uniqueness* and *Model Uniqueness* constraints are ensured by the algorithm itself (as the labels of a variable are incompatible among them). Thus, only **Role Uniqueness** and **Role Inconsistence** are modeled as CLP constraints.

- **Role Uniqueness:** The same role cannot be assigned to different chunks, e.g.:

$$[c6.role = (entity, basic, c5)] \approx [c3.role = (entity, basic, c5)]$$

This constraint penalizes the current weight of the assignment of the role *entity* of the verb *hablar* (*c5*) to the *pensión* (*c6*) $[c6.role = (entity, basic, c5)]$ according to the current weight of the assignment of the same role to *partido* (*c3*) ($[c3.role = (entity, basic, c5)]$). Thus, the higher the weight for the latter assignment is, the faster the weight of the former will decrease.

- **Role Inconsistence:** A role assignment is incompatible with the *non existence* of the assignment of its own model, e.g.:

$$[c6.role = (entity, basic, c5)] \approx \neg [c5.model = basic]$$

V.4.5 PP-attachment constraints

Additionally, a special set of constraints has been introduced to deal with PP-attachment:

- **Local PP attachment:** A prepositional phrase tends to be attached to its nearest head. The weight assigned to each constraint will decrease along with the distance (in words) between the semantic objects involved, e.g.:

$$[pension.role = (entity, impersonal, c5)] \sim^{-distance(c6,c5)} [].$$

Note that there is no right-hand side on the constraint as it is valid for any context

V.4.6 Structural Constraints

As described in section IV.7.2.1, some further constraints must be included to force the solution to have a tree-like structure. These constraints are not derived from the subcategorization models:

- **TOP Uniqueness:** Different assignments of the label TOP are incompatible, e.g.: $[c3.role = TOP] \approx [c5.role = TOP]$

- **TOP Existence:**

There is at least one TOP. We will give support to all TOPs

$$e.g.: [hablar.role = TOP] \sim []$$

an alternative will be to give support to a TOP existence according to the other TOP assignment

$$e.g.: [c5.role = TOP] \approx [c1.role = TOP] \vee [c2.role = TOP] \vee [c3.role = TOP] \vee [c4.role = TOP] \vee [c6.role = TOP]$$

- **No Cycles:** Two assignments forming a direct cycle are incompatible¹³, e.g.:

$$[c6.role = (modif, N_{de}, c3)] \approx [c3.role = (modif, N_{de}, c6)]$$

- **NONE Support:** The NONE model is compatible with the inexistence of any role assignment of the semantic object models, e.g.:

$$[congreso.model = NONE] \sim \neg [c6.role = (modif, N_{de}, c2)] \wedge \neg [c3.role = (modif, N_{de}, c2)]$$

If these constraints were not included, the NONE model would never be selected, since there would always be some other model with a very small non-zero support.

¹³In this first prototype of PARDON indirect cycles are not taken into account.

V.4.7 Modeling Lexical Attraction

In a similar way to [Yuret, 1998] we also define a language model based on lexical attraction. In our case, we estimate the likelihood of a syntactic relation not between two words but between two semantic objects.

Our hypothesis is that the relations between two semantic objects can be determined taking into account two special elements of their associated chunks, the *handle* and the *head*. The *handle* of a chunk is usually the preposition that specifies the type of relation that chunk has with another chunk, while the *head* of a chunk is supposed to capture the meaning of the chunk [Basili et al., 1998]. For instance, the chunk “de las pensiones” (*about the pensions*) has handle “de” (*about*) and head “pensión” (*pension*).

Since related words are expected to occur together more likely than unrelated words, the lexical attraction (the likelihood of a syntactic relation) between two words can be estimated/modelled through co-occurrence. Co-occurrence data can also indicate negative relatedness, when the probability of co-occurrence is lower than by chance. Thus, we will measure the lexical attraction between two semantic objects as the co-occurrence of both heads and the co-occurrence of the head and the handle (which gives an implicit direction of the dependence).

Since the co-occurrences were taken from the definitions of a Spanish dictionary, lemma co-occurrences were used instead of word co-occurrences in order to minimize the problems caused by unseen words [Dagan et al., 1999]. 175,333 head-handle co-occurrences and 961,470 head-head co-occurrences were obtained out of 40,591 different head-lemmas and 160 different handle-prepositions. The co-occurrences were used to compute Mutual Information for each lemma-preposition pair.

$$MI(head_i, handle_j) = \log \frac{P(head_i \cap handle_j)}{P(head_i) \times P(handle_j)}$$

In the case of lemma-lemma pairs, sparseness is much higher. Thus, an indirect measure was applied, namely context vector cosine (also used in IR and WSD [Schütze, 1992]) in order to calculate the lexical attraction between heads:

$$\cos(head_i, head_j) = \frac{\sum_k a_{ki} a_{kj}}{\sqrt{\sum_k a_{ki}^2 \sum_k a_{kj}^2}}$$

where a_{pq} is the co-occurrence frequency of lemma p and lemma q , and k ranges over all the lemmas co-occurring with any of both heads.

Thus, for any two semantic objects the following constraints are added:

- A_i-H_j constraint, which supports any assignment of a role from $object_j$ to $object_i$, e.g.:

$$[c3.role = (modif, N_{de}, c2)] \sim^{MI(congreso,de)} []$$

- H_i-H_j constraint, which supports any assignment of a role from $object_i$ to $object_j$, or viceversa, e.g.:

$$[c6.role = (entity, impersonal, c5)] \sim^{\cos(hablar,pension)} []$$

H_i-H_j and A_i-H_j constraints can be used to identify adjuncts or relations for which we have no models. For instance, in the result obtained for the sentence shown in Figure V.3, the semantic object “en el congreso” (*in the congress*) will be identified as depending on the verb “hablar”, even when its role can not be determined.

V.4.8 Initial State

In order to establish the initial weight for each type of variable we choose the following heuristics:

- **Roles** are initialized according to the static similarity function.
- **Models** are initialized according to the result of the static evaluation of the similarity function of their roles assignments.

V.5 Experiments

170 real sentences were taken from a Spanish newspaper and were labelled by hand with their verbal models and meaning components. The sentence average length is 8.1 words, ranging from 3 to 23. Our approach to semantic parsing has been designed to manage multiple models simultaneously competing for their arguments. However, since our knowledge base does not include models needed for complex sentences, such as models for subordination or coordination, only one-verb sentences were selected.

Each sentence in the corpus was tagged and parsed with a wide-coverage grammar of Spanish [Castellón et al., 1998] to obtain a chunk parse tree. Spanish Wordnet [Atserias et al., 1997] was used to semantically annotate the corpus with the 79 semantic labels defined in the preliminary version of the EuroWordNet Top Concept Ontology [Vossen, 1998].

As mentioned in section V.4, in order to reduce the complexity of the relaxation process, the possible role labels (which indicate the roles an object can play in any of the models retrieved) are filtered considering the unary constraints about POS and prepositions, while constraints about semantics and agreement are taken as a measure of how similar (*sim*) the semantic object and the role are. Models which can not match compulsory roles are not considered.

For instance, the semantic object *año* (*year*) in the example sentence will be allowed to match the role *starter* of the impersonal model of the verb *hablar* even though its semantics is not Human, but the semantic object *congreso* will not be considered as a candidate to fill the *entity* role of *hablar*, since the preposition *en* in the semantic object does not match the model requirements for that role (preposition *de, sobre*). All these filters produce the candidate labels shown in Figure V.5, which are the input to PARDON *Selection* step.

V.5.1 Results

The results reported have been calculated using evaluation metrics from Message Understanding Conferences [MUC, 1995] applied to our particular case of verbal model identification and case-role filling.

Model identification metrics evaluate how well our system identifies the right model for a semantic object. Our corpus has 2.7 models per verbal semantic object as average ambiguity.

Since it is assumed that there is only one right model per chunk in each sentence, the answer can only be correct (*COR*) or incorrect (*INC*), thus, the used metrics are precision and recall. Table V.4 shows the results obtained in the verbal model identification task: 95% precision and 91% recall.

<i>COR</i>	<i>INC</i>	<i>PRE</i>	<i>REC</i>
155	8	95%	91%

Table V.4: Verbal Model identification results

Case-role filling consists in assigning each semantic object to the right role in the models for other semantic objects. In this case, the casuistry is more complex, since in addition to the correct/incorrect distinctions, other cases must be considered, such as the roles that are (correctly/incorrectly) left unassigned (because they were optional, or because there was no semantic object that fitted them, etc.). The MUC evaluation metrics establish the following cases:

- **Correct** (*COR*): Roles correctly assigned by the system.
- **Incorrect** (*INC*): Roles incorrectly assigned by the system.
- **Missing** (*MIS*): Roles unassigned by the system when they should have been assigned.
- **Spurious** (*SPU*): Roles assigned by the system when they should have been unassigned.

These cases lead to the definition of the following measures, where **Possible** (*POS*) are the roles that should be assigned (*COR*+*INC*+*MIS*) and **Actual** (*ACT*) are the roles actually assigned by the system under evaluation (*COR*+*INC*+*SPU*):

- **Undergeneration** $UND = 100 \times \frac{MIS}{POS}$
- **Overgeneration** $OVR = 100 \times \frac{SPU}{ACT}$
- **Substitution** $SUB = 100 \times \frac{INC}{COR+INC}$
- **Error** $ERR = 100 \times \frac{INC+SPU+MIS}{COR+INC+SPU+MIS}$
- **Precision** $PRE = 100 \times \frac{COR}{ACT}$
- **Recall** $REC = 100 \times \frac{COR}{POS}$

In addition, precision and recall may be combined in different F-measures (*P&R*, *2P&R* and *P&2R*). Table V.5 shows the results in the case-role filling for verbal arguments.

<i>COR</i>	<i>INC</i>	<i>MIS</i>	<i>SPU</i>	<i>POS</i>	<i>ACT</i>
203	27	60	51	290	281

<i>UND</i>	<i>OVR</i>	<i>SUB</i>	<i>ERR</i>	<i>PRE</i>	<i>REC</i>	<i>P&R</i>	<i>2P&R</i>	<i>P&2R</i>
20%	18%	12%	40%	72%	70%	71%	70%	72%

Table V.5: Verbal case-role filling results

To our knowledge there is neither a similar general approach nor case-role filling experiments to which our results can be compared. In any case, our preliminary results (72% *PRE* - 70% *REC*) are very encouraging.

It is also remarkable that our system produces low values for *UND*, *OVR* and *SUB* measures, pointing that it properly uses the different kinds of knowledge, and that it does not take uninformed or gratuitous decisions.

Errors in the preprocessing steps caused most of the mis-identified models (table V.4, *INC*). The *missing* and *spurious* roles (table V.5, *MIS* and *SPU*) were due either to the lack of semantic information or to the lack of a verbal model for adjuncts, which caused mis-identification of adjuncts as arguments, as in “(*Juan*) (*esquía*) (*este fin*) (*de año*)”¹⁴, where the chunk “*este fin de año*” (*on New Year’s Eve*) is wrongly identified to fill the *route* role even though its semantics is *Time*. This is due to the lack of a selectional preference that forces the *route* to be a *Place*, and to the lack of a model that identifies the chunk as time adjunct.

V.6 Discussion

This chapter has described a new approach to Semantic Role Labeling for non domain-specific texts based on the *Interactive Model*. The robustness and flexibility of PARDON are achieved combining a chunk parsing approach with the framing of the semantic parsing problem in a CLP. The flexibility of our approach enables the integration of different types of knowledge (linguistically motivated subcategorization models plus statistical information obtained from corpora).

Currently, PARDON obtains a 95% precision on model identification and 72% precision on role filling. Although the experiments have been carried out on a limited corpus and lexicon, they have proved the feasibility of the method.

Further work should include a more realistic evaluation of the system, using a larger corpus with sentences having multiple verbs (maybe using the models and corpus related to other lexical resources available for English such as FrameNet or PropBank). In this case, verbal models would compete for their arguments in a sentence.

We also plan to include more statistical knowledge (measures/language models) and to extend the coverage and expressiveness of the subcategorization models. Ex-

¹⁴ *John is going skying on New Year’s Eve.*

exploiting the integration of other semantic resources related to Wordnet (e.g the Multilingual Central Repository [Atserias et al., 2004f], developed inside the MEANING Project¹⁵ [Rigau et al., 2002] which contains selectional preferences automatically acquired from corpus). Furthermore, the output of the current system could also be used as feedback to improve the existing verbal models.

The problem of the recognition and classification of Named Entities (such as proper nouns denoting people and companies, amounts of money, dates, etc.) should be addressed, studying how to extend the current lexical attraction model to cover Named Entities. Named Entities do not appear in the training data of the lexical attraction model but are quite frequent in our test corpus. Thus, the set of constraints generated using this model tends to support assignments which do not involve Named Entities.

Finally, the exploration of linguistic and statistical models for the identification/distinction of verbal adjuncts should also be investigated, since it seems to be one of the main causes of verbal argument mis-identification.

¹⁵<http://www.lsi.upc.es/~nlp/meaning/meaning.html>

CHAPTER VI.

A PARDON prototype for Word Sense Disambiguation

*“When I use a word”, Humpty Dumpty said in rather a scornful tone,
“it means what I choose it to mean—neither more nor less.”
“The question is,” said Alice, “whether you can make words mean different things.”
“The question is,” said Humpty Dumpty, “which is to be master—that’s all.”*

Lewis Carroll *“Alice in Wonderland”*

The main goals of the set of experiments presented in this chapter are, first, to prove the capability of PARDON to combine different sources of information to better solve a particular task (knowledge integration) and secondly, to demonstrate that combining syntax and semantics, even with noisy and poor coverage models, PARDON is able to obtain interesting results on Word Sense Disambiguation (hereafter WSD).

VI.1 Different Approaches to WSD

WSD can be defined as the process of deciding the meaning of a word in its context. The possible senses for a word are previously defined in a sense repository (that is, a dictionary or lexical resource). WordNet [Fellbaum, 1998], a lexical taxonomy built at Princeton University, has become the *de facto* standard sense repository in Natural Language Processing for English. Although WordNet was not designed to serve as a lexical resource, its public availability and reasonable comprehensiveness have been dominant factors in its selection as the lexical resource of choice.

Word sense disambiguation is an important objective in the language engineering community. The first attempts to perform a kind of WSD, were embedded modules in sentence interpretation systems. Since then WSD has been evolving rapidly and the NLP community has developed multiple approaches and systems for WSD, but it

still remain an open problem, not only about how to solve but also on the exact definition of the WSD problem itself. [Stevenson, 1999] differentiates different levels of WSD based on the information used: *knowledge based* (e.g. based on dictionary definitions), *corpus based* (e.g. supervised or unsupervised machine learning techniques) and *hybrid approaches* (combining in some way the two previous categories).

A promising current line of research on WSD uses semantically annotated corpora to train Machine Learning (ML) algorithms to decide which word sense to choose in which contexts. Five of the most frequently used ML methods are: Naive Bayes, Example based, Support Vector Machines, Decision Lists and Vector Models [Márquez et al., 2006].

Supervised WSD systems are data hungry, they suffer from the “knowledge acquisition bottleneck”. These approaches are named “supervised” because they learn from previously sense annotated data and therefore they require a large amount of human intervention to annotate the training data. Although ML classifiers are undeniably effective, they will not be feasible until obtaining reliable unsupervised training data.

Thus, some recent work is focusing on reducing the knowledge acquisition cost and the need for supervision in corpus-based methods for WSD. [Leacock et al., 1998; Mihalcea and Moldovan, 1999; Agirre and Martinez, 2000; Martinez, 2004; Cuadros et al., 2006] automatically generate arbitrarily large corpora for unsupervised WSD training, using the knowledge contained in WordNet to formulate search engine queries over large text collections or the Web.

As different systems and approaches emerged, there was a need to compare fairly these systems. A really difficult task if they use different sense repositories or test corpus. The *initiative for the Evaluation of Systems for the Semantic Analysis of Text* (SENSEVAL) framework was designed to address this problem [Kilgariff, 1998]. SENSEVAL¹ is devoted to the evaluation of Word Sense Disambiguation Systems. Its mission is to organise and run evaluations and related activities to test the strengths and weaknesses of WSD systems in different tasks. From SENSEVAL-II to SENSEVAL-III there was not a significant improvement in performance for English. The best systems are still about 65-70%. In fact, it seems that new approaches to WSD are needed.

This chapter wants to explore two approaches: on the one hand whether the integration of knowledge already available (knowledge integration) could improve WSD. On the other hand whether the integration of WSD with other NLU process (process integration) could also improve the overall figures on this task.

Using the PARDON’s architecture, we would formulate the *Word Sense Disambiguation* problem in a similar way than the *Semantic Role Labeling* in chapter V. Each word sense would have associated a set of models with syntactic and semantic information and our task will be to establish which of these models is more similar to the input sentence. Thus, the syntactic-semantic model selected will establish the correct word sense, not only for the syntactic head but also, if possible, for the rest of the roles. Thus, the word is assigned the sense of the most similar example

¹<http://www.senseval.org>

already seen, as in example-based learning: LEXAS [Ng and Lee, 1996], TIMBLE [Hoste et al., 2001], GAMBL [Decadt et al., 2004].

During the pre-processing phase the input sentence containing the word to disambiguate will be syntactically parsed and the syntactic dependencies between their elements obtained. Then, each word and its grammar dependencies is transformed into a feature structure. The resulting ordered sequence of feature structures will be used by the PARDON-WSD system.

The following sections, explains our approach to WSD and how we have adapted the PARDON's architecture for WSD.

VI.2 Applying the PARDON approach

Despite the fact that WSD and SRL are strongly correlated, traditionally, most of the systems treat both separately. Paradoxically, WSD can improve SRL, as the different senses of a word could present different syntactic structures (specially verbs) and the other way round, SRL can help WSD (e.g. selectional preferences could determine the right sense of the verb and its objects [Carroll and McCarthy, 2000]). Moreover, there are few examples of a real use of syntactic information for WSD, [Lin, 1997], [Mihalcea and Faruque, 2003]. Most WSD system rely on low level attributes (e.g. local features and bag of words) ignoring syntax or using syntax in a shallow manner.

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

VI.2.1 PARDON's input

During the first pre-processing step, the input sentence containing the word to disambiguate is syntactically parsed using RASP [Carroll et al., 1998], obtaining the syntactic dependencies between the words in the sentence. Figure VI.1 shows the dependency analysis obtained for the sentence “*The cat eats fish*”.

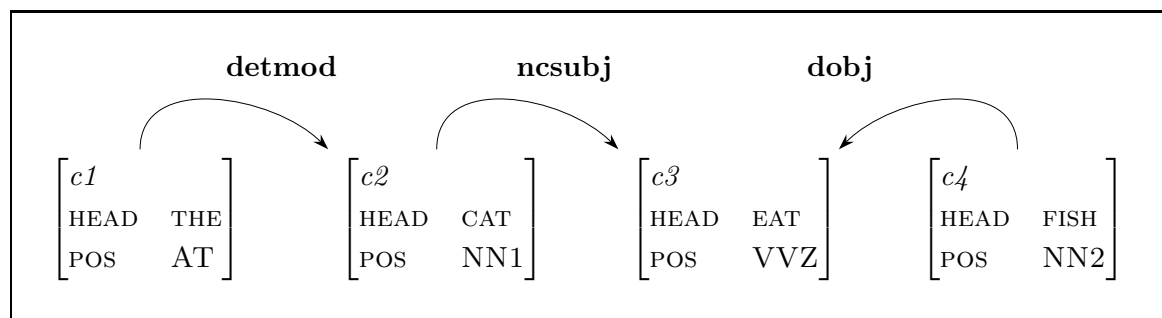


Figure VI.1: Dependencies for “*The cat eats fish*”

Then, each word is tagged with all its possible senses in WordNet. We use an specific tool for lemmatizing and recognizing multi-word expressions (MWEs) according to WordNet [Arranz et al., 2005] instead of the lemmatization/tokenization provided by RASP. Lemmatizing and recognizing MWEs is not only relevant to WSD (as they tend to be less ambiguous) but also to PoS tagging and parsing as many of them have an idiosyncratic syntactic structure.

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., 2004f], that is: the expanded EuroWordNet’s Top Concept Ontology [Atserias et al., 2004a], Suggested Upper Merged Ontology (SUMO) [Niles and Pease, 2001] and MultiWordNet Domains [Magnini and Cavaglia, 2000].

The resulting information (syntactic dependencies and semantic information) for each word is converted to a feature structure. Figure VI.2 shows the feature structure obtained for *fish*, which contains the information related to its two senses: 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.

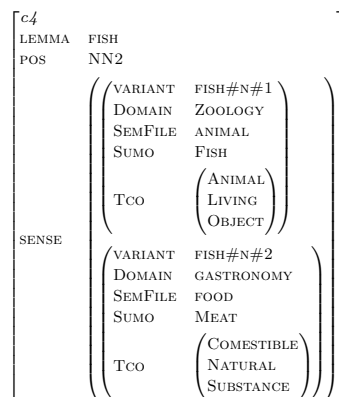


Figure VI.2: Object *Fish* enriched with MCR information

VI.2.2 Lexicalized Models for SRL and WSD

The PARDON approach to WSD relies on the existence of lexicalized models associated to a word which determines how this word can be combined with other words. We will also refer to this word as the *head* of the model and the rest of components of the models as *roles*.

In order to integrate SRL and WSD following the PARDON approach we need to relate these lexicalized models to WordNet. That is, having explicit information of the WordNet senses not only for the *head* of the model but also for the semantic preferences of the *roles*. Moreover, once we have build models with WordNet sense information, we can enrich those models using all the information stored into the MCR (see chapter III).

model <i>basic</i> for “hablar” (to talk)					
Synt.	Prep.	Rol	Semantics	Agree.	Optional.
NP	x	starter	Human	yes	yes
PP	de, sobre	entity	Top	no	yes
PP	con	destination	Top	no	yes

Table VI.1: Example of LEXPIR Syntactic-Semantic model for SRL

Apart from being related to WordNet, the models used in this chapter for the formalization of PARDON as a WSD system will be similar to those used in the previous chapter for Semantic Role Labeling, (as the one in table VI.1), but using syntactic grammatical dependencies instead of syntactic label of *chunks*.

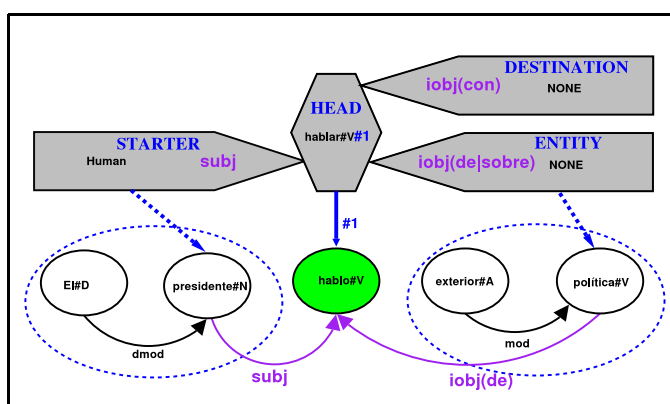


Figure VI.3: Model Matching

Figure VI.3 shows the application of an adaptation of the model seen in table VI.1 (converting the chunk information into dependencies) to a simple Spanish sentence (literally, *the president talked about foreign policy*). The models whose head is *hablar* are ‘anchored’ on the word *habló* (to talk) which is graphically represented by a blue thick arrow. Two of the three roles of the models can be instantiated by words in the real sentence (blue dotted arrows). That is, the word *presidente* (president) could be the STARTER while the word *política* (literally politics) could be the ENTITY. Notice that since we are now relying on dependencies not chunks or parsed trees, to retrieve the ‘whole’ role (blue dotted circles) we need to follow the dependency relations, that is ‘*El presidente*’ (the president) for the STARTER and ‘*política exterior*’ (foreign affairs/policy) for the ENTITY.

In order to disambiguate all the content words in the sentence, we will need to use all the models for the content words of that sentence. However, having to disambiguate only one word (that is, the target word) in the sentence (i.e. lexical sample task), arises a new issue in PARDON’s modelization: We need to determine which are the words whose models should be considered in order to disambiguate the target word.

Her remorse was shallow and brief. Although she was kind and playful to her children, she was dreadful to her war-damaged husband; she openly brought her lover into their home. As presented by Mr. Chabrol, and <head>played</head> with thin-lipped intensity by Isabelle Huppert, Marie-Louise (called Marie La tour in the film) was not a nice person.

Figure VI.4: play.131 example of the SENSEVAL-II English lexical sample

For instance, figure VI.4 shows a test sentence (play.131) of the SENSEVAL-II English Lexical Sample. The word to be disambiguated (target word) appears enclosed inside the tag *head* (in this example, the verb *to play*). The models that can take part directly or indirectly in the disambiguation of the target word depend greatly in the dependency analysis obtained for the sentence, shown in figure VI.5.

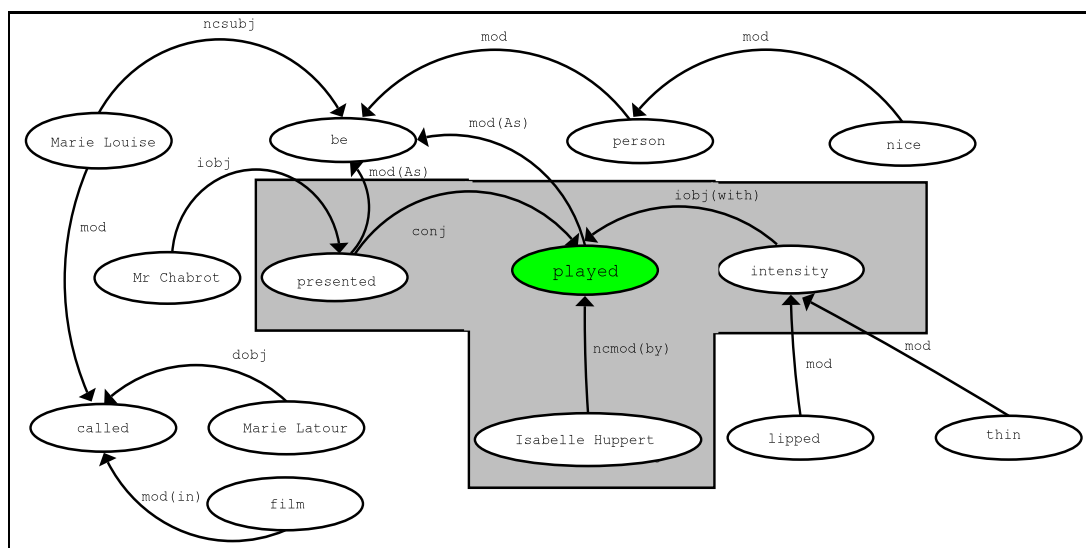


Figure VI.5: Dependency Analysis

First, in order to disambiguate the target word, we should consider the models associated to the target word itself (that is the models for all the senses of the verb *play*). The application of those models will also relay indirectly in the disambiguation of the nodes that directly depend on the word *to play* (the grey area). That is, the words: *presented*, *intensity*, *Isabelle Huppert*. Thus, we also need to take into account those models that could be applied to those words and try to obtain semantic information for these nodes.

We can also consider the models for the word in the sentence on which the verb *to play* depends. In the example, applying a model for the verb *to be* may help to determine the sense of the verb *to play*. Again, disambiguating any word which could take part of all the models already considered can help to better apply those models.

In fact, we are calculating a closure over the grammatical relations, activating all the models of the words which are in this closure (that is, the connected graph which contains the target word). Notice that, in the current formalization, it does not matter what the analysis and models of the first sentence of the example are, as they are not connected to the target word (no referential analysis is performed), they can not help in its disambiguation.

VI.2.2.1 *Determining the applicable models to disambiguate a Word*

Given a word in a sentence, we decide to retrieve only those models whose syntactic head has the same lemma and PoS than the word. However, usually it is not the case that we have enough models for all the senses of the words to be disambiguated. Although it is reasonable to apply models associated to similar words to cover these lack of models, it seems difficult to establish how to extend the set of models that should be considered.

Many criteria can be devised, for instance based on the fact that the syntactic head is not necessarily the semantic head, or that similar verbs could tend to have similar behaviours. We could retrieve models where the word is doing the same syntactic function (e.g. in the example sentence we can retrieve all the models where *play* appears as modifier with the preposition *as*), or even retrieve all the models from semantically related words (e.g. the variants of the same synset, Levin Classes, etc).

VI.2.3 *WSD Methods using PARDON's models*

Having our SRL models related to WordNet enables four different disambiguation strategies, two of them supervised and two other unsupervised.

Next subsections will explain the different WSD strategies that can be used to transfer sense information from the model which is being applied/identified to the words in the input sentence. The first two strategies are *supervised* in the sense that requires sense information associated to the model. On the other hand the last two strategies are *unsupervised*. That is, they do not need any explicit sense annotation of the models.

VI.2.3.1 *Supervised Strategies*

There are two supervised strategies for transferring the sense information from the model to the words which are instantiating the model. The first, when the target word is the head of the model being applied (supervised-head-disambiguation), and the second when the target word is instantiating a role of the model (supervised-role-disambiguation).

In a first case (supervised-head-disambiguation), as we are restricting the models associated to a word to have the same lemma and PoS, the sense can be transferred directly, from the model's head to the word in the sentence.

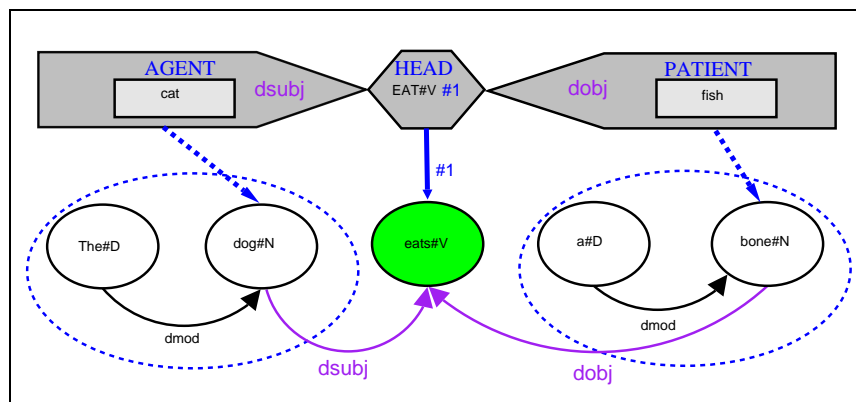


Figure VI.6: Model Matching

Figure VI.6 shows, when applying a model, how the sense information related to the model's *head* (i.e. *eat#v#1*) can be projected directly to the word in the sentence (i.e. *eats#v*) to which the model is associated (thick line).

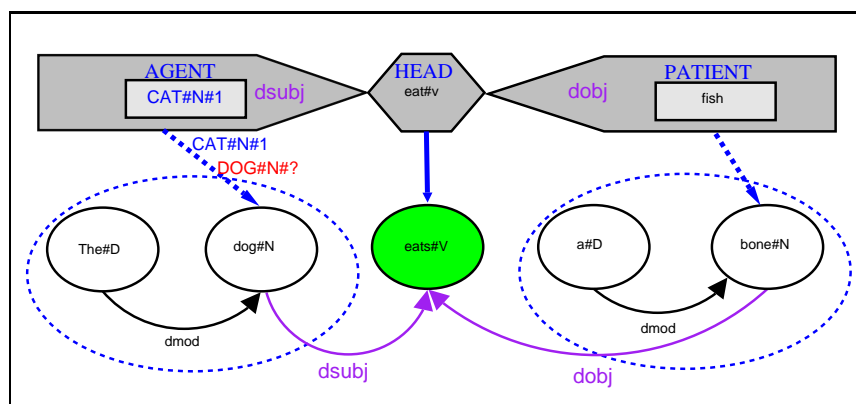


Figure VI.7: Role Matching

In a similar way, for the second case (supervised-role-disambiguation), figure VI.7 shows, when applying a model, how the sense information related to the role *agent* can be projected directly to the word in the sentence (*dog*) which is instantiating that role. In this case, as the word in the sentence and the word in the model are not necessarily the same, in order to project the sense information we must determine which are the senses of the word which are closer to the sense of the roles.

For instance, in the example it is necessary to determine which of all the possible senses of *dog* in WordNet (see figure VI.8) is closest to the sense of the role (that is, *cat#n#1*, a feline mammal unable to roar). The distance between senses can be calculated based on the different information relating senses encoded in WordNet: That is, the WordNet hierarchy, glosses or relations, the TCO, SUMO, Domains, etc. (see Chapter III for a complete overview of the current content of the MCR).

WordNet sense	Gloss
dog_1 domestic_dog_1	a member of the genus Canis
frump_1 dog_2	a dull unattractive unpleasant girl or woman
dog_3	informal term for a man
cad_1 bounder_1 dog_4	someone who is morally reprehensible
pawl_1 detent_1 click_3 dog_5	a hinged device that fits into a notch of a ratchet ...
andiron_1 dog_6 dogiron_1	metal supports for logs in a fireplace

Figure VI.8: Senses for the noun dog in WordNet

VI.2.3.2 Unsupervised

The two strategies above are based on having our head or roles of the model annotated with sense information (that is, supervised WSD). However, two similar methodologies using unannotated models can also be designed: unsupervised-role-disambiguation and unsupervised-head-disambiguation.

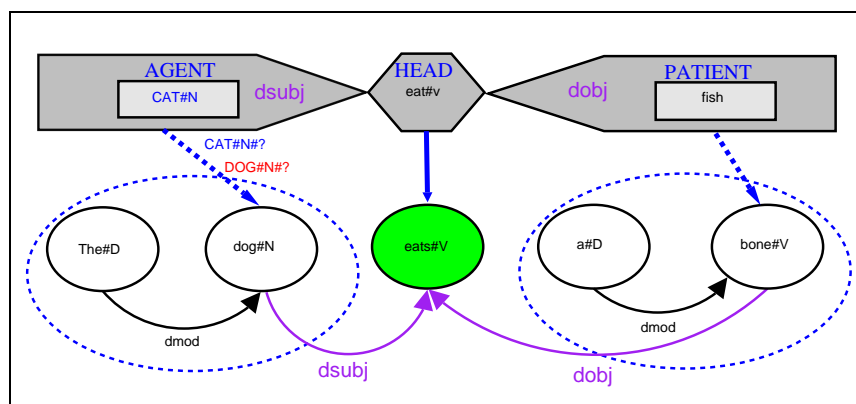


Figure VI.9: Role Matching

Figure VI.9 shows the unsupervised-role-disambiguation, a similar case than the one used when projecting the sense information from a role to the word of the input sentence, but in this case, the role has no explicit sense information. Although, comparing all the possible senses from the role against all possible senses of the word which is filling the role, we can determine which pair of senses are the closest and thus select a word sense for the word in the input sentence (and also for the role).

As can be seen in figure VI.8, both *cat* and *dog* have different senses related to *animal* and related to *person*. A semantic distance measure is not likely to determine which pair of senses, the ANIMAL ones or the HUMAN ones, are better. However, it

is important to notice that this sense projection will not be applied in isolation (as a unique selectional preference) but as the same time that the whole model is being applied. Moreover, other criteria, such as sense frequency, could be used to select the right pair of senses.

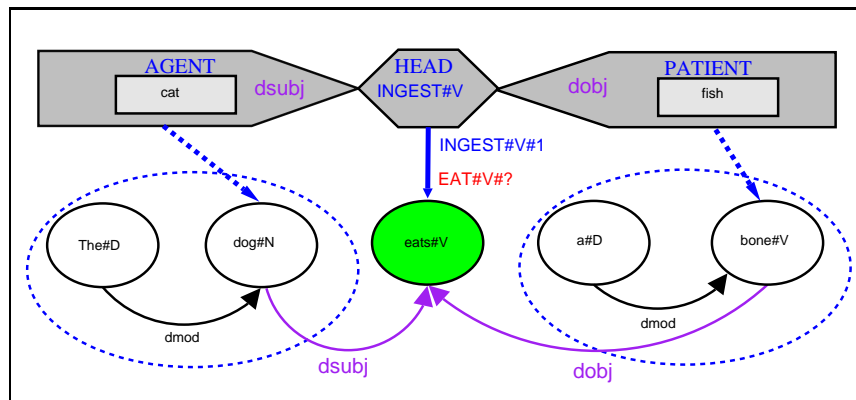


Figure VI.10: Model Matching

On the other hand, as shown in figure VI.10 a similar case could also apply for the *head* of a model. However, in order to apply new head-based strategies (supervised or unsupervised) the lexical restrictions on the applicable models, that is same lemma and PoS, must to be soften. For instance, we can consider applying not only the models directly associated to the word but also the models which are 'closer' to that word (e.g. from variants of the same synset, hypernyms/troponyms, from verbs in the same levin's class, etc). That would ease the lack of models but also implies the activation of less accurate models.

Once relaxed the lexicalization constraint of the models, a similar strategy to the *unsupervised-role* could be applied to the head. As shown in figure VI.10, if a role from a model with a 'closer-head' could be instantiated, we can consider projecting sense information from the head-model to the word in the sentence which instantiates the head. These unsupervised-head-disambiguation strategies arise many issues, such as: how much unsupervised models are useful, how the unsupervised models interact with the supervised, how to keep the trade between precision and coverage, etc, due to all these issues the head unsupervised strategies will not be addressed in the current experiments.

VI.3 PARDON's Formalization for WSD

In the previous chapter, the input of PARDON Semantic Parser was a sequence of chunks. This time, we will feed PARDON Word Sense Disambiguator with grammatical relations (dependencies). When using PARDON for Semantic Parsing, the solution (the roles and the diathetic model involved) was directly obtained through the application of the models. Now, applying PARDON's Architecture to WSD, the correct senses of each word are not only determined by the direct application of a model for this word but also for the models where this word is playing a role.

Models would have senses associated, to both head and roles. Thus, we would need to add constraints to ensure the consistence between the models applied and the sense selected.

VI.3.1 Knowledge Representation

Basically, the formulation is similar to the one presented in the chapter V. Regarding the variables and labels in the CLP, we will also use a triplet (role, *model*, object) to identify a *role* from a *model* of an *object*. Similarly, since a CLP always assigns a label to all the variables; we will also use the two previously defined null-labels: 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).

VI.3.2 Attribute Representation

As an example, Figure VI.11 shows the CLP variables and labels for the sentence “*The cat eats fish*”. As we are focusing on WSD, the only dynamic attribute is the sense. Thus, we have three variables per object, role, model and sense.

Variable Name	Possible Labels
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	{ ag.m1.c2, ag.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	{ pat.m1.c3 }

Figure VI.11: CLP for *The cat eats fish*

Regarding constraints, most of the formalization is exactly the same than in the previous chapter. Thus, we will mainly focus on the differences.

VI.3.3 Role and Model Application

In order to apply the $sim(Object, Role)$ measure we established a particular similarity measure for each feature. Some of these measures can be defined *ad-hoc* and each one could also be easily improved individually. However it is our belief that since we are combining all these measures, the overall result will not improve dramatically. It remains as a future work to study this issue. Moreover, our final goal is not to overtune the system but to prove the feasibility of the application of the PARDON's architecture to different NLP tasks and not to build the best WSD system.

PARDON uses both syntactic and semantic features:

- **Syntatic Features:**

- **PoS:** We compare the main category. However more complex similarity functions between different CLAWS5 PoS tags can be defined (e.g. according to the number of characters of the longest common prefix).
- **Grammatical Relation:** The grammatical relations used in RASP [Carroll et al., 1998] are organised hierarchically: see Figure VI.12. The hierarchy could be used to calculate the distance between two different grammatical relations.

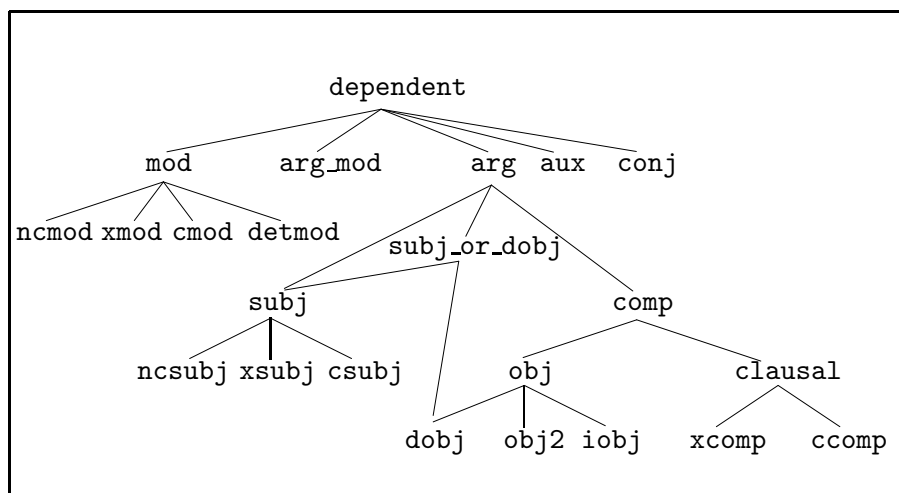


Figure VI.12: The grammatical relation hierarchy.

- **Side:** Whether the dependent is on the left side of the anchor or in the right side.

- **Semantic Features**

All the distances regarding semantic features are presented as sense comparison. The semantic measure used will be applied not just between two sense/synsets/domains/etc, but also between a word and a synset/domain/etc, or even between two words, by calculating the minimum distance between all the senses of these two words. Although other approaches are possible, e.g. the distances between values can be defined based on the training examples [Cost and Salzberg, 1993]), we will define the similarity base on the pre-existing knowledge about the attributes (e.g. their hierarchical structure).

- **Lexicographer File:** Being the possible lexicographer files a small set, we use the strict equality.
- **Sense:** WordNet can be used in many ways to calculate the *semantic distance* between two senses. There is an extensive literature² on different semantic measures using WordNet (e.g. [Agirre and Rigau, 1995],[Leacock and Chodorow, 1998],[Resnik, 1995], [Wu and Palmer, 1994], [Lin, 1998], etc). However, we decide to use a fast and simple measure to calculate this similarity: taking into account the level (from the top) of the lowest common subsumer (LCS). That is the *first* common ancestor of the two senses. In cases where there is no common ancestor (e.g. between *fish#n#1* and *material#n#2*) we consider this similarity null, so that the constraint does not hold.

It is our belief that the level (from the top) is a good measure for how abstract is the common ancestor. Due to the different granularity in the development of WordNet, it seems that the abstraction of the LCS could be a better measure than other distances between concepts usually calculated based on the number of edges of the path between the two senses.

In order to normalize the measure and keep the scores in the range [0-1] we take into account the depth of the hierarchy:

$$sim(a, b) = \frac{level(LCS(a, b))}{depth_of_hierarchy}$$

For instance, the noun *fish* has two different senses (*food* and *animal*). In order to determine which sense of the the noun *fish* is semantically closer to the first nominal sense of *meal* we used the WordNet1.7 hierarchy to calculate the *semantic similarity* between the two senses of *fish* and *meal#n#1*. Figure VI.13 shows a piece of the WordNet 1.7 hierarchy. It can be seen that *substance#n#1* is the LCS between *fish#n#1* and *meal#n#1* which is less abstract (lower position in the hierarchy) than the LCS between *fish#n#2* and *meal#n#1* (that is *entity#n#1*). So

²For an extensive survey consult the Ted Pedersen's bibliography recopilation at <http://www.d.umn.edu/~tpederse/wnsim-bib/>

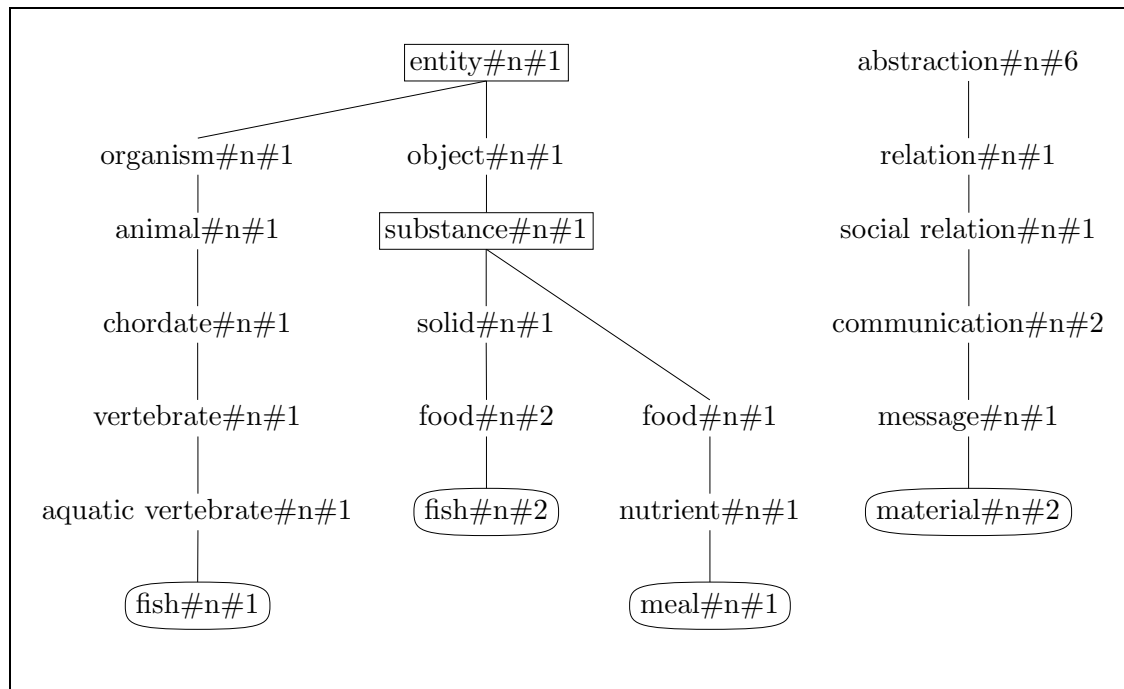


Figure VI.13: Example of semantic similarity over the WordNet hierarchy

that, the *food* sense of *fish* will be more similar to *meal* than the *animal* sense.

- **Domains:** Domains are also organised hierarchically. The size (165 labels) and depth (5 levels) of the hierarchy are smaller than WordNet's (around 100.000 labels and 15 levels). Thus, we decide to refine our measure. The previous measure does not take into account how much we have to generalize to reach a common ancestor.

Specially in case that one concept subsumed the other. That is, $LCS(a, b) = a$. Although it could argue that the concepts are similar whatever the level of the concept a , we think that the similarity measure should prefer closer subconcepts. Thus, we penalize the similarity according to the number of nodes between a and b (that is, $level_distance(a, b)$), so the measure in this case is:

$$sim(a, b) = 1 - \frac{level_distance(a, b)}{2 * depth_of_hierarchy}$$

It can also be argued that not being prototypical examples, we probably should also adapt the similarity measure for the case $LCS(a, b) = b$. That is, when b is an hypernym (or supra-concept) of a . However, we think that this will cause more over-generalization errors.

- **SUMO**: SUMO classes are also structured hierarchically. Although we only take into account the subclass relation and ignore the 'type' relationship. Using the same similarity function than for Domains seems to perform better than using the strict equality.
- **Top Concept Ontology**: The expanded Top Concept Ontology assigned several labels to each WordNet sense (which may contain labels which are subsumed). Since TCO set of labels is tiny and its hierarchy is extremely flat (5 levels), we decide to use the strict equality.

Nevertheless, for simplicity, we have made an obvious simplification defining our *similarity* measure to be calculated comparing slot to slot. Moreover, complex multiple slot match functions could also be formalized in this framework.

VI.3.4 Model Application Constraints

We should establish a set of constraints to ensure the right application of roles and models in isolation we should establish a set of constraints (**model instantiation Constraints**).

- **Model Support** This set of constraints gives support to a model according to its instantiated roles.

$$[c_x.model = m] \sim [c_y.role = (r, m, x)] \forall (r, m, x) \in Roles, \forall y \in Obj$$

For instance, if the model *eat-V4* has three possible roles (*ag*-ent, *pat*-ient, *ins*-trument), the constraint which supports this model according to the assignment of the role *pat*-ient will be $[c3.model = eat-V4] \sim^{\frac{1}{3}} [c3.role = (pat, eat-V4, c2)]$. The model will also have two similar constraints for the other two roles. The support received for a model is normalized by the number of its roles in order to not penalise small models. Thus, if we decide to give a weight of 1 to the model support constraint, this weight will be divided into the three support binary constraints corresponding to each role.

- **Role Support** The role support must take into account the senses which are associated to the object. Thus we need to compare each sense of the object with the possible senses of 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 (as defined in chapter IV).

For instance, the constraint $[c3.role = (pat, eat-V4, c2)] \sim^{.245} [c3.sense = fish\#n\#2]$ will give support to the assignment $(pat, eat-V, c2)$ taking into account the current weight of the assignment representing the sense *fish#n#2* and their similarity with the sense/s of the role $(pat, eat-V4, c2)$ (.245).

VI.3.4.1 Sense Constraints

The following set of constraints ensures that when a model is applied, the senses associated to this model are also selected, both for the head of the model and roles. As the current formalization does not include any constraint that modifies *Domain*, SUMO or TCO, these features do not need to be represented in the CLP and can be considered as *static*.

- **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:

$$[c.sense = s] \sim^{100} Or_{i=1}^n [c.model = m_i] \forall s \in c.sense$$

where $m_1 \dots m_n$ 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 assignment of the third sense of *eat* if any of the models associated to that sense (that is, eat-V17, eat-V52 or eat-V50) is selected.

- **Role Sense Disambiguation** This set of constraints associate 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 $sim_{static}(c_r.sense, r_r.sense)$ and $c_r.sense$ and $r_r.sense$ are the representation of the object and role associated to the sense $r.sense$.

For instance, $[c3.sense = fish\#n\#2] \sim^2 [c3.role = (pat, eat-V4, c2)]$ will select the second sense of *fish* if the object $c3$ fulfills the role *pat*-ient 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 *pat*-ient of *eat-V4* (which is 0.2).

VI.3.5 Structural Constraints

In order to gain flexibility on the type of models we can apply, the current formalization we will relax the models. The new models do not determine which parts are compulsory and which are optional. Thus, as there is no compulsory/optional roles we must establish a different criteria for both **Model Support** and **Model Inconsistence**.

- **Role Uniqueness:** A role can only be fulfilled by one object:

$$[c_x.role = a] \approx [c_y.role = a] \forall x, y \in Obj \forall a \in Roles \mid x \neq y$$

This constraint will avoid, for instance, that in the example sentence, (“*The cat eats fish*”), the object *cat* and *fish* fulfill the same role simultaneously.

- **Model Inconsistence:** A role can not be fulfilled by an object if the model to which the role belongs is not being instantiated:

$$[c_x.model = m_b] \approx [c_y.role = (r, m_a, x)] \\ \forall x, y \in Obj (r, m_a, x) \in Roles(y) \ m_b, m_a \in Models(x) \ | \ m_a \neq m_b$$

- **TOP Uniqueness** Only one TOP:

$$[c_x.model = TOP] \approx [c_y.model = TOP] \ \forall x, y \in Obj, x \neq y$$

- **TOP Existence** At least a TOP:

$$[c_x.model = TOP] \sim \nexists [c_y.model = TOP] \ \forall x, y \in Obj \ | \ x \neq y$$

- **NONE Support** The model NONE is compatible with the inexistence of the role assignments:

$$[c_y.model = NONE] \sim \nexists [c_y.role = a] \ \forall y \in Obj$$

VI.3.6 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. We initialize the role and model assignments according to the static similarity function, while for the sense assignments we can uniformly distribute the probability or use the sense frequency calculated on a particular corpus (e.g. SemCor or the SENSEVAL-II Lexical Sample Training corpus).

VI.4 Experiments

In order to prove the flexibility and robustness of our approach we applied our system to the *English Lexical Sample* of SENSEVAL-II. This task consists on 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 provided for the exercise and also because for verbs, WordNet senses were not directly used in SENSEVAL-III.

In order to apply our system to this task, we need models which contain syntactic and semantic information about roles and about WordNet senses.

On the one hand, as far as we are concerned there is no wide coverage resource that can be used for this task. Although there has been remarkable efforts to relate FrameNet and VerbNet with WordNet [Shi and Mihalcea, 2005], the coverage is still very low even to the Lexical Sample task. Only 50 senses of the test are directly associated to a FrameNet frame. That means, that even if our system was able to disambiguate all the words perfectly, only 640 sentences of the 4,328 could be solved correctly.

On the other hand, to automatically obtain models by parsing data which has been semantically hand-tagged has a lot of drawbacks. The acquisition of 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 WordNet senses and b) to have examples of all the possible syntactic subcategorization patterns for a sense. Second, state-of-the-art WSD systems and parsers still have significant error rates that machine learning algorithms can not cope with.

This chapter aims to demonstrate the robustness and flexibility of the PARDON's architecture rather than to obtain a *better than yours* WSD system. Thus, although its inherent complexity and the possible impact in the performance of the WSD system, we decide to automatically acquire those models.

The next subsections will describe the general methodology used to obtain models from semantically hand-tagged corpora and the models obtained applying the general methodology to several concrete corpora.

VI.4.1 Obtaining Lexical Models for SRL and WSD

In order to obtain models useful for both, SRL and WSD from raw text (with full, partial or null sense information) we can use the same pre-processing steps to obtain lexicalized models. As described before, RASP [Carroll et al., 1998] is used to extract grammatical relations from the corpus (e.g. subj / obj / dobj) and then the resulting data is enriched with all the semantic information gathered from the MCR.

Many times the corpora is not raw text and provides useful information (e.g. tokenization, lemmatization, PoS, MWEs, NE information). However, to take advantage of this information is difficult because of the differences in: PoS sets, lemmatization (e.g. *holy-of-holy* vs *holy-of-holies*), the identification criteria and the set of Multiword Expressions considered. As a simple example, the set of MWEs considered could differ greatly between a general corpus, WordNet and RASP. This differences could also have a large impact on performance of the different NLP processes. WordNet MWEs can misled the PoS tagger, or even contractions such as *n't* splitted from the verb, which must be converted to *not*, in order to be correctly interpreted by the parser.

Thus, although we process the corpus from scratch, we keep information about the original information provided by the corpus (e.g. tokenization, lemmatization, PoS, etc) in order to, later, look up the information from WordNet.

Once the sentence has been parsed, the set of dependencies extracted is used to build the models in a straightforward manner. For each syntactic head, we build a model containing the set of direct dependencies which arrive to it. No generalization

process is carried out. As the state-of-the-art SRL are far from being generally applicable, we decide to simplify our models and take this set of relations as if they were the set of roles of a model for this word.

For example, one of my favourite movies is the 1949 British comedy “ Kind Hearts and Coronets ” , in which the entire comedy is based on actor Dennis Price’s murdering eight titled relatives (all <head>played</head> by Alec Guinness) because they snubbed his mother and stand in the way of his acquiring the family title.

Figure VI.14: SENSEVAL-II English Lexical Task Test Paragraph *play.009* corresponding to sense *play#v#4*

As an example, consider the sentence in figure VI.14 where there is only a disambiguated word (enclosed between the tag <head>). Then, a set of grammatical dependencies related to the target is obtained using RASP (see figure VI.16). Each box in the figure illustrates the sets of words clustered by the grammatical dependencies around a word (*play*, *Dennis Price* and *relative* respectively).

A lexicalized model for that word can be build extracting these sets of *depending* words, that is, extracting for each content word all the direct grammatical dependencies that point to it.

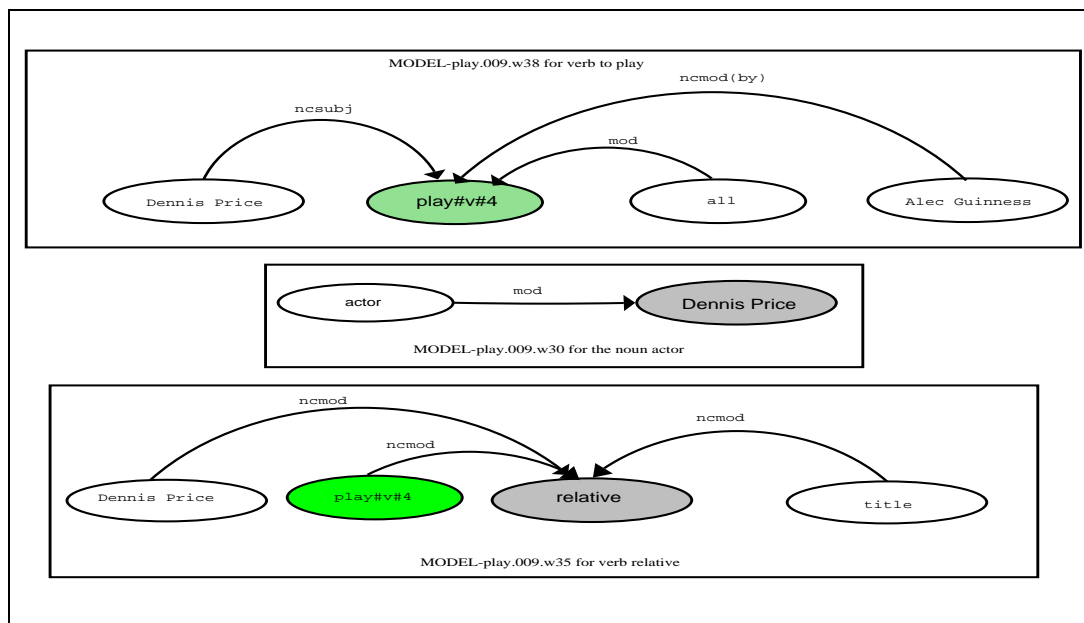


Figure VI.15: Models for the words *play*, *relative* and *Dennis Price* obtained from the senseval example *play.009*

Figure VI.15 shows the three different models obtained using this approach. The model *play.009.w38* for the verb *play* which only has its head disambiguated (in green), a model for the noun *relative*, named *play.009.w35*, which only has one

of its role disambiguated (in green) and a model for the proper noun *Dennis Price*, *play.009.w30*, with no explicit sense information. These examples illustrate the three different types of models which could be obtained (head-disambiguated, role-disambiguated, no-sense-information) and whose applicability depends on the selected WSD strategies.

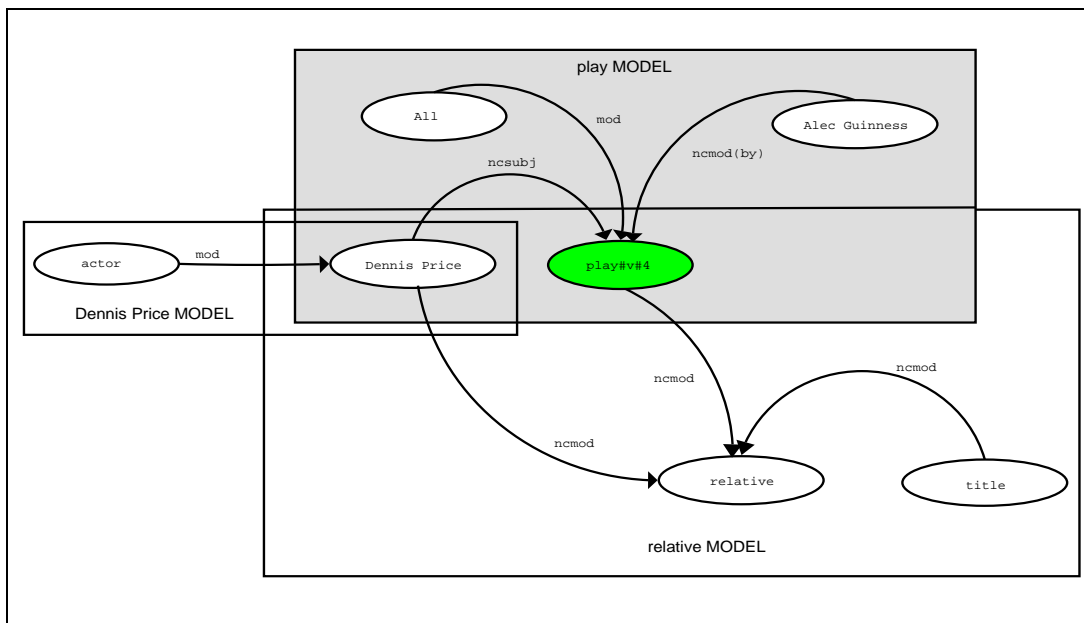


Figure VI.16: Extracting models from a set of dependencies

Another big difference with respect the SRL system, is that the models are not necessarily mutually exclusive. That is, two models could be different realizations of the same subcategorization frame or even be exactly equal. However, it can not be easily determined whether the two models obtained are part of the same subcategorization frame or not. Thus, we decide to formalize them as mutually exclusive.

VI.4.1.1 Extracting Models from several sense-tagged corpora

Although the proposed architecture allows the integration of WSD and SRL, the lack of wide coverage resources for SRL which can be related to WordNet synsets has forced us to acquire automatically the lexical models needed to carry out these tasks. Even though the models acquired are based not on semantic roles but on syntactic dependencies, they allow to test the flexibility and robustness of our approach against a well established WSD task.

The models used in the experiments have been obtained from two corpus with different characteristics. On the one hand, we used the SENSEVAL-II training corpus for the *English Lexical Sample* task whose 8,611 examples have only one word disambiguated. On the other hand, we used SemCor [Miller et al., 1993], which is a subset of the Brown Corpus (about 250,000 words), consisting of texts that have been tagged with PoS information and semantic information.

SemCor was semantically annotated with WordNet-1.6 senses including Named Entities and MWEs, and actually, automatically mapped to WordNet-1.7, WordNet-1.7.1 and WordNet-2.0³. We should keep the original tokenization in order to be able to recover the semantic information (synsets) after the parsing.

The SENSEVAL-II lexical sample corpus is provided without any kind of preprocessing (tokenization, Named Entity Recognition and classification, detection of MWEs⁴, etc). Thus, our preprocessing could bias greatly the results, as we will be evaluating not only the performance of the WSD system, but also the performance of our preprocessors. That is, how good are our tokenizer, PoS Tagger, lemmatizer, etc).

	#models	#models-senses-in-test	#sense-head	#sense-role
SemCor	246,083	1,015	667	348
Senseval	75,707	13,068	6,073	6,995

Table VI.2: Models acquired for the 73 words included in SENSEVAL-II test corpus

Table VI.2 shows the figures for the models obtained from each corpus, the amount of models obtained (#models), as well as the models containing any of the senses of the 73 words to be disambiguated in the SENSEVAL-II test corpus (#models-senses-in-test). Finally, it appears their distribution, that is whether the sense information is placed on the head (#sense-head) or in a role (#sense-role). As expected, the sense distribution and coverage of the models obtained are different for SemCor and the SENSEVAL-II training corpus. While the models obtained from the SENSEVAL-II training corpus are distributed among all the senses in the test corpus, the models obtained from SemCor are associated to the most frequent senses and have a partial coverage of the senses involved in the SENSEVAL-II English Lexical Sample task. The 1,015 models obtained from SemCor for supervised-WSD not only are few but also cover few senses (126 different senses, 71 sense-head and 55 role-sense). That is less than 29% of 436 senses which appears in the test.

³<http://www.cs.unt.edu/~rada>

⁴Although target phrasal verbs are tagged

There are many differences between the LEXPIR subcategorization models used in the previous chapter and the models we obtained from sense-tagged corpora. Even their roles are fully disambiguated (e.g. the models obtained from SemCor), they are less rich than the LEXPIR models in the sense that they do not contain information about compulsory or optional roles.

I have observed^{observe#v#1} that being^{be#v#3} up^{up#r#1} on a horse^{horse#n#1} changes^{change#v#2} the whole^{whole#a#1} character^{character#n#3} of a man^{man#n#1}, and when a very^{very#r#1} small^{small#a#1} man^{man#n#1} is^{be#v#3} up^{up#r#1} on a saddle^{saddle#n#1}, he'd like^{like#v#1} as not prefer^{prefer#v#2} to eat^{eat#v#2} his meals^{meal#n#1} there^{there#r#1}

Figure VI.17: Sentence example from SemCor brown1/br-k09.xml p4 s9

Figure VI.17 shows a sentence from SemCor. The main difference from the SENSEVAL-II English Lexical Sample is that all the content word are disambiguated. Thus, most of the models are fully disambiguated. Figure VI.18 shows the model obtained for *eat#v#2* from the SemCor sentence above, where all the model items have a unique sense and provide access to the semantic information from the MCR.

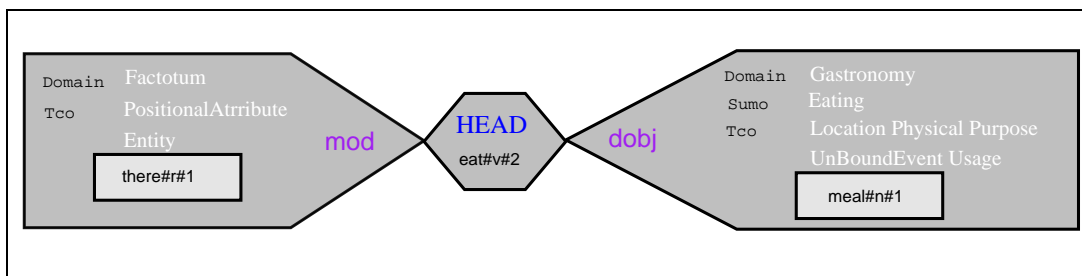


Figure VI.18: Extracting models from SemCor

VI.4.1.2 Collapsing models per sense

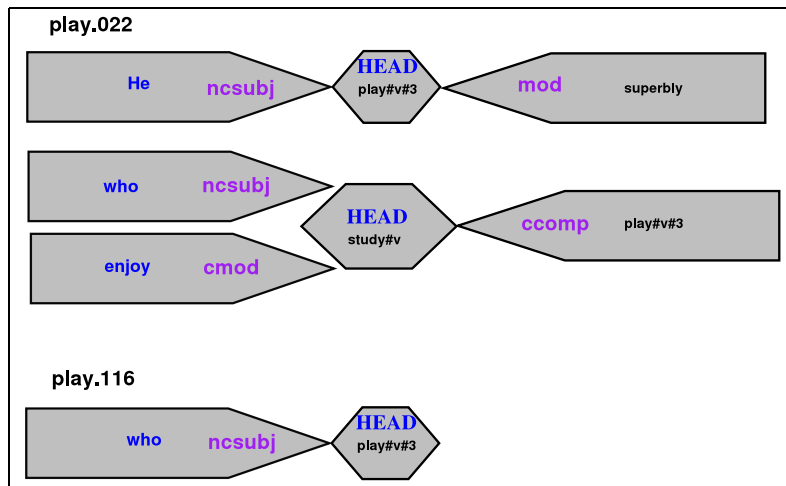
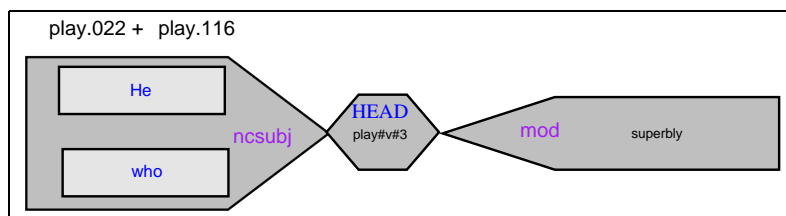
Since we are obtaining models automatically, we should cope with errors or missing syntactic dependencies coming from complicated sentences. Moreover, we have no clear clues of the relevance of a role in the model (e.g. adjuncts vs. arguments) or whether the models obtained belong or not to the same diathetical models.

Sense	Examples	Models	Head-Wsd	Role-Wsd
play#v#1	23	33	17	16
play#v#2	25	40	23	17
play#v#3	2	3	2	1
play#v#4	12	25	11	14
play#v#5	19	32	17	15
play#v#6	5	9	5	4
play#v#7	4	7	4	3
play#v#8	2	3	2	1
play#v#9	3	3	3	0
play#v#15	3	4	3	1
play#v#16	4	7	4	3
play#v#17	9	12	6	6
play#v#20	1	1	1	0
play#v#22	1	1	1	0
play#v#23	1	(2)	(1)	(1)
play#v#25	1	2	1	1
play#v#28	1	2	1	1
play#v#29	2	4	2	2
play#v#31	1	4	1	3
play_around#v#1	1	0	0	0
play_down#v#1	5	0	0	0
play_out#v#2	1	0	0	0
play_out#v#3	1	0	0	0
play_out#v#4	1	0	0	0
play_up#v#1	1	0	0	0

Table VI.3: Number of examples for *play* in the SENSEVAL-II training corpus

The numbers of models associated to a sense vary greatly not only among words but also for a given word as the training corpus reflects the sense frequency. For instance, table VI.3 shows the sense distribution of the models obtained from the SENSEVAL-II lexical sample training corpus for the verb *to play*⁵.

⁵The models for play#v#23 are inconsistent because the lemma is recognized as a MWE (play_round). We did not obtain any model for MWEs including play.

Figure VI.19: Models obtained for *play#v#3*Figure VI.20: Model obtained by joining all the models for *play#v#3*

The PARDON's formalization establishes that all the models are incompatible among them. In order to soften these problems, we joint all the models of a word sense in a single one. To illustrate this process, we will describe a simple example of joining two of the models shown in Figure VI.19 obtained from the training examples *play.022* and *play.116*. We will restrict the joining of the models to those with the same sense head (lemma, PoS and word sense). Thus, we can join the incomplete model obtained from *play.116* with the model obtained from *play.022* whose head is *play*. Figure VI.20 shows the model resulting from joining the two models. This join is carried out without generalization by simple adding the union of roles (which will be calculated as an *or*).

VI.5 Results

This section analyses the results using PARDON on the SENSEVAL-II English Lexical Sample task. First, pointing out some relevant issues about the task and its evaluation, then presenting the upper and lower bounds of the system, and finally present the results for the different experiments carried out.

VI.5.1 Senseval-II Evaluation Issues

There are several issues that can mislead our evaluation of the SENSEVAL-II English Lexical task:

- **MWEs:** The systems have to tokenize and recognise multiwords (most of them having low ambiguity figure). Given than for some test words the number of multiwords involved could be around 50%, the results of a system could be greatly affected by the preprocessing. Both because it could not recognize MWE senses and because it may happen that the system votes for MWE senses where there is not a MWE. It could penalize a good WSD system if it is unable to deal with MWEs. Notice that currently the preprocessing used in PARDON can not deal with discontinuous multiwords (e.g. phrasal verbs).
- **Special votes:** In the SENSEVAL-II English Lexical Sample task, the systems were allowed to vote for ProperNouns (P) and Unknown senses (U). The first could eventually increase the scores of a system with a good NE recognition (or viceversa), while the second could increase the noise in the system as ‘Unknown’ examples in the training could be ignored or not.
- **Inconsistencies in the training data:** There are a few inconsistencies in the data. For instance, there are few senses which appears in the test but not in the training data, senses which are not codified in WordNet1.7 or inconsistencies due to the American versus British spelling (e.g. colorless vs colourless). These inconsistencies are fully listed in appendix E.
- **Processing issues:** There are quite a few typographical tags in the training corpus (not always consistent) that difficults greatly the preprocessing (specially the syntactic analysis of the text).

VI.5.2 Baselines and Upper bounds

We established two different baselines based on the Most Frequent Sense. The frequency information will be used to determine the initial state in the relaxation labelling but using no model. Thus, the program converges to the most frequent senses. Table VI.4 shows these baselines using frequencies calculated using WordNet frequencies (MFS SemCor) or using the frequencies from the training data of the SENSEVAL-II English Lexical Sample task (MFS training).

	Fine			Coarse		
	P	R	F1	P	R	F1
PARDON-MFS Training	47.0	46.2	46.6	53.9	53.1	53.5
PARDON-MFS SemCor	40.8	40.1	40.4	50.0	49.0	49.5

Table VI.4: Baseline using MFS for SENSEVAL-II English Lexical task

We are also able to establish some upper bound for the system considering the existence of an applicable model or role of the correct sense. That is, checking for each test sentence, if there is at least an “applicable” role (supervised-role-WSD) or head (supervised-head-WSD).

On the one hand, regarding supervised-head-WSD, an upperbound can be calculated based on the existence of a model of the correct senses (without taking into account whether the model can be applied in the current sentence).

On the other hand, regarding supervised-role-WSD, since we require the objects to have the same syntactic relation and anchor than the role they are instantiating, an upper bound can be calculated based on the existence of a role corresponding to the right sense with the same syntactic relation-anchor. However, the real upper bound is even lower, as we are not taken into account the cases where the model to which the role belongs would be retrieved for the input sentence⁶.

Strategy	Upper Bound
head supervised	49% (2,122)
head (pre-process)	71% (3,081)
head-role supervised	60% (2,628)
head-role (pre-process)	87% (3,793)

Table VI.5: Upper Bounds using the SENSEVAL-II Training

Table VI.5 shows the different upper-bounds of PARDON using the models from the SENSEVAL-II training data. Using supervised strategies the system could not perform up to 60% (49% using only head-sense supervised strategies and 60% using only the role-sense supervised strategy). The limit when using unsupervised strategies it is far more complex as it will imply knowing whether we consider that two senses are similar. Thus, it could be only estimated in coarse grained manner on

⁶That is, if the head word of the model appears in the input sentence and it is also tagged with the same PoS

the base of some of the pre-processing information. Even considering that our system have a perfect semantic distance measure and that if there exists a role having same syntactic relation and anchor than the word to be disambiguated the semantic distance will select the correct sense, our system can not reach 87%.

VI.5.3 Results using the models acquired from English Lexical Sample training corpus

Using the models obtained from the English Lexical Sample training corpus, different experiments have been performed. First, we studied the impact of the integration of different knowledge (Knowledge Integration), second we compared the results obtained using different training corpora. We also compared the results obtained with the systems presented at the competition the SENSEVAL-II English Lexical Sample competition.

In all these experiments, we constrained 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 improvement of the results when using more semantic information.

Using the models obtained from the English Lexical Sample training corpus, the system is unable to decide a sense for 1,979 sentences out of 4,328. Although the general figures are quite far from the WSD systems that attends the SENSEVAL-II English Lexical Sample task (shown in table VI.21) which are around 64% in fine Precision and 71% in coarse Precision.

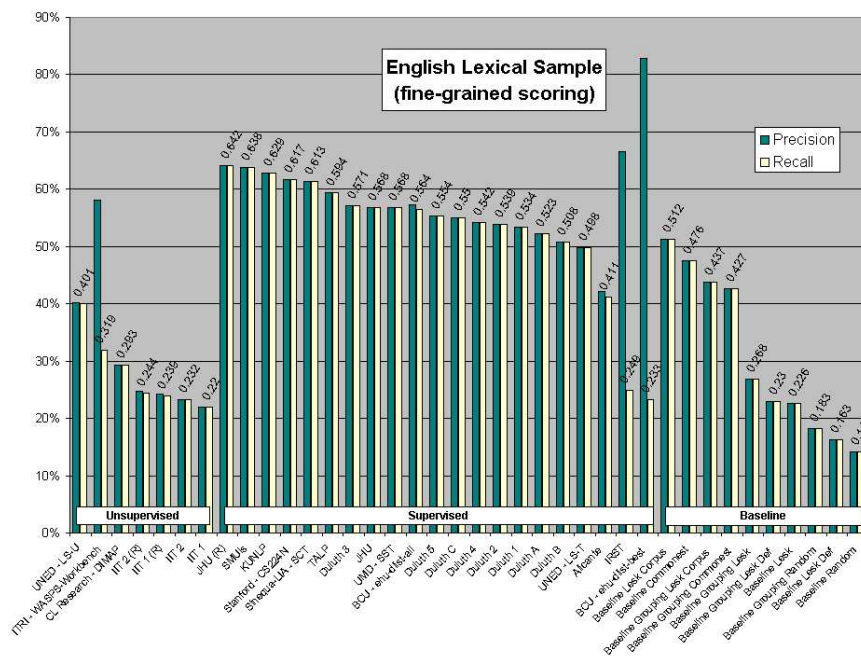


Figure VI.21: SENSEVAL-II English Lexical Task Results

VI.5.3.1 Impact of the Knowledge Integration

The first set of experiments aims to explore the impact of the integration of different semantic knowledge. Thus, different experiments have been performed, varying the level of semantic information used to determine the similarity between object and role. In order to stress the impact of the semantic information, for these experiments all the senses were instantiated with the same probabilities (PARDON-UFS) and without collapsing the models. *Head* and *Role* disambiguation strategies were used without generalization on the *anchor* neither the syntactic dependency was performed.

Semantics					Fine			Coarse		
LF	Wn	SUMO	Domain	TCO	P	R	F1	P	R	F1
					37.0	26.2	30.7	47.1	33.4	39.1
x					37.0	26.8	31.1	47.2	34.2	39.7
	x				38.6	27.4	32.0	48.7	34.7	40.5
		x			38.7	27.4	32.1	48.6	34.5	40.3
			x		42.5	29.8	35.0	52.5	36.9	43.3
				x	42.8	30.1	35.3	53.0	37.2	44.0
x	x				43.8	31.8	36.8	53.3	38.7	44.8
x	x	x			43.6	31.7	36.7	53.3	38.7	44.8
x	x	x	x		43.5	31.6	36.5	53.3	38.7	44.8
x	x	x	x	x	41.8	29.3	34.4	51.6	36.1	42.5

Table VI.6: Results for SENSEVAL-II English Lexical Sample task

Table VI.6 shows the figures obtained for the SENSEVAL-II English Lexical Sample task according to the official scorer for the different experiments. The first row in the table corresponds to the use of PARDON without any semantic information, the second row to the fifth row, correspond to the results using only one semantic resource: the Lexicographer File (**LF**), the hyperonym information from WordNet (**Wn**), the SUMO attribute, the Domain attribute and the Top Concept Ontology attribute (**TCO**). Finally from the sixth row to the ninth row show the results obtained when using more attributes incrementally.

The results show that combining different semantic attributes the system improves, although the combination of all the semantic attributes seems to perform worse than combining two or three attributes. This behaviour could be explained by the fact that most of these resources are not orthogonal. The fact that the coarse evaluation is almost the same when combining more than one semantic resource points that we are probably changing the votes between senses that are closer (coarse evaluation is the same). This also may point that the current system relies too much on the syntactic information. Whether we do not have enough examples of different syntactic behaviours of a word or there is a need of some more flexibility in the application of the syntactic constraints (that is, the preposition and syntactic dependency).

VI.5.3.2 Impact of the training corpora

In order to compare the impact of the models acquired from different corpora, we run the experiment using all the semantic attributes using the models obtained for SemCor and the models obtained for the SENSEVAL-II *English Lexical Sample*. Table VI.7 shows the results (**P**recision and **R**ecall) obtained for the SENSEVAL-II *English Lexical Sample* test using the models obtained from SemCor or the SENSEVAL-II training corpus respectively.

Models					
Senseval			SemCor		
P	R	F1	P	R	F1
41.8	29.3	34.4	28.3	15.9	20.4

Table VI.7: Results in Fine **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 51.6% of precision (41% using SemCor). We believe that this big difference 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 chooses the most similar model among all the applicable models).

This relatively poor figures are due to limitations of the current system (e.g. current system depends completely on having a dependency analysis, it is unable to detect discontinuous MWEs, etc.), limitation on the context of the sentence and limitation of the models used. Moreover, PARDON has a wider scope for WSD, in the current system all the WordNet senses are taking into account not only those senses which appear in the training data but all the WordNet senses. Senses which do not appear in the training data should not appear in the test data⁷. The lack of applicable supervised models, could drive PARDON to vote for senses which do not appear in the training data by means of unsupervised heuristics. Although it is quite an *ad-hoc* and unrealistic over-tune, it will be possible to restrict the set of senses taking into account for the target words.

We consider that the results obtained prove the feasibility of our approach, although they are slightly below the state-of-the-art of WSD. 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 in the experiments where obtained fully automatically.

⁷See the Appendix E for a list of some of the inconsistencies found in the test data

VI.5.3.3 Comparing with the SENSEVAL-II English lexical task participants

Table VI.8 shows a comparison per PoS of PARDON and some of the six best systems that participate in the SENSEVAL-II English Lexical Sample task.

	Verbs			Nouns			Adjectives		
	P	R	F1	P	R	F1	P	R	F1
JHU(R)	56.6	56.6	56.6	68.2	68.2	68.2	73.2	73.2	73.2
SMUIs	56.3	56.3	56.3	69.5	6.95	69.5	66.8	66.8	66.8
KUNLP	57.6	57.6	57.6	66.8	66.8	66.8	66.8	66.8	66.8
CS224n	52.3	52.3	52.3	68.3	68.3	68.3	61.6	61.7	61.6
Sinequa	53.5	53.5	53.5	63.3	63.3	63.3	66.4	66.4	66.4
Talp	51.3	51.3	51.3	65.5	65.5	65.5	64.5	64.5	64.5
PARDON-UFS	47.1	37.7	41.9	47.9	34.3	38.7	47.4	31.5	35.2
PARDON-MFS	46.4	40.1	43.0	55.7	53.6	54.6	60.7	54.2	57.3

Table VI.8: Results in **P**recision and **R**ecall for each PoS

For this comparison we use two versions of PARDON using the collapsed models. **PARDON-UFS** which initializes all the senses with the same initial probability and **PARDON-MFS training** which initializes the senses using the sense frequency of the SENSEVAL-II Lexical Sample task training corpus. **PARDON-MFS training** reaches a 52.9% Precision and 48.1% still far from the best WSD system.

PARDON seems to have similar figures for all the different PoS although we expect the results to work better on verbs. Moreover, we should take into account that we did not define a semantic distance for adverbs and that our current system is relatively poor-informed for adjectives.

However, when using Most Frequent Sense information, the results increase in precision and recall for adjectives and nouns. For verbs, it decreases minimally the precision although the recall improves.

Regarding the particularities of the SENSEVAL-II English lexical task, [Escudero Bakx, 2006] makes a deep study on the impact in the evaluation of the MWE, *Proper Nouns* and *Unknown* votes. As PARDON almost does not vote models for *ProperNouns* nor *Unknown senses*, we will only make a brief study on the MWE issue.

We divided the study between noun and adjective MWEs and phrasal verbs as the last ones are tagged on the test corpora. On one hand, table VI.9 shows the results for noun and adjectives MWEs. It can be seen that PARDON has a relatively low precision on MWE identification but an F1 on average. On the other hand, table VI.10 shows the results for phrasal verbs. Since PARDON does not use the information about phrasal verbs, the figures are much worst than the ones obtained by the other systems.

	P	R	F1	Ok	Att
JHU(R)	85.8	74.7	79.9	127	148
SMUls	88.8	68.0	77.0	151	222
KUNLP	53.9	48.4	51.0	83	154
CS224n	78.4	34.1	47.5	58	74
Sinequa	90.3	54.7	68.1	93	103
Talp	89.3	54.7	67.8	109	122
PARDON-MFS	67.3	57.7	62.1	101	150

Table VI.9: Results in **P**recision and **R**ecall for MWEs

	P	R	F1	Ok	Att
JHU(R)	66.5	65.4	65.9	119	179
SMUls	58.0	57.7	57.8	105	181
KUNLP	49.8	58.8	53.9	107	215
CS224n	43.0	26.9	33.1	49	114
Sinequa	59.5	42.9	49.8	78	131
Talp	45.5	39.0	42.0	71	156
PARDON-MFS	53.8	19.2	28.3	35	65

Table VI.10: Results in **P**recision and **R**ecall for Phrasal Verbs

VI.6 Discussion

We have shown that it is possible to develop a robust and flexible architecture for Semantic Role Labeling 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. PropBank or FrameNet).

In this chapter we presented an integrated architecture where both SRL and WSD tasks can 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 between SRL and WSD using richer semantic models, and second to improve the system itself (e.g. tuning the similarity functions, propagating semantic information, etc).

On the other hand, models obtained automatically 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 misleading semantic representation. The fundamental piece of information that a *dental surface* is also a *body-part* is not captured by our models obtained automatically. Conversely, more simple WSD systems, such as the ones using a bag of words, are able to capture and use that relation.

As a consequence the verb *clean* in this sentence will be wrongly disambiguated, as the models associated to *clean#v#3* (to clean a house) are the ones more related to clean a *surface*. On the other hand, the current prototype makes a shallow integration of the syntactic and semantic levels, causing the system to be sensitive to errors in the syntactic analysis. That is, being unable to disambiguate a word if a dependency analysis was not obtained for the input sentence.

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 to be able to distinguish between senses whose MCR representation is almost the same (e.g. the five senses of *child*).

Moreover, SemCor and SENSEVAL sentences are usually very complex. In our formalization syntax biases too much our model application, over-constraining the semantic generalization process. The current system is also unable to deal with diathesis or with syntactic structures that are not present in the models.

Using only the models obtained from SemCor, the results are hard to compare with other WSD systems, as our coverage of the models per sense is poor, and this has a great impact on the performance, as it can not be known in advance for which sentences we do not have models of the correct sense. On the other hand, combining the current model with other WSD heuristics, (e.g. Domain based WSD) could increase greatly our coverage.

However, either increasing the coverage of the models, or improving the WSD rates are out of the scope of this thesis. The results obtained are good enough to demonstrate that the PARDON architecture is applicable.

VI.6.1 What PARDON can not do

- To overcome errors from the preprocessing steps (e.g in the PoS, the lemmatization or the tokenization).
- To disambiguate senses from which the system do not have an example. Although the current version of PARDON can use unsupervised techniques based on examples from other words to solve unseen senses, that hardly happens when using the SENSEVAL-II Lexical Sample training data due to its small size and its biases.

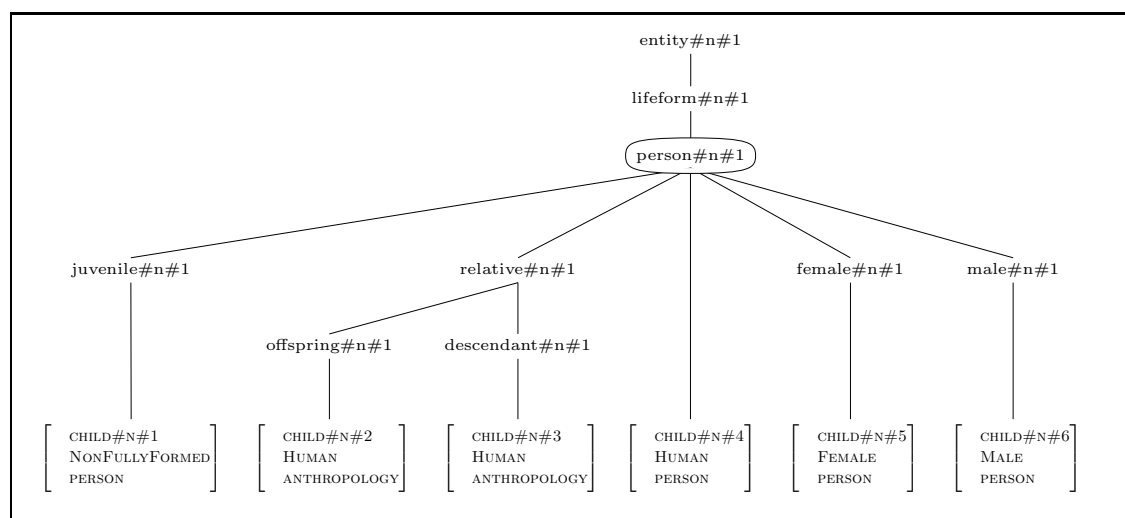


Figure VI.22: Different senses for the noun *child* in the WordNet hierarchy

- Disambiguate words whose syntactic behaviour does not vary and whose senses are similar according all the semantic information of MCR (that is, the WordNet hierarchy and all the different semantic resources, SUMO, TCO, Lexicographer Files, Domains).

For instance Figure VI.22 shows the six different senses of the noun *child* in the WordNet hierarchy and their associated semantic information. It can be seen that they almost share all the semantic attributes from MCR (that is, SUMO, Domains, etc) and its position in the WordNet hierarchy (all senses as descendant of *person#n#1* does not help much to disambiguate them. In most cases, discourse, extra-sentencial knowledge or statistical knowledge (e.g. frequency, co-occurrences) will be needed to disambiguate among these senses.

- The most important limitation of the currently prototype is that we only use information from words which have a direct syntactic connection. Thus, contextual information outside the sentence boundaries or even with elements which are not directly linked to the word to be disambiguated is not used. Consider the sentence example from the test corpus shown in Figure VI.23 where the semantic information carried by the word *film* could not help the

disambiguation of the word *play* as there isn't any direct syntactic relation between them.

...

As presented by Mr. Chabrol, and **<head>***played***</head>**
with thin-lipped intensity by Isabelle Huppert, Marie-Louise
(called Marie Latour in the *film*) was not a nice person.

...

Figure VI.23: Sentence example from SENSEVAL-II English Lexical Sample test (play.131)

All that could also explain the relatively poor performance of the WSD system for the SENSEVAL-II English lexical sample task. Although, our aim was to prove the feasibility of our approach, it is our believe that the PARDON framework can be enriched with more broad coverage heuristics.

CHAPTER VII.

Conclusions and Future Work

“All progress is precarious, and the solution of one problem brings us face to face with another problem.”

Martin Luther King Jr. (1929 - 1968) *“Strength to Love”*

“ Th...th...that’s all folks ! ”

The Porky pig

This thesis has explored a new integrated architecture for robust NLU, exploiting constraint-based optimization techniques. The goal of this work is to find robust and flexible architectures able to deal with the complexity of advanced NLP. We have successfully used the PARDON’s architecture using the relaxation labeling algorithm in two different NLP tasks, namely, SRL and WSD.

This chapter is organised in two sections. The first one summarises the contributions of this thesis and the second one outlines the future research directions outcoming from this work.

VII.1 Contributions

The main contributions of this thesis are:

VII.1.1 Proposing a novel NLP Architecture

We have proposed a novel architecture named PARDON, which is orthogonal to the traditional NLP task decomposition, and applies different types of knowledge (syntactic, semantic, linguistic, statistical) at the earliest opportunity but retaining an independent representation of the different kinds of knowledge. PARDON aims to give a general framework in which different NLP tasks can be formalized homoge-

neously. The framework allows different ways to perform NLP tasks: independently or simultaneously (following an integrated approach).

PARDON's architecture is based on the idea of *compositionality*. An element combines itself with other elements to build a new element. In most cases, the new element shares or contains the representation of the combined elements. Elements can not be freely combined. The correct combination of elements is determined by models and these models are associated to the initial elements.

The final goal of this architecture is to be flexible and robust using a frame-like semantic representation, as well as the compositional and pattern matching process of PARDON's architecture. PARDON's architecture can be formalized as a CLP, profiting from of the robust properties of the optimization techniques that could be applied to solve CLP problems.

VII.1.2 NLU Knowledge Integration

We have adopted a hybrid and simple approach. No claim of completion will be made. Different resources, knowledge repositories, are different views of the language. None of them can claim to cover completely the richness of the language. All these knowledge sources do not need to be either equivalent or even compatible as they will stand as independent information.

Different semantic resources, such as ontologies (Top Concept Ontology, SUMO), lexical databases (English WordNet, Spanish WordNet, Extended WordNet, different versions of Princeton WordNet), domain classifications, and sets of selectional preferences, have been uploaded to build a multilingual knowledge base based on the EuroWordNet structure, the Multilingual Central Repository (MCR).

PARDON's architecture gives us a natural way to integrate different knowledge sources, as a set of constraints inside a CLP, in order to solve different NLP tasks. The only condition required is that different knowledge sources may be related to each other (as it is inside the MCR through the ILI record). Despite the integration effort inside the MCR, since these different knowledge views are usually incompatible or contradictory, CLP will also give us a natural way to integrate them. Then, NLP tasks will be faced as an optimization problem, transforming the appropriate pieces of knowledge into a set of constraints and trying to find a solution that satisfies them, to a possible maximum degree.

VII.1.3 NLU Process Integration

PARDON's architecture is based on the idea of *compositionality*. An element combines itself with other elements so as to build a new element by means of models. PARDON proposes a formalization in a CLP framework which integrates the frame-like **Knowledge Representation**, the **Model Application** (that is the application of a model in isolation) and an **Inference Engine** which decides how to recursively apply the models.

Roughly speaking, PARDON combines objects from one level in order to build the objects corresponding to the next level of the task under consideration. The

resulting object is calculated simultaneously with the task of determining which models are to be applied to find the best solution (in a similar way to Hearst's *Polaroid Words* [Hirst, 1987]).

This integration of processes suffers from an explosion of possibilities to be explored due to its inherent combinatorial complexity. Amalgaming the search space and using optimization techniques such as relaxation labeling can soften this problem, but the complexity of the problem can not be avoided.

VII.1.4 Use of Optimization Techniques in NLU

The use of optimization techniques in spoken and written language processing has developed rapidly during the last years in conjunction with the statistical methods. Optimization methods are used to find the best solution among all the possible solutions by applying some evaluation criteria. Since the number of possible solutions can be large, the search needs to be highly efficient. In this thesis we have demonstrated that it is possible to use optimization techniques at a large scale in NLP.

VII.1.5 Robust NLU

Robustness has always been an important problem in NLP. Statistical methods are often presented as its only solution. However, in recent years, linguistic formalisms have also been aiming robustness using non-atomic information encoded in feature structures. Such fine-grained structures needs a relaxation in the unification mechanism. The subsequent growth in the search space is controlled by a selective form of success (not everything can be unified), and by measuring the 'goodness' of intermediate parsing results. The work in this thesis integrates successfully both approaches, statistical and linguistic, towards robustness.

VII.2 Further Work

VII.2.1 Regarding PARDON's Architecture

Regarding PARDON's Architecture, there are still many open issues to explore and test. The use of other optimization techniques to solve CLP/CSP (such as dynamic CSP), the integration of multiple levels of NLP, e.g. PoS and Parsing, or Parsing and WSD, or the application of PARDON to other NLP tasks: MWE detection, Parsing, Anaphora resolution, etc.

VII.2.2 Regarding PARDON as a Semantic Parser

Our approach to Semantic Parsing is basically a semantic role labeler. Further work should include a more realistic evaluation of the system, using a larger corpus with sentences having multiple verbs (maybe using models and corpus related to other

lexical resources available for English such as FrameNet or PropBank). In this case, verbal models would compete for their arguments in a sentence.

We also plan to include more statistical knowledge (measures/language models) and to extend the coverage and expressiveness of the subcategorization models. Exploiting the integration of other semantic resources related to Wordnet (e.g. the Multilingual Central Repository [Atserias et al., 2004f], developed inside the MEANING Project¹ [Rigau et al., 2002] which contains selectional preferences automatically acquired from corpora) is also among our plans for future work. Furthermore, the output of the current system could be also used to improve the existing verbal models.

Finally, the exploration of linguistic and statistical models for the identification/distinction of verbal adjuncts should also be investigated, since it seems to be one of the main causes of verbal argument mis-identification.

VII.2.3 Regarding PARDON as a Word Sense Disambiguator

Future lines of research in this regard include, first, extending the level of integration of *Semantic Parsing* and *Word Sense Disambiguation* using richer semantic models, and second, improving the system itself (e.g. tuning the similarity functions, propagating semantic information, etc).

The most important limitation of the current prototype is that, in order to disambiguate a word, we only use information from words which have a direct syntactic connection with it. Thus, contextual information outside the sentence boundaries or even with elements which are not directly linked to the word to be disambiguated is not used. A short-term future work will combine other WSD methods (e.g. WSD based on Domains [Magnini and Strapparava, 2000]) with the PARDON framework to overcome this problem.

The current PARDON prototype for WSD could be also used to explore unsupervised WSD and related issues such as how to integrate supervised and non-supervised models or which models should be activated for a given word.

¹<http://www.lsi.upc.es/~nlp/meaning/meaning.html>

Bibliography

Steven P. Abney. 1991. Parsing by Chunks. In *Principle-Based Parsing: Computation and Psycholinguistics*, Boston, MA. Kluwer Academic Publishers.

Eneko Agirre and David Martinez. 2000. Exploring automatic word sense disambiguation with decision lists and the web. In *Proceedings of the COLING workshop on Semantic Annotation and Intelligent Annotation*, Luxembourg.

Eneko Agirre and David Martinez. 2001. Learning class-to-class selectional preferences. In *Proceedings of Conference on Computational Natural Language Learning (CoNLL'2001)*, Toulouse, France.

Eneko Agirre and David Martinez. 2002. Integrating selectional preferences in WordNet. In *Proceedings of the first International WordNet Conference in Mysore, India*, 21-25 January.

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

Eneko Agirre, Olatz Ansa, Xavier Arregi, J.M. Arriola, Arantza Diaz de Ilarraza, Eli Pociello, and L. Uria. 2002. Methodological issues in the building of the basque wordnet: quantitative and qualitative analysis. In *Proceedings of the first International Global WordNet Conference (GWC'2002)*, Mysore, India, 21-25 January.

J. Agnese, N. Bataille, E. Bensana, D. Blumstein, and G. Verfaillie. 1995. Exact and Approximate methods for the daily management of Earth observation satellite. In *Proceedings of the 5th ESA Workshop on Artificial Intelligence and Knowledge based System for Space*, The Netherlands.

Enrique Alfonseca and Suresh Manandhar. 2002. An unsupervised method for general named entity recognition and automated concept discovery. In *Proceedings of the 1st International Global WordNet Conference (GWC'2002)*, Mysore, India, 21-25 January.

James Allen. 1995. *Natural Language Understanding*. Benjamin/Cummings Publishing Company Inc., 2nd edition.

Hiyan Alshawi, David Carter, Richard Crouch, Steve Pulman, Manny Rayner, and Arnold Smith. 1992. Clare: A contextual reasoning and cooperative response framework for the core language engine. Technical report, SRI International, Cambridge Computer Research Centre.

Hiyan Alshawi. 1990. Resolving quasi logical forms. *Computational Linguistics*, 16(3):133–144.

Hiyan Alshawi. 1992. *The Core Language Engine*. ACL-MIT press.

Jan W. Amtrup. 1998. Incremental Speech Translation: A Layered Chart Approach. In *Proceedings of the 28th Jahrestagung der Gesellschaft für Informatik*.

D. Appelt, J. Hobbs, J. Bear, D. Israel and M. Kameyama, A. Keheler, D. Martin, K. Myers, and M. Tyson. 1996. SRI International FASTUS System:MUC-6 Test Results and Analysis. In *Proceedings of the 6th MUC Conference*.

Shlomo Argamon, Ido Dagan, and Yuval Krymoloswky. 1998. A Memory-Based Approach to Learning Shallow Natural Language Patterns. In *Proceedings of the 17th International Conference on Computational Linguistics (COLING'1998)*. URL: <http://www.lanl.gov/abs/cmp-lg/9806011>.

Victoria Arranz, Jordi Atserias, and Mauro Castillo. 2005. Multiword Expressions and Word Sense Disambiguation. In Alexander Gelbukh, editor, *Proceedings of the Conference on Intelligent Text Processing and Computational Linguistics (CicLing'2005)*, volume LNCS 3406 of *Lecture Notes in computer Science*. Springer. ISSN 0302-9743 ISBN 3-540-24523.

Jordi Atserias, Salvador Climent, Xavier Farreres, German Rigau, and Horacio Rodríguez. 1997. Combining Multiple Methods for the Automatic Construction of Multilingual WordNets. In *Recent Advances in Natural Language (RANLP'1997)*, pages 143–149, Tzgov Chark, Bulgaria.

Jordi Atserias, Irene Castellón, Montse Civit, and German Rigau. 1999. Using Diathesis for Semantic Parsing. In *Venecia per il Trattamento automatico delle lingue (VEXTAL'99)*, pages 385–392, Venice, Italy. Unipress. ISBN: 88-8098-1-12-9.

Jordi Atserias, Irene Castellón, Montse Civit, and German Rigau. 2000. Semantic analysis based on verbal subcategorization. In *Proceedings of the Conference on Intelligent Text Processing and Computational Linguistics (CicLing'2000)*.

Jordi Atserias, Salvador Climent, and German Rigau. 2004a. Towards the meaning top ontology: Sources of ontological meaning. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC'2004)*, Lisbon, Portugal.

Jordi Atserias, Montse Cuadros, Eva Naqui, and German Rigau. 2004b. D4.3 PORT2. Technical report, MEANING project.

Jordi Atserias, Montse Cuadros, Eva Naqui, and German Rigau. 2004c. WP4.6 UPLOAD2. Technical report, MEANING project.

Jordi Atserias, Eva Naqui, Montse Cuadros, German Rigau, and Samir Kanaan. 2004d. WP4.4 UPLOAD1. Technical report, MEANING project.

Jordi Atserias, German Rigau, and Luis Villarejo. 2004e. WP4.1 UPLOAD0. Technical report, MEANING project.

Jordi Atserias, Luis Villarejo, German Rigau, Eneko Agirre, John Carroll, Bernardo Magnini, and Piek Vossen. 2004f. The MEANING multilingual central repository. In *Proceedings of the Second International Global WordNet Conference (GWC'04)*, Brno, Czech Republic, January. ISBN 80-210-3302-9.

Jordi Atserias, Salvador Climent, German Rigau, and Joaquim Moré. 2005. A Proposal for a Shallow Ontologization of Wordnet. In *XXI Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN'2005)*, pages 161–167, Granada, Spain. ISSN 1135-5948.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *Proceedings of the International Conference on Computational Linguistics (COLING/ACL'1998)*, Montreal, Canada.

Roberto Basili, Maria Tereza Pazienza, and Paola Velardi. 1996. An empirical symbolic approach to natural language processing. *Artificial Intelligence*, (85):59–99.

Roberto Basili, Maria Teresa Pazienza, and Fabio Massimo Zanzotto. 1998. Efficient parsing for information extraction. In Henri Prade, editor, *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI-98)*, pages 135–139, Chichester, August 23-28. John Wiley & Sons Ltd.

Stephen Beale and Sergei Nirenburg. 1995. Dependency-directed text planning. In *Proceedings of the International Joint Conference in Artificial Intelligence (IJCAI'1995)*, Montreal, Canada.

Stephen Beale, Sergei Nirenburg, and Kavi Mahesh. 1996. HUNTER-GATHERER: Three Search Techniques Integrated for Natural Language Semantics. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI'1996)*, Portland, Oregon.

Stephen Beale. 1996. *Hunter-Gatherer: Applying Constraint Satisfaction, Branch-and-Bound and Solution Synthesis to computational Semantics*. Ph.D. thesis, Computer Research Laboratory, New Mexico State University, Las Cruces, New Mexico.

Laura Benítez, Sergi Cervell, Gerard Escudero, Monica López, German Rigau, and Mariona Taulé. 1998. Methods and Tools for Building the Catalan WordNet. In *Proceedings of the ELRA Workshop on Language Resources for European Minority*

Languages, First International Conference on Language Resources and Evaluation (LREC'1998), Granada, Spain.

Luisa Bentivogli, Emanuele Pianta, and Marcello Ranieri. 2005. Multisemcor: an english italian aligned corpus with a shared inventory of senses. In *Proceedings of the 2nd Meaning Workshop*, page 90, Trento, Italy, February.

Lawrence Birnbaum. 1989. A Critical Look at the Foundations of Autonomous Syntactic Analysis. In *Proceedings of the 11th Annual Conference of the Cognitive Science Society*, pages 99–106.

Christian Boitet and Mark Seligman. 1994. The 'WHITEBOARD' Architecture: A Way to Integrate Heterogeneous Components of NLP Systems. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING'1994)*.

Kalina Bontcheva and Yorick Wilks. 2001. Dealing with dependencies between content planning and surface realisation in a pipeline generation architecture. In *Proceedings of the International Joint Conference in Artificial Intelligence (IJCAI'2001)*, Seattle, USA.

James E. Borrett and Edward P.K. Tsang. 1996. Towards a formal framework for comparing constraint satisfaction problems formulations. Technical Report CSM-264, Dept. of computer Science, University of Essex.

Michael Richard Brent. 1991. Automatic acquisition of subcategorization frames from untagged text. In *Proceedings of 29th annual meeting of the Association for Computational Linguistics (ACL'1991)*, Berkeley, CA.

Michael Richard Brent. 1993. From grammar to lexicon: unsupervised learning of lexical syntax. *Computational Linguistics*, 19(2):243 – 262.

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

Ted Briscoe and John Carroll. 1997. Automatic Extraction of Subcategorization from Corpora. In *Proceedings of 5th Conference on Applied Natural Language Processing*, pages 356 – 363, Washington DC, USA.

Sabine Buchholz. 1998. Distinguishing Complements from Adjuncts using Memory-Based Learning. In *Proceedings of the ESSLLI-98 Workshop on Automated Acquisition of Syntax and Parsing*.

Lynne Cahill, Christy Doran, Roger Evans, Chris Mellish, Daniel Paiva, Mike Reape, and Donia Scott. 1999a. Achieving theory-neutrality in reference architectures for nlp: To what extent is it possible/desirable? *Proceedings of the AISB'99 workshop on reference architectures and data standards for NLP*, pages 32–35.

Lynne Cahill, Evans R, Mellish C, Reape M, and Scott D. 1999b. Towards a Reference Architecture for Natural Language Generation Systems - The RAGS project. Technical Report 2000 8/1999, Technical Report. Brighton: Information Technology Research Institute.

Lynne Cahill, John Carroll, Roger Evans, Daniel Paiva, Richard Power, Donia Scott, and Kees Deemter. 2001a. From RAGS to RICHES: Exploiting the Potential of a Flexible Generation Architecture. In *Proceeding of ACL'2001*, pages 98–105. <http://www.aclweb.org/anthology/P01-1015>.

Lynne Cahill, John Carroll, Roger Evans, Daniel Paiva, Richard Power, Donia Scott, and Kees van Deemter. 2001b. From RAGS to RICHES: exploiting the potential of a flexible generation architecture. In *Proceedings of ACL/EACL 2001*, pages 98–105, Toulouse, France.

Charles Callaway. 2003. Integrating Discourse Markers into a Pipelined Natural Language Generation Architecture. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL'2003)*, pages 264–271, Sapporo, Japan, July.

R. Carpenter. 1992. *The Logic of Typed Feature Structures*. Cambridge University Press.

John Carroll and Diana McCarthy. 2000. Word sense disambiguation using automatically acquired verbal preferences. *Computers and the Humanities. Senseval Special Issue*.

Glenn Carroll and Mats Rooth. 1998. Valence induction with a head-lexicalized pcfg. In *Proceedings of the 3rd conference on empirical methods in natural language processing (EMNLP 3)*, Granada.

John Carroll, Guido Minnen, and Ted Briscoe. 1998. Can subcategorisation probabilities help a statistical parser? In *Proceedings of the 6th ACL/SIGDAT Workshop on Very Large Corpora*, pages 118–126, Montreal, Canada.

John Carroll, Guido Minnen, and Ted Briscoe. 1999. Corpus annotation for parser evaluation. In *Proceedings of EACL'99 Workshop on Linguistically Interpreted corpora, LINC'99*. Bergen, Norway.

Irene Castellón, Montse Civit, and Jordi Atserias. 1998. Syntactic Parsing of Spanish Unrestricted Text. In *Proceedings of the 1th Conference on Language Resources and Evaluation (LREC'1998)*, Granada. Spain.

Joyce Yue Chai and Alan W. Biermann. 1997. The Use of Lexical Semantics in Information Extraction. In *Proceedings of the ACL Workshop on automatic IE and building of Lexical Semantic Resources*, Madrid, Spain.

Eugene Charniak, Glenn Carroll, John Adcock, Anthony R. Cassandra, Yoshihiko Gotoh, Jeremy Katz, Michael L. Littman, and John McCann. 1996. Taggers for parsers. *Artificial Intelligence*, 85(1-2):45–57.

Fabio Ciravegna and Nicola Cancedda. 1995. Integrating shallow and linguistic techniques for information extraction from text. In *AI*IA: Proceedings of the Congress of the Italian Association for Artificial Intelligence on Trends in Artificial Intelligence*, pages 127–138.

Fabio Ciravegna and Alberto Lavelli. 1997. Controlling Bottom-Up Chart Parsers through Text Chunking. In *Proceedings of the 5th International Workshop on Parsing Technologies (IWPT'1997)*, Boston.

Scott Cost and Steven Salzberg. 1993. A weighted nearest neighbor algorithm for learning with symbolic features. *Machine Learning*, 10(1):57–78.

Dan Cristea, Georgiana Puscasu, Oana Postolache, Eleni Galiotou, Maria Grigoriadou, Anastasia Charcharidou, Evangelos Papakitsos, Stathis Selimis, Sofia Stamou, Cvetana Krstev, Gordana Pavlovic-Lazetic, Ivan Obradovic, Dusko Vitas, Ozlem Cetinoglu, Dan Tufis, Karel Pala, Tomas Pavelek, Pavel Smrz, Svetla Koeva, and George Totkov. 2003. Tracing of the common base concepts. Technical Report D.4.2, WP4.

B. Crysmann, A. Frank, B. Kiefer, H. Krieger, S. Muller, G. Neumann, J. Piskorski, U. Schafer, M. Siegel, H. Uszkoreit, and F. Xu. 2002. An Integrated Architecture for Shallow and Deep Processing. In *Proceedings of the 40th Conference of the Association for Computational Linguistics (ACL'2002)*, Philadelphia, USA, July 7-12.

Montse Cuadros, Lluís Padró, and German Rigau. 2006. An empirical study for automatic acquisition of topic signatures. In *Proceedings of Third International WordNet Conference*, pages 51–59, Jeju Island (Korea). ISBN 80-210-3915-9.

Hamish Cunningham, Yorick Wilks, and Robert Gaizauskas. 1996. GATE- A General Architecture for Text Engineering. In *Proceedings of the International Conference on Computational Linguistics (COLING'1996)*, pages 1057–1060.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, and Valentin Tablan. 2002. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Conference of the Association for Computational Linguistics (ACL'2002)*.

Ido Dagan, Lillian Lee, and Fernando C. N. Pereira. 1999. Similarity-based models of word cocurrence probabilities. *Machine Learning*, (34):43–69.

Jordi Daudé, Lluís Padró, and German Rigau. 1999. Mapping Multilingual Hierarchies Using Relaxation Labeling. In *Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC'99)*, Maryland, US.

- Jordi Daudé, Lluís Padró, and German Rigau. 2000. Mapping WordNets Using Structural Information. In *Proceedings of 38th annual meeting of the Association for Computational Linguistics (ACL'2000)*, Hong Kong.
- Jordi Daudé, Lluís Padró, and German Rigau. 2001. A Complete WN1.5 to WN1.6 Mapping. In *Proceedings of NAACL Workshop 'WordNet and Other Lexical Resources: Applications, Extensions and Customizations'*, Pittsburg, PA, United States.
- Jordi Daudé. 2005. *Enlace de Jerarquías Usando el Etiquetado por Relajación*. Ph.D. thesis, Dept. de LSI, Universitat Politècnica de Catalunya, July.
- Michael Daum, Kilian A.Foth, and Wolfgang Menzel. 2002. Constraint Based Integration of Deep and Shallow Parsing Techniques. In *Proc. 11th Conference of the European Chapter of the Association for Computational Linguistics, EACL*, Budapest.
- Michael Daum. 2004. Dynamic dependency parsing. In *Proceedings of the ACL Workshop on Incremental Parsing*.
- Bart Decadt, Véronique Hoste, Walter Daelemans, and Antal van den Bosch. 2004. GAMBL, Genetic Algorithm Optimization of Memory-Based WSD. In R. Mihalcea and P. Edmonds, editors, *Proceedings of the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text (SENSEVAL-III)*, pages 108–112, Barcelona, Spain, July. Proceedings of ACL'2004.
- Bonnie Dorr. 1993a. Interlingual machine translation: a parameterized approach. *Artificial Intelligence*, 63(1&2):429–492.
- Bonnie Dorr. 1993b. A view from the lexicon. *Machine Translation*. MIT Press.
- Bonnie Dorr. 1997. Large- Scale Acquisition of LCS- Based Lexicons for Foreign Language Tutoring. In *Proceedings of the ACL Fifth Conference on Applied Natural Language Processing (ANLP'1997)*, pages 139–146.
- D. Dowty. 1991. Thematic proto-roles and argument selection. *Language*, 67(3):547–619.
- T. Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational linguistics*, 19(1):61–74.
- Lee Erman, Rick Hayes-Roth, Victor Lesser, and Raj Reddy. 1980. The hearsay-ii speech understanding system. In *First National Conference of the American Association of Artificial Intelligence (AAAI'1980)*.
- Gerard Escudero Bakx. 2006. *Machine Learning Techniques for Word Sense Disambiguation*. Ph.D. thesis, Departament de Llenguatges i Sistemes Informàtics. Universitat Politècnica de Catalunya. forthcoming.

- Eva Esteve Ferrer. 2004. Towards a Semantic Classification of Spanish Verbs Based on Subcategorisation Information. In Daniel Midgley Leonoor van der Beek, Dmitriy Genzel, editor, *Proceedings of the ACL Student Research Workshop*, pages 37–42, Barcelona, Spain, July. Association for Computational Linguistics.
- Collin F. Baker and Josef Ruppenhofer. 2002. FrameNet’s Frames vs. Levin’s Verb Classes. In *Proceedings of the 28th Annual Meeting of the Berkeley Linguistics Society*, February 15-18.
- Christiane Fellbaum, editor. 1998. *WordNet. An Electronic Lexical Database*. The MIT Press.
- Ana Fernández and Maria Antonia Martí. 1996. Classification of psychological verbs. *Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN’1996)*, (20).
- Ana Fernández, Maria Antonia Martí, Gloria Vázquez, and Irene Castellón. 1999. Establishing semantic oppositions for typification of predicates. *Language Design*, (2).
- Charles J. Fillmore. 1968. The case for case. *Bach and Harms (Ed.): Universals in Linguistic Theory*, pages 1–88. Holt, Rinehart, and Winston, New York.
- Kilian Foth, Wolfgang Menzel, and Ingo Schröder. 2003. Robust parsing with weighted constraints. to appear in *Natural Language Engineering*.
- E. Franconi. 2002. Description logics for natural language processing. In *Description Logics Handbook*, pages 460–471. Cambridge University Press, January.
- S. Gahl. 1998. Automatic extraction of subcorpora based on subcategorization frames from a part-of-speech tagged corpus. In *Proceedings of 36th annual meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (COLING/ACL’1998)*, Montreal, Canada.
- N. Ge, J. Hale, and E. Charniak. 1998. A statistical approach to anaphora resolution. In *Proceedings of the Sixth ACL/SIGDAT Workshop on Very Large Corpora*, pages 161–171.
- Daniel Gildea and Julia Hockenmaier. 2003. Identifying semantic roles using combinatory categorial grammar. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP’2003)*, pages 57–64. Sapporo, Japan.
- Daniel Gildea and Daniel Jurafsky. 2000. Automatic Labeling of Semantic Roles. In *Proceedings of the 38th Annual Conference of the Association for Computational Linguistics (ACL’2000)*, pages 512–520, Hong Kong, October.
- Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

Daniel Gildea and Martha Palmer. 2002. The Necessity of Syntactic Parsing for Predicate Argument Recognition. In *Proceedings of the 40th Conference of the Association for Computational Linguistics (ACL'2002)*, Philadelphia, PA.

A-M Giuglea and A Moschitti. 2004. Knowledge discovery using framenet, verbnet and propbank. In *Proceedings of the Workshop on Ontology and Knowledge Discovering at ECML*.

Fernando Gomez, Carlos Segami, and Richard Hull. 1997. Determining prepositional attachment, prepositional meaning, verb meaning and thematics roles. *Computational Intelligence*, 13(1).

Fernando Gomez. 1998. Linking WordNet Verb Classes to Semantic Interpretation. In *Proceedings of the COLING-ACL Workshop on the Usage of WordNet on NLP Systems*, Universite de Montreal, Quebec, Canada, August.

Fernando Gomez. 2001. An Algorithm for Aspects of Semantic Interpretation Using an Enhanced WordNet. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL'2001)*, CMU, Pittsburgh.

Ralph Grishman and John Sterling. 1994. Generalizing automatically generated selectional patterns. In *Proceedings of the International Conference on Computational Linguistics (COLING'1994)*.

Ralph Grishman. 1995. Nyu system or where's the syntax? In *MUC-6*, pages 167–175.

Nicola Guarino and Christopher A. Welty. 2000. A formal ontology of properties. In *Proceedings of ECAI'2000 Workshop on Knowledge Acquisition, Modeling and Management*, pages 97–112.

Sanda Harabagiu and Steven Maiorano. 1999. Finding answers in large collections of texts: Paragraph indexing + abductive inference. In *Proceedings of the National Conference on Artificial Intelligence (AAAI'1999) Fall Symposium on Question Answering Systems*, pages 63–71.

Sanda Harabagiu and Dan Moldovan. 1998. Knowledge processing on extended wordnet. In *WordNet: An Electronic Lexical Database and Some of its Applications*, Editor C. Fellbaum. MIT Press.

D. Hindle and M. Rooth. 1993. Structural ambiguity and lexical relations. *Computational linguistics*, 19(2):103–120.

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

- Véronique Hoste, Anne Kool, and Walter Daelemans. 2001. Classifier Optimization and Combination in the English All Words Task. In *Proceedings of the Second International Workshop on Evaluating Word Sense Disambiguation Systems (Senseval-II)*, pages 83–86, Toulouse, France.
- K. Humphreys, R. Gaizauskas, S. Azzam, C. Huyck, B. Mitchell, H. Cunningham, and Y. Wilks. 1998. University of Sheffield: Description of the LaSIE-II System as Used for MUC-7. In *Proceedings of the MUC-7*.
- R Jackendoff. 1972. *Semantic Interpretation in Generative Grammar*. Cambridge, Massachusetts, The MIT Press.
- Daniel Jurafsky. 1992. An On-Line Computational Model of Human Sentence Interpretation. In *Proceedings of the National Conference on Artificial Intelligence (AAAI'1992)*, pages 302–308.
- J. Justeson and S. Katz. 1995. Principled disambiguation: discriminating adjective senses with modified nouns. *Computational Linguistics*, 21(1):1–28.
- H. Kamp and U. Reyle. 1993. *From Discourse to Logic: Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Kluwer, Dordrecht.
- R. Kibble and R. Power. 2000. An integrated framework for text planning and pronominalisation. In *Proceedings of the International Conference in Natural Language Generation (INLG'2000)*, pages 77–84, Mitzpe Ramon, Israel.
- Adam Kilgarriff. 1997. Foreground and background lexicons and word sense disambiguation for information extraction. In *Proceedings of the Workshop on Lexicon Driven Information Extraction*, Frascati, Italy.
- Adam Kilgarriff. 1998. SENSEVAL: An exercise in evaluating word sense disambiguation programs. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC'1998)*, pages 581–588, Granada, Spain.
- Paul Kingsbury and Martha Palmer. 2002. From Treebank to PropBank. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC'2002)*, Las Palmas, Spain.
- Paul Kingsbury, Martha Palmer, and Mitch Marcus. 2002. Adding semantic annotation to the penn treebank. In *Proceedings of the Human Language Technology Conference*.
- Karin Kipper, Hoa Trang Dang, and Martha Palmer. 2000. Class-based construction of a verb lexicon. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence, (AAAI'2000)*.
- Karin Kipper, Martha Palmer, and Owen Rambow. 2002. Extending propbank with verbnet semantic predicates. In *Proceedings of AMTA'2002*.

- Jeffrey D. Kirtner and Steven L. Lytinen. 1991. ULINK: A Semantics-Driven Approach to Understanding Ungrammatical Input. In *Proceedings of the National Conference on Artificial Intelligence (AAAI'1991)*, pages 137–142.
- Anna Korhonen. 1998. Automatic extraction of subcategorization frames from corpora - Improving filtering with diathesis alternations. In *Proceedings of the ESSLLI Workshop on Automated Acquisition of Syntax and Parsing*, Saarbrücken, Germany.
- Anna Korhonen. 2002. *Subcategorization acquisition*. Ph.D. thesis, University of Cambridge.
- Maria Lapata. 1993. Acquiring lexical generalizations from corpora: A case study for diathesis alternations. In *Proceedings of 37th annual meeting of the Association for Computational Linguistics (ACL'1999)*, College Park, Maryland.
- Maria Lapata. 2001. *The Acquisition and Modeling of Lexical Knowledge: A Corpus-based Investigation of Systematic Polysemy*. Ph.D. thesis, University of Edinburgh.
- Alberto Lavelli and Bernardo Magnini. 1991. Lexical discrimination within a multilevel semantics approach. In *AI*IA: Proceedings of the 2nd Congress of the Italian Association for Artificial Intelligence on Trends in Artificial Intelligence*, pages 455–459. Springer-Verlag.
- Claudia Leacock and Martin Chodorow, 1998. *Combining Local Context and WordNet Similarity for Word Sense Identification*, chapter 11, pages 265–283. MIT Press.
- Claudia Leacock, Martin Chodorow, and George Miller. 1998. Using Corpus Statistics and WordNet Relations for Sense Identification. *Computational Linguistics*, 24(1):147–166.
- Jochen L. Leidner. 2003. Current issues in software engineering for natural language processing. In *Proceedings of the Workshop on Software Engineering and Architecture of Language Technology Systems (SEALTS) held at the Joint Conference for Human Language Technology and the Annual Meeting of the North American Chapter of the Association for Computational Linguistics 2003 (HLT/NAACL'03)*, pages 45–50, Edmonton, Alberta, Canada, May.
- Beth Levin. 1993. *English Verb Classes and Alterations: A preliminar Investigation*. The University of Chicago Press.
- Hang Li and Naoki Abe. 1995. Generalizing case frames using thesaurus and the mdl principle. In *Proceedings of the International Conference "Recent Advances in Natural Language Processing" (RANLP'1995)*, pages 239–248.
- Hang Li and Naoki Abe. 1998. Generalizing Case Frames Using a Thesaurus and the MDL Principle. *Computational linguistics*, 24(2):217–244.

- Dekang Lin and Patrick Pantel. 1994. Concept Discovery from Text. In *15th International Conference on Computational Linguistics (COLING'2002)*, Taipei, Taiwan.
- Dekang Lin. 1997. Using Syntactic Dependency as Local Context to Resolve Word Sense Ambiguity. In *Proceedings of Conference of the Association for Computational Linguistics (ACL'1997)*, Madrid, Spain.
- Dekang Lin. 1998. An Information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning*, pages 296–304.
- Kenneth Litkowski. 2000. Question-answering using semantic relation triples. In E.M. Voorhess and D.K Harman, editors, *Information Technology: The Eighth Text REtrieval Conference (TREC-8)*, pages 349–356. NIST Special Publication 500-246.
- Kenneth Litkowski. 2001a. SENSEVAL: The CL Research Experience. *Computer and the Humanities*, 34(1–2):153–158.
- Kenneth Litkowski. 2001b. SENSEVAL Word-Sense Disambiguation Using a Different Sense Inventory and Mapping to WordNet. In *Proceedings of the Second International Workshop on Evaluating Word Sense Disambiguation Systems (Senseval-II)*.
- Steven L. Lytinen. 1986. Dynamically Combining Syntax and Semantics in Natural Language Processing. In *Proceedings of the National Conference on Artificial Intelligence (AAAI'1986)*, pages 574–587.
- Bernardo Magnini and Gabriela Cavaglia. 2000. Integrating subject field codes into wordnet. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'2000)*, Athens, Greece.
- Bernardo Magnini and Carlo Strapparava. 2000. Experiments in word domain disambiguation for parallel texts. In *Proceedings of the ACL Workshop on Word Senses and Multilinguality*.
- Kavi Mahesh. 1993. A theory of interaction and independence in sentence understanding. Master's thesis, Georgia Institute of Technology.
- Kavi Mahesh. 1995. Syntax-semantics interaction in sentence understanding. Technical Report GIT-CC-95-10.
- Christopher D. Manning. 1993. Automatic acquisition of a large subcategorization dictionary from corpora. In *Proceedings of 31th annual meeting of the Association for Computational Linguistics (ACL'1993)*, Columbus, Ohio.
- Lluís Màrquez, Gerard Escudero, David Martínez, and German Rigau, 2006. chapter Supervised Corpus-based Methods for Word Sense Disambiguation. Algorithms and Applications. Kluwer, Eneko Agirre and Phil Edmonds (Eds.). Due out in June 2006 (preprint available).

- David Martinez. 2004. *Supervised Word Sense Disambiguation: Facing Current Challenges*. Ph.D. thesis, Euskal Herriko Universtsitea.
- Diana McCarthy and Anna Korhonen. 1997. Detecting verbal participation in diathesis alternations. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguists (ACL'1997)*, volume 2, pages 1493–1495, Montreal.
- Diana McCarthy, John Carroll, and Judita Preiss. 2001. Disambiguating noun and verb senses using automatically acquired selectional preferences. In *Proceedings of the SENSEVAL-2 Workshop at ACL/EACL'2001*, Toulouse, France.
- Diana McCarthy. 1997. Word sense disambiguation for acquisition of selectional preferences. In *Proceedings of the ACL/EACL Workshop on Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 52–61.
- Diana McCarthy. 2000. Using semantic preferences to identify verbal participation in role switching alternations. In *Proceedings of the first Conference of the North American Chapter of the Association for Computational Linguistics. (NAACL'2000)*, Seattle, WA.
- Diana McCarthy. 2001. *Lexical Acquisition at the Syntax-Semantics Interface: Diathesis Alternations, Subcategorization Frames and Selectional Preferences*. Ph.D. thesis, University of Sussex.
- Wolfgang Menzel. 1998. Constraint Satisfaction for Robust Parsing of Spoken Language. *Journal of Experimental and Theoretical Artificial Intelligence*.
- Wolfgang Menzel. 2002. System architecture as a problem of information fusion. In *Proc. Int. Symposium Natural Language Processing between Linguistic Inquiry and Systems Engineering*, pages 74–84, Hamburg.
- Pedro Messeguer and Javier Larossa. 1995. Constraint satisfaction as global optimization. In *IJCAI'95*.
- A. Meyers, R. Reeves, C. Macleod, R. Szekely, V. Zielinska, B. Young, and R. Grishman. 2004. Annotating Noun Argument Structure for NomBank. In *Proceedings of International Conference on Language Resources and Evaluation (LREC'2004)*, Lisbon, Portugal.
- Nuno Miguel, Gabriel Pereira, and Varlso Agra. 1998. Learning verbal transitivity using loglinear models. In *Lecture Notes in AI (LNAI): Proceeding of the 10th European Conference on Machine Learning*, Berlin, April. Springer Verlag.
- Nuno Miguel, Gabriel Pereira, and Varlso Agra. 1999. Using loglinear clustering for subcategorization identification. In *European Conference on Machine Learning (ECML'1999)*.

- Rada Mihalcea and Ehsanul Faruque. 2003. SenseLearner: Minimally Supervised Word Sense Disambiguation for All Words in Open Text. In *Proceedings of ACL/SIGLEX SENSEVAL-III*, Barcelona, Spain, July.
- Rada Mihalcea and Dan Moldovan. 1999. An Automatic Method for Generating Sense Tagged Corpora. In *Proceedings of the 16th National Conference on Artificial Intelligence*. AAAI Press.
- Rada Mihalcea and Dan Moldovan. 2001. eXtended WordNet: Progress Report. In *Proceedings of NAACL Workshop on WordNet and Other Lexical Resources*.
- George Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller. 1990. Wordnet: An On-line Lexical Database. *Special Issue of International Journal of Lexicography*, 3(4):235–312.
- George Miller, Claudia Leacock, Randee Teng, and Ross Bunker. 1993. A Semantic Concordance. In *Proceedings of the ARPA Workshop on Human Language Technology*.
- George Miller, Katherine Miller, Christiane Fellbaum, Randee Teng, Marti Hearst, Karen Kohl, Douglas Jones, Robert Bernick, Naoyuki Nomura, Uta Priss, Shari Landes, Claudia Leacock, , Joachim Grabowski, Shari Landes, Philip Resnik, Martin Chodorow, Elen Voorhees, Graeme Hirst, David St-Onge, Reem Al-Halimi, Rick Kazman, J.F.M. Burg, R.P. Riet, Sanda Harabagiu, and Dan Moldovan. 1998. *Wordnet an Electronic lexical Database*. MIT Press. ISBN 0-262-06197-X.
- Ruslan Mitkov. 1999. Anaphora resolution: the state of the art. Technical report, School of languages and European Studies University of Wolverhampton.
- Roser Morante, Irene Castellón, and Gloria Vázquez. 1998. Los Verbos de Trayectoria. In *Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN'1998)*.
1991. *Third Message Understanding Conference (MUC-3)*, San Diego, California, May. Morgan Kaufmann Publishers, Inc. ISBN 1-55860-236-4.
1992. *Fourth Message Understanding Conference (MUC-4)*, McLean, Virginia, June. Morgan Kaufmann Publishers, Inc. ISBN 1-55860-273-9.
1993. *Fifth Message Understanding Conference (MUC-5)*. Morgan Kaufmann Publishers, Inc. ISBN 1-55860-336-0.
1995. *Sixth Message Understanding Conference (MUC-6)*. ISBN 1-55860-402-2.
1998. *Seventh Message Understanding Conference (MUC-7)*. Only available on line at <http://www.muc.saic.com/>.

Hwee Tou Ng and Hian Beng Lee. 1996. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In Arivind Joshi and Martha Palmer, editors, *Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics*, pages 40–47, San Francisco. Morgan Kaufmann Publishers.

Ian Niles and Adam Pease. 2001. 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.

Sergei Nirenburg and Victor Raskin. 2004. *Ontological Semantics*. Language, Speech, and Communications series. MIT Press. ISBN: 0262140861.

Chikashi Nobata and Satoshi Sekine. 1999. Automatic acquisition of patterns for information extraction. In *International Conference on Computer Processing of Oriental Languages*, Tokushima Japan.

Lluís Padró. 1998. *A Hybrid Environment for Syntax-Semantic Tagging*. Ph.D. thesis, Departament de Llenguatges i Sistemes Informàtics. Universitat Politècnica de Catalunya. Barcelona.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2002. Phase I of the Proposition Bank. Submitted to *Computational Linguistics*.

Marcello Pelillo and Mario Refice. 1994. Learning Compatibility Coefficients for Relaxation Labelling Processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9).

Marcello Pelillo. 1991. Syntactic category disambiguation through relaxation processes. In *Proceedings of the 2nd European Conference on Speech Commun. and Technologies (EuroSpeech'1991)*, Genova, Italy.

Miriam R.L. Petruck. 1996. Frame semantics. *Jef Verschueren, Jan-Ola Östman, Jan Blommaert, and Chris Bulcaen (eds.). Handbook of Pragmatics*. Philadelphia. John Benjamins.

Emanuele Pianta, Luisa Bentivogli, and Christian Girardi. 2002. Multiwordnet: developing an aligned multilingual database. In *First International Conference on Global WordNet*, Mysore, India.

Massimo Poesio, George Ferguson, Peter Heeman, Chung Hee Hwang, David R. Traum, James F. Allen, Nathaniel Martin, and Lenhart K. Schubert. 1994. Knowledge representation in the trains system. In *Proceedings of the AAAI 1994 Fall Symposium on Knowledge Representation for Natural Language Processing in Implemented Systems*, New Orleans, November.

A. Quast, H. Scheideck, T. Geutner, and P. Korthauer. 2003. RoBoDiMa: A Dialog-Object-Based Natural Language Speech Dialog System. In *Proceedings of*

the 8th biannual IEEE workshop on Automatic Speech Recognition and Understanding - (ASRU'2003).

Philip Resnik. 1993. *Selection and Information: A Class-Based Approach to Lexical Relationships*. Ph.D. thesis, University of Pennsylvania.

Philip Resnik. 1995. Using Information Content to evaluate semantic similarity in a taxonomy. In *Proceedings of the International Joint Conference in Artificial Intelligence (IJCAI'1995)*, Montreal, Canada.

Philip Resnik. 1997. Selectional preference and sense disambiguation. In *Proceedings of ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*, Washington, D.C.

Fransesc Ribas. 1995. *On Acquiring Appropriate Selectional Restrictions from Corpora Using a Semantic Taxonomy*. Phd. Thesis, Software Department (LSI). Technical University of Catalonia (UPC), Barcelona, Spain.

German Rigau, Jordi Atserias, and Eneko Agirre. 1997. Combining unsupervised lexical knowledge methods for word sense disambiguation. In *Proceedings of joint 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics ACL/EACL'1997*, Madrid, Spain.

German Rigau, Bernardo Magnini, Eneko Agirre, Piek Vossen, and John Carroll. 2002. MEANING: A Roadmap to Knowledge Technologies. In *Proceedings of Association for Computational Linguistics (COLING'2002) Workshop on A Roadmap for Computational Linguistics*, Taipei, Taiwan.

Ellen Riloff, 1999. *Information Extraction as a Stepping Stone toward Story Understanding*. MIT press, Montreal, Canada.

Douglas Roland and Daniel Jurafsky. 1998. How verb subcategorization frequencies are affected by corpus choice. In *Proceedings of 36th annual meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (COLING/ACL'1998)*, Montreal, Canada.

Douglas Roland and Daniel Jurafsky. 2002. Verb sense and verb subcategorization probabilities. In *Stevenson S. and Merlo P. (eds.) The Lexical Basis of Sentence Processing: Formal, Computational, and Experimental Issues*, John Benjamins, Amsterdam.

Douglas Roland, Daniel Jurafsky, Lise Menn, Susanne Gahl, Elizabeth Elder, and Chris Riddoch. 2000. Verb subcategorization frequency differences between business-news and balanced corpora. In *Proceedings of ACL Workshop on Comparing Corpora*, Hong Kong, China.

- M. Rooth, S. Riezler, D. Prescher, G. Carroll, and F. Beil. 1999. Inducing a semantically annotated lexicon via EM-based clustering. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 104–111.
- Michael Rudolf. 1999. Utilizing constraint satisfaction techniques for efficient graph pattern matching. In Gregor Engels and Grzegorz Rozenberg, editors, *Proceedings of the 6th International Workshop on Graph Grammars and their Application to Computer Science*, Lecture Notes in Computer Science. Springer-Verlag.
- Hana Rudova. 2001. *Constraint Satisfaction with Preferences*. Ph.D. thesis, Faculty of Informatics, Masaryk University,.
- Ivan A. Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. 2001. Multiword expressions: A pain in the neck for nlp. Technical report, Lingo WP2001-03.
- Erik F. Tjong Kim Sang and Jorn Veenstra. 1999. Representing text chunks. In *EACL*, pages 173–179.
- Patrick Schone and Dan Jurafsky. 2001. Is knowledge-free induction of multiword unit dictionary headwords a solved problem? In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'2001)*, pages 100–108.
- Ingo Schröder. 2002. *Natural Language Parsing with Graded Constraints*. Ph.D. thesis, Department of Computer Science. University of Hamburg, Germany.
- Sabine Schulte im Walde. 2000. Clustering Verbs Semantically According to their Alternation Behaviour. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING'2000)*, pages 747–753, Saarbrücken, Germany, August.
- Hinrich Schütze. 1992. Dimensions of meaning. In *Proceedings of Supercomputing*, Los Alamitos, California.
- Dennis Shasha, Jason Tsong-Li Wang, Kaizhong Zhang, and Frank Y. Shih. 1994. Exact and Approximate Algorithms for unordered Tree Matching. *IEEE transactions on System Man and Cybernetics*, 28(5):668–678, April.
- Lei Shi and Rada Mihalcea. 2005. Putting pieces together: Combining framenet, verbnet and wordnet for robust semantic parsing. In *Proceedings of the Conference on Intelligent Text Processing and Computational Linguistics (CicLing'2005)*, pages 100–111.
- Wojciech Skut and Thorsten Brants. 1998. Chunk tagger - statistical recognition of noun phrases. *CoRR*, cmp-lg/9807007.
- Koenraad De Smedt, Helmut Horacek, and Michael Zock, 1996. *Lecture notes in Artificial Intelligence*, volume 1036, chapter Some problems with current architectures in Natural Language Generation, pages 17–46. Springer Verlag.



- John F. Sowa. 1976. Conceptual graphs for a database interface. *IBM Journal of Research and Development*, 20(4):336–357.
- S. Stevenson and P. Merlo. 1998. Automatic verb classification using distributions of grammatical features. In *Proceedings of the 9th Conference of the European Chapter of the Association of Computational Linguistics (EACL'1998)*, Bergen, Norway.
- Mark Stevenson. 1999. *Multiple Knowledge Sources for Word Sense Disambiguation*. Ph.D. thesis, University of Sheffield.
- Carme Torras. 1989. Relaxation and neural learning: Points of convergence and divergence. *Journal of Parallel and Distributed Computing*, 6:217–244.
- Andrea Torsello and Edwin Hancock. 2003. Computing approximate tree edit distance using relaxation labeling. *Pattern Recognition Letters*, (24):1089–1097. Elsevier.
- Dan Tufis, Dan Cristea, Results Sofia Stamou BalkaNet: Aims, Methods, and Perspectives. A General Overview. 2004. Balkanet: Aims, methods, results and perspectives. a general overview. *Romanian Journal of Information Science and Technology Special Issue on BalKanet*, 7(1-2).
- Jordi Turmo, Horacio Rodríguez, and Neus Catala. 1999. An adaptable IE System to New Domains. *Applied intelligence*, 10:225–246.
- A. Ushioda, D. Evans, T. Gibson, and A. Waibel. 1993. The automatic acquisition of frequencies of verb subcategorization frames from tagged corpora. In *Proceedings of ACL Workshop on the Acquisition of Lexical Knowledge from Text*, Columbus, Ohio.
- Takeito Utsuro and Yuji Matsumoto. 1997. Learning Probabilistic Subcategorization Preference by Identifying Case Dependencies and Optimal Noun Class Generalization Level. *Proceedings of the Conference on Applied Natural Language Processing (ANLP'1997)*.
- Glória Vázquez, Ana Fernández, and Maria Antonia Martí. 2000. *Clasificación Verbal, Alternancias de Diátesis*. Number 3 in Cuaderns de Sintagma. Sintagma.
- J.L. Vicedo and A. Ferrández. 2000. Importance of pronominal anaphora resolution to question answering systems. In *Proceedings of 38th Annual Meeting of the Association for Computational Linguistics (ACL'2000)*, pages 555–562.
- Aline Villavicencio. 2003a. Verb-particle constructions and lexical resources. In *Proc. of the ACL Workshop on MWEs*, Sapporo, Japan.
- Aline Villavicencio. 2003b. Verb-particle constructions in the world wide web. In *Proc. of the ACL-SIGSEM Workshop on the Linguistic Dimensions of Prepositions and their use in Computational Linguistics Formalisms and Applications*.

- Piek Vossen, editor. 1998. *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Kluwer Academic Publishers .
- Atro Voutilainen and Lluís Padró. 1997. Developing a hybrid np parser. In *Proceedings of the 5th Conference on Applied Natural Language Processing, (ANLP'1997)*, pages 80–87, Washington DC.
- Volkang Wahlster, editor, 2000. *VerbMobil: Foundations of Speech-to-Speech Translation*, chapter System Architectures and Software Integration. Springer-Verlang.
- Jason Tsong-Li Wang, Kaizhong Zhang, and Karpjoo Jeong and Dennis Shasha. 1994. A system for Approximate Tree Matching. *IEEE transactions on Knowledge and Data Engineering*, 6(4).
- Yorick Wilks and Roberta Catizone, 1999. *Can We Make Information Extraction More Adaptive*, pages 1–16. Lecture Notes in artificial Intelligence. Springer-Verlang. Subseries of Lectures Notes in Computer Science.
- W. A. Woods. 1985. What's in a link: Foundations for semantic networks. In R. J. Brachman and H. J. Levesque, editors, *Readings in Knowledge Representation*, pages 217–241. Kaufmann, Los Altos, CA.
- Zhibiao Wu and Martha Palmer. 1994. Verb semantics and lexical selection. In *Proceedings of the 32nd Annual meeting of the Association for Computational Linguistics*, pages 133–138.
- R. Yangarber and R. Grishman. 1998. NYU: Description of the Proteus/PET System as Used for MUC-7 ST. In *MUC-7*.
- Deniz Yuret. 1998. *Discovery of Linguistic Relations Using Lexical Attraction*. Ph.D. thesis, Massachusetts Institute of Technology. cmp-lg/9805009.
- A. Zarkar and D. Zeman. 2000. Automatic extraction of subcategorization frames for czech. In *Proceedings of 18th International Conference on Computational Linguistics (COLING'00)*, Saarbrücken, Germany.
- Klaus Zechner and Alex Waibel. 1998. Using chunk based partial parsing of spontaneous speech in unrestricted domains for reducing word error rate in speech recognition. In *Proceedings of the International Conference on Computational Linguistics (COLING-ACL'1998)*.
- Wendy-M. Zickus. 1994. A Comparative Analysis of Beth Levin's English Verb Class Alternations and WordNet's sense for the verb classes Hit, Touch, Break and Cut. In *Proceedings of The Post-Coling'1994 International Workshop on Directions of Lexical Research*.

APPENDIX A.

Author's Most Relevant Publication

A.1 Book Chapters

-  Jordi Atserias, Salvador Climent, Xavier Farreres, German Rigau and Horacio Rodríguez “Combining Multiple Methods for the Automatic Construction of Multilingual WordNets” in *Recent Advances in Natural Language Processing II*. Current Issues in Linguistic Theory 189. pages 327-441. Eds. Nicolas Nicolov and Ruslan Mitkov. John Benjamins Pub. Co. 1997. ISBN: 90-272-3695-X
-  Jordi Atserias “A Robust Semantic Parsing Approach” in *Artificial Intelligence and Computer Science*. Ed. Susan Shannon, Nova Science Publisher Inc. 2005. Chapter 7, pages 177-196. ISBN 1-59454-411-5.

A.2 Journals

- Jordi Atserias, Núria Castell, Neus Català, Horacio Rodríguez and Jordi Turmo “Del Texto a la Información” *NOVATICA* vol. 133, pages 31-36, ISBN: 0211-2124. May-June 1998.
- Eneko Agirre, German Rigau, Lluís Padró and Jordi Atserias “Combining supervised and unsupervised lexical knowledge methods for word sense disambiguation” in Volume 34, Issue 1/2, *Special Double Issue on The Senseval Word Sense Disambiguation Exercise*. Computer & the Humanities. Kluwer Academic Publishers. 2000. ISSN: 0010-4817

A.3 Conferences

- Victoria Arranz, Jordi Atserias, and Mauro Castillo. Multiword expressions and word sense disambiguation. In *Computational Linguistics and Intelligent Text Processing, 6th International Conference (CICLing'2005)*, pages 250–262, Mexico, 2005. Lecture Notes in Computer Science 3406 Springer. ISBN 3-540-24523-5
- Jordi Atserias, Salvador Climent, Xavier Farreres, German Rigau, and Horacio Rodríguez. Combining Multiple Methods for the Automatic Construction of Multilingual WordNets. In *Recent Advances in Natural Language (RANLP'1997)*, pages 143–149, Tzigov Chark, Bulgaria, 1997.
- Jordi Atserias, Josep Carmona, Irene Castellón, Sergi Cervell, Montserrat Civit, Lluís Màrquez, Maria-Antonia Martí, Lluís Padró, Roser Placer, Horacio Rodríguez, Mariona Taulé, and Jordi Turmo. Morphosyntactic Analysis and Parsing of Unrestricted Spanish Text. In *1st International Conference on Language Resources and Evaluation (LREC'1998)*, pages 1267–1272, Granada, Spain, 1998. ISBN 3-922641-46-6.
- Jordi Atserias, Irene Castellón, Montse Civit, and German Rigau. Using Diathesis for Semantic Parsing. In *Venecia per il Trattamento automatico delle lingue (VEXTAL'1999)*, pages 385–392, Venice, Italy, 1999. Unipress. ISBN: 88-8098-1-12-9
- Jordi Atserias, Irene Castellón, Montse Civit, and German Rigau. Semantic Analysis based on Verbal Subcategorization. In *Conference on Intelligent Text Processing and Computational Linguistics (CICLing'2000)*, pages 330–340, Mexico, 2000. ISBN: 970-18-4206-5.
- Jordi Atserias, Lluís Padró, and German Rigau. Integrating Multiple Knowledge Sources for Robust Semantic Parsing. In *Recent Advances on Natural Language Processing (RANLP'2001)*, Tzigov Chark, Bulgaria, 2001. ISBN 954-90906-1-2.
- Jordi Atserias, Mauro Castillo, Francis Real, Horacio Rodríguez, and German Rigau. Exploring large-scale Acquisition of Multilingual Semantic Models for Predicates. In *XIX Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN'2003)*, pages 39–46, Alcalá de Henares, Spain, 2003. ISSN 1136-5948.
- Jordi Atserias, Luis Villarejo, and German Rigau. Integrating and porting Knowleges across Languages. In *Recent Advances in Natural Language Processing (RANLP'2003)*, pages 31–37, Borovets, Bulgaria, 2003. ISBN: 954-90906-6-3.
- Jordi Atserias, Luis Villarejo, and German Rigau. Starting up the Multilingual Central Repository. In *XIX Congreso de la Sociedad Española para el*

Procesamiento del Lenguaje Natural (SEPLN'2003), pages 261–268, Alcalá de Henares, Spain, 2003. ISSN 1136-5948.

- Jordi Atserias, Salvador Climent, and German Rigau. Towards the MEANING Top Ontology: Sources of Ontological Meaning. In *4th International Conference on Language Resources and Evaluation (LREC'2004)*, pages 11–14, Lisbon, Portugal, 2004. ISBN 2-9517408-1-6.
- Jordi Atserias, Bernardo Magnini, Octavian Popescu, Eneko Agirre, Aitziber Atutxa, German Rigau, John Carroll, and Rob Koeling. Cross-Language Acquisition of Semantic Models for Verbal Predicates. In *4th International Conference on Language Resources and Evaluation (LREC'2004)*, pages 33–36, Lisbon, Portugal, 2004. ISBN 2-9517408-1-6.
- Jordi Atserias, German Rigau, and Luis Villarejo. Spanish WordNet 1.6: Porting the Spanish Wordnet across Princeton versions. In *4th International Conference on Language Resources and Evaluation (LREC'2004)*, pages 161–164, Lisbon, Portugal, 2004. ISBN 2-9517408-1-6.
- Jordi Atserias, Luis Villarejo, German Rigau, Eneko Agirre, John Carroll, Bernardo Magnini, and Piek Vossen. The MEANING Multilingual Central Repository. In *2nd International Global WordNet Conference (GWC'2004)*, pages 23–30, Brno, Czech Republic, 2004. ISBN 80-210-3302-9
- Jordi Atserias, Salvador Climent, Jordi Atserias, German Rigau, and Joaquim Moré. A Proposal for a Shallow Ontologization of Wordnet. In *XXI Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN'2005)*, pages 161–167, Granada, Spain, 2005. ISSN 1135-5948.
- Jordi Atserias, Eli Comelles, and Aingeru Mayor. TXALA un analizador libre de dependencias para el castellano. In *XXI Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN'2005)*, pages 455–456, Granada, Spain, 2005. ISSN 1135-5948.
- Jordi Atserias, Lluís Padró, and German Rigau. An Integrated Approach to Word Sense Disambiguation. In *Recent Advances in Natural Language Processing (RANLP'2005)*, pages 82–88, Borovets, Bulgaria, 2005. ISBN 954-91743-3-6
- Jordi Atserias. Un Enfoque Integrado para la Desambiguación. In *XXI Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN'2005)*, pages 179–186, Granada, Spain, 2005. ISSN 1135-5948
- Irene Castellón, Montse Civit, and Jordi Atserias. Syntactic Parsing of Unrestricted Spanish Text. In *1st International Conference on Language Resources and Evaluation (LREC'1998)*, pages 603–609, Granada, Spain, 1998. ISBN 3-922641-46-6.

- Mauro Castillo, Francis Real, Jordi Atserias, and German Rigau. The TALP Systems for Disambiguating WordNet Glosses. In *SENSEVAL-3 3rd International Workshop on the Evaluation of Systems for the Semantic Analysis of Text, Task: Word-Sense Disambiguation of WordNet Glosses*, pages 93–96, Barcelona, Spain, 2004. ISBN 1-932432-44-2.
- Montse Cuadros, Jordi Atserias, Mauro Castillo, and German Rigau. Automatic Acquisition of Sense Examples Using ExRetriever. In *IBERAMIA Workshop on Lexical Resources and The Web for Word Sense Disambiguation*, pages 97–104, Mexico, 2004. ISBN 968-863-786-6
- Juan Fernandez, Mauro Castillo, German Rigau, Jordi Atserias, and Jordi Turmo. Automatic Acquisition of Sense Examples using ExRetriever. In *4th International Conference on Language Resources and Evaluation (LREC'2004)*, pages 25–28, Lisbon, Portugal, 2004. ISBN 2-9517408-1-6.
- German Rigau, Jordi Atserias, and Eneko Agirre. Combining Unsupervised Lexical Knowledge Methods for Word Sense Disambiguation. In *Joint 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics (ACL/EACL'1997)*, pages 48–55, Madrid, Spain, 1997.
- German Rigau, Eneko Agirre, and Jordi Atserias. The MEANING project. In *XIX Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN'2003)*, pages 307–308, Alcalá de Henares, Spain, 2003. ISSN 1136-5948.
- Luis Villarejo, Jordi Atserias, Gerard Escudero, and German Rigau. First Release of the Multilingual Central Repository of MEANING. In *XIX Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN'2003)*, pages 307–308, Alcalá de Henares, Spain, 2003. ISSN 1136-5948.

APPENDIX A.

Consistent Labeling Problems and Relaxation Labeling

This appendix will briefly introduce the formulation of Consistent Labeling Problems (CLP) and the relaxation labeling algorithm used in this thesis to find the local most consistent solution of the formulated CLP.

A.1 Consistent Labeling Problems

A natural way to model Constraint Satisfaction Problem (CSP) is the *Consistent Labeling Problems* (CLP) [Messeguer and Larossa, 1995]. A *Consistent Labeling Problem* basically stands as the problem of finding the most consistent assignments of a set of variables, given a set of constraints.

Both, CLP and CSP are been successfully used in several NLP task, Part of Speech tagging [Pelillo, 1991], [Pelillo and Refice, 1994], [Padró, 1998], for parsing, using *Constraint Grammars* [Voutilainen and Padró, 1997] and *Weighted Constraint Dependency Grammars* (WCDG), ([Schröder, 2002], [Daum et al., 2002] [Foth et al., 2003] [Daum, 2004] which uses constraint optimization techniques to integrate deep and shallow parsing techniques for German). But also to more complex task such as, Machine Translation (Mikrokosmos [Beale, 1996]) or Text planning (ICONOCLAST¹ [Kibble and Power, 2000]) or mapping taxonomies [Daudé, 2005].

A Labeling Problem is defined by a set of variables (or units) V_i , a set of labels (domain) for each variable D_i , a compatibility relation over tuples. Compatibilities are real-value functions $r_{ij} : D_i \times D_j \rightarrow \mathfrak{R}$ where $r_{i,j}(a,b)$ refers to the compatibility of the simultaneous assignment of a to V_i and b to V_j .

A *labelling* is a weighted assignment of labels to variables. More than one label can be assigned to the same variable, provided that the sum of he weights for each variable is 1. In a similar way than CSP aims to find total assignments

¹<http://www.itri.brighton.ac.uk/projects/iconoclast>

where constraints are not violated, CLP looks for labelling where variables are highly compatible with respect to compatibility functions.

A.2 Algorithms to solve CLP

Consistent Labeling Problems (CLP) can be solved via Relaxation Labeling. Relaxation labeling is a generic name for a family of iterative algorithms which perform function optimization, based on local information. The algorithm finds a combination of values for a set of variables such that satisfies -to a maximum possible degree- a set of given constraints. This formulation allow to naturally integrate different kinds of knowledge coming from different sources (linguistic and statistical), which may be partial, partially incorrect or even inconsistent.

In this section the relaxation algorithm is described from a general point of view.

Let $V = \{v_1, v_2, \dots, v_n\}$ be a set of variables.

Let $T_i = \{t_{i1}, t_{i2}, \dots, t_{im_i}\}$ be the set of possible labels for variable v_i (where m_i is the number of different labels that are possible for v_i).

Let \mathcal{C} be a set of constraints between the labels of the variables. Each constraint r has the form:

$$C_r \quad [(v_{r_1}, t_{r_1k_1}), (v_{r_2}, t_{r_2k_2}), \dots, (v_{r_{d_r}}, t_{r_{d_r}k_{d_r}})]$$

That is, each constraint is a combination of pairs (variable,label) associated to a real value C_r expressing *compatibility*. For instance, the constraint $[v_1 = A] \sim^{0.53} [v_3 = B]$ states that the combination of variable v_1 having label A , and variable v_3 having label B has a compatibility value of 0.53. Constraints can be of any order d_r , so we can define the compatibility value for combinations of any number of pairs (variable,label). Obviously, we can have combinations of at most n variables.

The aim of the algorithm is to find a *weighted labeling* such that *global consistency* is maximized. A *weighted labeling* is a weight assignment for each possible label of each variable:

$\bar{P} = (P_1, P_2, \dots, P_n)$ where each P_i is a vector containing a weight for each possible label of v_i , that is: $P_i = (p_{i1}, p_{i2}, \dots, p_{im_i})$, being p_{ij} the weight for label t_{ij} .

Maximizing *global consistency* is defined as maximizing the average support that *each* variable labeling receives from the others. The goal is selecting a weight distribution such that the labeling is consistent -to the maximum possible extent- with the compatibilities expressed by the constraint set.

Also, we need to define:

R_{ij} is the set of constraints on label t_{ij} for variable v_i , i.e. the constraints formed by any combination of variable–label pairs that includes the pair (v_i, t_{ij}) .

$Inf(r, i, j) = C_r \times p_{r_1 k_1}(s) \times \dots \times p_{r_{d_r} k_{d_r}}(s)$, is the *influence* of constraint r on label t_{ij} , computed as the product of the current weights (at time step s) for the labels appearing in the constraint except (v_i, t_{ij}) (representing *how applicable* the constraint is in the current context) multiplied by C_r which is the constraint compatibility value (stating *how compatible* the pair is with the context).

S_{ij} is the support received by the pair (v_i, t_{ij}) from the context. The support for a pair variable–label (S_{ij}) expresses *how compatible* is the assignment of label t_{ij} to variable v_i with the labels of neighboring variables, according to the constraint set.

Although several support functions may be used, we chose the following one, which defines the support as the sum of the influence of every constraint on a label, following the results of [Padró, 1998],

$$S_{ij} = \sum_{r \in R_{ij}} Inf(r, i, j)$$

After these definitions, we can define more formally that maximizing *global consistency* consists of maximizing, for each v_i , the weighted sum of the support received by each of its possible labels, that is:

$$\sum_{j=1}^{m_i} p_{ij} \times S_{ij} \quad \forall i (1 \leq i \leq n)$$

The pseudo-code for the relaxation labeling algorithm can be found in Figure 4. It consists of the following steps (step numbers refer to pseudo-code) :

- (1) start in a initial labeling (random or heuristically chosen) \bar{P}_0 .
- (4-6) for each variable, compute the support S_{ij} that each label receives from the current weights for the labels of the other variables (i.e. see how compatible is the current weighting with the current state of the other variables, given the set of constraints).
- (7-9) Compute weight for each variable label at time step $s + 1$ according to the support they receive (that is, increase weight for labels with high support, and decrease weight for those with low support). Although there are several possibilities [1998; 1989], the chosen updating function in our case was the following:

$$p_{ij}(s + 1) = \frac{p_{ij}(s) \times (1 + S_{ij})}{\sum_{k=1}^{m_i} p_{ik}(s) \times (1 + S_{ik})}$$

- (11) iterate the process until a convergence criterion is met. The usual criterion is waiting until there are no significant changes.

Algorithm 4 Pseudo code of the relaxation labeling algorithm.

```

 $P \leftarrow \bar{P}_0$ 
 $s \leftarrow 0$ 
repeat
  for each variable  $v_i$  do
    for each  $t_{ij}$  do
       $S_{ij} \leftarrow \sum_{r \in R_{ij}} \text{Inf}(r, i, j)$ 
    end for
  end for
until no more changes

```

Advantages of the algorithm are:

- Its highly local character (each variable can compute its new label weights given only the state at previous time step). This makes the algorithm highly parallelizable (we could have a processor to compute the new label weights for each variable, or even a processor to compute the weight for each label of each variable).
- Its expressivity: The problem is stated in terms of constraints between variable labels.
- Its flexibility: We do not have to check absolute consistency of constraints.
- Its robustness: It can give an answer to problems without an exact solution (incompatible constraints, insufficient data, ...)

Drawbacks of the algorithm are:

- Its cost. Being n the number of variables, v the average number of possible labels per variable, c the average number of constraints per label, and I the average number of iterations until convergence, the average cost is $n \times v \times c \times I$, that is, it depends linearly on n , but for a problem with many labels and constraints, or if convergence is not quickly achieved, the multiplying terms might be much bigger than n .
- Since it acts as an approximation of gradient step algorithms, it has their typical convergence problems: Found optima are local, and convergence is not guaranteed, since the chosen step might be too large for the function to optimize.

APPENDIX B.

Integration Example

As the biggest library if it is in disorder is not as useful as a small but well-arranged one, so you may accumulate a vast amount of knowledge but it will be of far less value to you than a much smaller amount if you have not thought it over for yourself.

Arthur Schopenhauer

In order to show the complexity of this integration, we will gather the different pieces of information that could be associated to the sentence “*The cat eats fish*” on some of the most broadly used resources in NLP, WordNet, VerbNet, FrameNet, SUMO and MultiWordNet Domains.

WordNet has a wide coverage for English and contains a huge amount of implicit and explicit semantic information (e.g. semantic relations such: hyperonym, meronymy, antonymy, entailment, etc). However, It has poor information about the syntax behavior. Other extensions of WordNet, could also bring other types of information, e.g. eXtended WordNet¹ [Mihalcea and Moldovan, 2001], [Harabagiu and Maiorano, 1999] could be used to explicitate indirect relations between concepts (lexical chaining).

Princeton WordNet1.6 contains different senses for the verb *eat* (see Figure B.1), but also for the nouns *cat* (see Figure B) and *fish* (See Figure B.3). Each sense is defined by a set of synonyms (named variants) and also contains a gloss. Apart from the different sense distinction that can be associated to a word, WordNet also contains semantic relations with other concepts, furthermore that *hyperonym* relations, for instance *entailment* or *holonomy*, see figure B.

Although WordNet is not an ontology and inference over WordNet is not sound, some relation can be carefully inherited, for instance see figure B.2 showing the inherited parts of *fish_1* according the Princeton WordNet interface.

¹<http://xwn.hlt.utdallas.edu/>

Variants	Gloss
eat_1	take in solid food: She was eating a banana;
eat_2	eat a meal; take a meal: We did not eat until 10 P.M. because there were so many phone calls;
eat_3, feed_6	take in food; used of animals only: This dog doesn't eat certain kinds of meat; What do whales eat?;
eat_4 consume_5 eat_up_2 use_up_1 deplete_1 exhaust_2 run_through_2 wipe_out_1	use up, as of resources or materials: this car consumes a lot of gas; We exhausted our savings; They run through 20 bottles of wine a week;
eat_5 eat_on_1	worry or cause anxiety in a persistent way: What's eating you?;
eat_6 corrode_1 rust_2	cause to rust: The acid corroded the metal;

Table B.1: The verb "eat" in WordNet1.6

Variants	Gloss
cat_1 true_cat_1	feline mammal usually having thick soft fur and being unable to roar; domestic cats; wildcats
cat_2 guy_1 hombre_1	an informal term for a youth or man: a nice guy; the guy's only doing it for some doll;
cat_3	a spiteful woman gossip: what a cat she is!;
cat_4 cat-o'-nine-tails_1	a whip with nine knotted cords: British sailors feared the cat;
cat_5 Caterpillar_2	a tractor that is driven by caterpillar tracks
cat_6 big_cat_1	any of several large cats typically able to roar and living in the wild

Table B.2: The noun "cat" in WordNet1.6

Variants	Gloss
fish_1	any of various mostly cold-blooded aquatic vertebrates usually having scales and breathing through gills
fish_2	the flesh of fish used as food
fish_3 chump_1 fool_2 gull_1 mark_8 patsy_1 fall_guy_1 sucker_1 schlemiel_1 shlemiel_1 soft_touch_1 mug_2	a person who is gullible and easy to take advantage of
fish_4 go_fish_1	a game for two players who try to assemble books of cards by asking the opponent for particular cards

Table B.3: The noun "fish" in WordNet1.6

```
eat_1 -- (take in solid food)
  ENTAILS; chew, masticate, manducate, jaw
  ENTAILS: swallow, get down

cat_5 -- ((trademark) a tractor that is driven by caterpillar tracks)
  HAS PART: caterpillar tread, caterpillar tracks
=> tractor
  => vehicle
  HAS PART: splashboard, dashboard

cat_6 -- (any of several large cats typically able to roar and living in the wild)
  MEMBER OF: Felidae, family Felidae
  MEMBER OF: Carnivora, order Carnivora
  MEMBER OF: Eutheria, subclass Eutheria
  MEMBER OF: Mammalia, class Mammalia
  MEMBER OF: Vertebrata, subphylum Vertebrata, Craniata
  MEMBER OF: Chordata, phylum Chordata
  MEMBER OF: Animalia, kingdom Animalia, animal kingdom
```

Figure B.1: Example of WordNet relations

Sense 1 fish -- (any of various mostly cold-blooded aquatic vertebrates usually having scales and breathing through gills)

- HAS PART: fish scale
- HAS PART: roe
- HAS PART: milt
- HAS PART: lateral line, lateral line organ
- HAS PART: fin
 - HAS PART: ray
- HAS PART: tail fin, caudal fin
- HAS PART: fishbone

=> aquatic vertebrate

- HAS PART: flipper

=> vertebrate, craniate

- HAS PART: belly
- HAS PART: tail
 - HAS PART: dock
- HAS PART: caudal appendage
- HAS PART: rib, costa
 - HAS PART: costal cartilage
- HAS PART: thorax, chest, pectus
 - HAS PART: sternum, breastbone
 - HAS PART: gladiolus, corpus sternum
 - HAS PART: manubrium
 - HAS PART: thoracic aorta
 - HAS PART: thoracic vein, vena thoracica
 - HAS PART: gallbladder
 - HAS PART: area of cardiac dullness
 - HAS PART: pectoral, pectoral muscle, pectoralis, musculus pectoralis
 - HAS PART: chest cavity, thoracic cavity
 - HAS PART: mediastinum
 - HAS PART: breast
 - HAS PART: rib cage
- HAS PART: vertebrate foot, pedal extremity
 - HAS PART: metatarsus

Figure B.2: Inherited parts of fish_1 according to Wn1.6

FrameNet provides the knowledge needed to identify case frames and semantic roles. FrameNet has a completely different semantic view (Frame Semantics) and contains a rich information about the semantics components of the predicate as well as their syntactic realizations. Frames are associated to Lexical Units (See table B.4). However, it has poor lexical coverage compared to WordNet. There is no frame directly associated to *cat* and and it only associates one frame for all the senses of *eat* (Ingestion Process) and *fish* (Food). Although, as FrameNet has a corpus associated, it also contains frame elements occurrences in different syntactic patterns (valences) (See table B.5).

Frame	Ingestion
Definition	An Ingestor consumes food, drink, or smoke (Ingestibles). This may include the use of an Instrument. Sentences that describe the provision of food to others are NOT included in this frame.
FEs	
<i>Core</i>	
<i>Ingestibles</i> [Ingible]	The Ingestibles are the entities that are being consumed by the Ingestor.
<i>Ingestor</i> [Ing]	The Ingestor is the person eating, drinking, or smoking.
<i>Uses</i>	Cause_motion, Intentionally_affect
<i>Lexical Units</i>	breakfast.v, consume.v, devour.v, dine.v, down.v, drink.v, eat.v, feast.v, feed.v, gobble.v, gulp.n, gulp.v, guzzle.v, have.v, imbibe.v, ingest.v, lap.v, lunch.v, munch.v, nibble.v, nosh.v, nurse.v, quaff.v, sip.n, sip.v, slurp.n, slurp.v, snack.v, sup.v, swig.n, swig.v, swill.v

Frame Element	Number Annotated	Realizations(s)
<i>Ingestibles</i>	26	NP.Ext 4 NP.Obj 17 -. 5
<i>Ingestor</i>	26	NP.Ext 21 -. 2 PP[by].Comp 3

Table B.4: FrameNet Frames for “Ingest”

Number Annotated	Patterns	
26 <i>TOTAL</i>	<i>Ingestibles</i>	<i>Ingestor</i>
5	– –	NP Ext
1	NP Ext	– –
3	NP Ext	PP[by] Comp
1	NP Obj	– –
16	NP Obj	NP Ext

Table B.5: Valences from Frame “Ingest”

Roles	
role name	selres
<i>Agent</i>	value= + type= animate
<i>Instrument</i>	value= + type= concrete
<i>Patient</i>	value= + type= comestible

Figure B.3: Roles for VerbNet Class for eat-39.1

VerbNet is a verb lexicon with explicit syntactic and semantic information based on Levin's verb classification. In VerbNet the arguments of the verb are represented at semantic level and thus they have associated semantic roles. In [Giuglea and Moschitti, 2004], VerbNet has been used to relate FrameNet and syntactic argument annotated in PropBank (a 300.000 word corpus from the Wall Street Journal annotated with predicate-arguments relations using VerbNet).

Following our example, VerbNet relates all the different senses of the verb *eat* to the same verbal class (*eat-39.1*). VerbNet includes information about semantic roles, selectional preferences, the possible diathesis alternation of a verb (See Figure B.4). It can be argued that the Selectional Preferences does not hold for some of the WordNet senses of *eat*. (e.g. *eat_5* (*eat_on*) and *eat_6* (*cause to rust*).)

On the other hand, comparing with FrameNet, which assigns only two FE (Ingestor, Ingestible), VerbNet explicitates three different roles (Agent/Patient/Instrument). Moreover, VerbNet also explicitates 4 different sub-categorization frames for *eat*: **NP VERB NP**, (*transitive*) **NP VERB** (*unspecified object alternation*), **NP VERB PP(at)** (*conative*) and **NP VERB NP ADJ** (*resultative*), while the valences present in the FrameNet corpus (see figure B.5) are not exactly the same **NP VERB NP**, **NP VERB**, **VERB NP**, **NP VERB PP(by)**.

Some of this differences are due to the different paradigm (the syntactic information in FrameNet comes from a representative corpus while in VerbNet the syntactic information explicitly encoded and theoretically ground), the use of different criteria (e.g. like the explicitation of the passive) but in most cases there are codifying complementary information (alternations NP VERB NP ADJ or VERB NP) or related information (roles vs Frame elements).

LCS: A database of Lexical conceptual Structures was built by hand by Dorr in 1994, organized into semantic classes that are a reformulated version of those in Beth Levin English Verb Classes and Alternations [Levin, 1993]. Figure B.5 shows the two LCS entries associated to the verb *eat*. These structures also contains information about Roles and PropBank arguments. Moreover, each entries is associated to one or more WordNet sense. For instance, in the example, the first one is associate to the senses *eat_2* and *eat_3* while the second is associated to the first sense. Also

frame	276			
desNum	0.2			
primary	Basic Transitive			
syntax	tag	value	SEL	SYN
	NP	Agent		
	VERB			
	NP	Patient		
examples	Cynthia ate the peach			

frame	277			
desNum	1.2.1			
primary	Unspecified Object Alternation			
syntax	tag	value	SEL	SYN
	NP	Agent		
	VERB			
example	Cynthia ate			

frame	278			
desNum	1.3			
primary	Conative			
syntax	tag	value	SEL	SYN
	NP	Agent		
	VERB			
	PREP	at		
	NP	Patient		
example	Cynthia ate at the peach			

frame	279			
desNum	0.4			
primary	Resultative			
syntax	tag	value	SEL	SYN
	NP	Agent		
	VERB			
	NP	Oblique		
	ADJ			
example	Cynthia ate herself sick			

Figure B.4: The four frames of VerbNet Class for eat-39.1

notice that there is not information for the rest of the WordNet senses.


```

(
:DEF_WORD "eat"
:CLASS "39.1.i"
:SOURCES (LEVIN)
:WN_SENSE (("1.5" 00663538 00662381 00670058)
           ("1.6" 00802008 00793267 00802008)
           ("1.7.1" 00932129 00921744 00932129)
           ("2.0" 01143746 01130349 01143746))
:PROPBANK ("arg0")
:THETA_ROLES ((1 "_ag"))
:LCS (act loc (* thing 1) (eat+ingly 26))
:VAR_SPEC ((1 (animate +)))
)

(
:DEF_WORD "eat"
:CLASS "39.1.ii"
:SOURCES (LEVIN)
:WN_SENSE (("1.5" --)
           ("1.6" 00794578)
           ("1.7.1" 00923270)
           ("2.0" 01132466))
:PROPBANK ("arg0 arg1")
:THETA_ROLES ((1 "_ag_th"))
:LCS (cause (* thing 1)
      (go loc (* thing 2)
        (toward loc (thing 2) (in loc (thing 2) (thing 1))))
      (eat+ingly 26))
:VAR_SPEC ((1 (animate +)) (2 (mass +)))
)

```

Figure B.5: Two LCS entries associated to the verb *eat*

Sumo (Suggested Upper Merged Ontology)[Niles and Pease, 2001] is an upper level ontology created as part of the IEEE Standard Upper Ontology Working Group. SUMO consists of a set of concepts, relations, and axioms that formalize a field of interest. Figure B.6 shows the axioms related to the *Eating* concept, the first one staying that “*if instance ACT Eating and patient ACT FOOD, then attribute FOOD Solid*”, and the second staying that “*if instance CARNIVORE Carnivore and instance EAT Eating and agent EAT CARNIVORE and patient EAT PREY, then instance PREY Animal*”. Although *cat* is not represented directly, it is related to the SUMO concept *Feline* (See Figure B.7).

SUMO has been connected to WordNet, thus we can see how the different word-net senses are represented according to SUMO. The WordNet senses of the verb *eat* are basically divided in *Eating* (3 senses), *Process* (1 sense) *IntentionalPsychologicalProcess* (1 sense) and *ChemicalSynthesis* (1 sense).

Eating "The Process by which solid Food is incorporated into an Animal."

```
(subclass Eating Ingesting)

(=>
  (and
    (instance ?ACT Eating)
    (patient ?ACT ?FOOD))
  (attribute ?FOOD Solid))

(=>
  (and
    (instance ?CARNIVORE Carnivore)
    (instance ?EAT Eating)
    (agent ?EAT ?CARNIVORE)
    (patient ?EAT ?PREY))
  (instance ?PREY Animal))
```

Figure B.6: SUMO Eating

Feline: "The Class of Carnivores with completely separable toes,
nonretractable claws, slim bodies, and rounded heads."

(subclass Feline Carnivore)

appearance as argument number 2

(disjoint Canine Feline)

Figure B.7: SUMO Feline

TCO: The EuroWordNet Top Concept Ontology [Vossen, 1998] is a hierarchy of language-independent concepts, reflecting important semantic distinctions, e.g. Object and Substance, Location, Dynamic. As shown in figures B.8, each sense could have multiple properties associated and the resulting representation is sounded and richer than the WordNet Lexicographer File (LF). It is interesting to highlight that *eat_2* and *eat_3* share exactly the same set of properties. While *eat_5* and *eat_6* do not share almost any property with the rest of the senses.

Sense	TCO	MW Domains	SUMO	LF
eat_1	Agentive= Dynamic= Location= Physical= Purpose= UnboundedEvent+ Usage=	gastronomy	Eating+	consumption
eat_2	Location+ Physical+ Purpose+ UnboundedEvent+ Usage+	gastronomy	Eating=	consumption
eat_3	Location+ Physical+ Purpose+ UnboundedEvent+ Usage+	gastronomy, zoology	Eating+	consumption
eat_4	Agentive+ BoundedEvent+ Possession+ Purpose+ Social+	factotum	Process+	consumption
eat_5	Dynamic+ Experience+ Mental+	psychology	IntentionalPsychologicalProcess+	emotion
eat_6	BoundedEvent+ Cause+ Condition+ Dynamic+ Physical+	chemistry	ChemicalSynthesis+	change

Figure B.8: TCO, MW Domains, SUMO and LF for the verb *eat*

MultiWordNet Domains is a hierarchy of domain labels, which are knowledge structures grouping meanings in terms of topics or scripts, e.g. Transport, Sports, Medicine, Gastronomy. As show in figure B.8, It could also bring up a new orthogonal view over the different senses of *eat*: *gastronomy* (2 senses), *gastronomy* and

Sense	TCO	MW Domains	SUMO	LF
cat_1	Animal+ Living+ Object+	zoology	Feline+	animal
cat_2	Function+ Human+ Living+ Object+	person	Male+	person
cat_3	Function+ Human+ Living+ Object+	person	Female+	person
cat_4	Artifact+ Instrument+ Object+	factotum	Device+	artifact
cat_5	Artifact+ Instrument+ Object+ Vehicle+	transport	TransportationDevice+	artifact
cat_6	Animal+ Living+ Object+	zoology	Feline+	animal

Figure B.9: TCO, MW Domains, SUMO and LF for the noun *cat*

Zoology (1 sense), *economy* (1 sense), *psychology* (1 sense) and *chemistry* (1 sense).

As it can be seen in the example, each resource seems to have a partial view of the complete information. Moreover, although some of them are representing similar things from similar point of view, the information provided seems complementary but it does not fit completely. Putting all this resources together can give us a more completed view of the semantic and syntactic behavior of the different elements involved in the sentence to build an appropriate interpretation. Although, the use of different kinds of knowledge coming from different sources (linguistic and statistical) could be inconsistent when used together, this alternative is more realistic than expecting a single resource to cope with all the language phenomena.

The integration of these different theories or pieces of knowledge can be carried in an integrated model (as in the Microtheories in Mikrokosmos). Thus, this integration of knowledge must be carried out in a flexible framework which allows knowledge to be incomplete, partially incorrect or even inconsistent (such as Constraint Satisfaction Techniques).

APPENDIX C.

MCR Examples

When uploading coherently all this knowledge into the Multilingual Central Repository (MCR) a full range of new possibilities appear for improving both SRL and WSD problems (and other Semantic Processes). We will illustrate these new capabilities by two simple examples.

C.1 The “Vaso” Example

The Spanish noun *vaso* has three possible senses. The first one is connected to the same ILI as the English synset *<drinking_glass glass>*. This ILI record, belonging to the Semantic File ARTIFACT has no specific WordNet Domain (FACTOTUM). However, the Top Concept Ontology provides further clues about its meaning: it has the following properties *Form-Object*, *Origin-Artifact*, *Function-Container* and *Function-Instrument*. The SUMO type for this synset is also ARTIFACT. A valuable information also comes from the disambiguated glosses included into the eXtended WordNet. This gloss has two ‘silver’ words¹ (glass, container) and three ‘normal’ words (the rest). For instance, *hold#VBG#8* corresponds to: *contain or hold*”; *have within*: “*The jar carries wine*”; “*The canteen holds fresh water*”; “*This can contains water*”. Further, coming from the Selectional Preferences acquired from SemCor, we know that the typical things that somebody does with this kind of *vaso* are for instance the corresponding equivalent translations to Spanish for *<polish, shine, smooth, smoothen>* or *<beautify, embellish, prettify>*. WordNet 2.0 also provides a new morphological derivational relation: to *glass#v#4* “put in a glass container”. Finally, we must add that this also holds for the rest of languages connected.

vaso_1 02755829-n
LF: 06-NOUN.ARTIFACT
DOMAIN: FACTOTUM

¹High confidence

SUMO: &%Artifact+

TO: 1stOrderEntity-Form-Object

TO: 1stOrderEntity-Origin-Artifact

TO: 1stOrderEntity-Function-Container

TO: 1stOrderEntity-Function-Instrument

EN: drinking_glass glass

IT: bicchiere

BA: edontzi baso edalontzi

CA: got vas

02755829-n drinking_glass glass:

GLOSS: a glass container for holding liquids while drinking

eXtended WordNet:

GLOSS: a glass#NN#2 container#NN#1 for hold#VBG#8 liquid#NNS#1 while drink#VBG#1

DOBJ SemCor

02755829 00849393-v 0.0074 polish shine smooth smoothen

02755829 00201878-v 0.0013 beautify embellish prettify

02755829 00826635-v 0.0010 get_hold_of take

02755829 00140937-v 0.0001 ameliorate amend better improve meliorate

02755829 00083947-v 0.0000 alter change

DOBJ Semicor-No Generalization

02755829 00826635-v get_hold_of take

02755829 00849393-v polish shine smooth smoothen

02755829 01526289-v pass hand reach pass_on turn_over give

02755829 01571054-v offer proffer

Proto-Classes

hear dobj 02755829 0.0003165225

turn dobj 02755829 0.0011137408

bear dobj 02755829 0.0011655012

send dobj 02755829 0.0005092687

lift dobj 02755829 0.0143220878

roll dobj 02755829 0.0056179775

find subj 02755829 6.85143e-05

like subj 02755829 0.0003032475

WN2.0

RELATED TO: glass#v#4 (put in a glass container)

The second sense of *vaso* is the equivalent translation of $\langle vessel, vas \rangle$. This ILI record, belonging to the Semantic File BODY has assigned a different WordNet Domain (ANATOMY). The EuroWordNet Top Ontology in this case, has the following properties *Form-Substance-Solid*, *Origin-Natural-Living*, *Composition-Part* and *Function-Container*. The SUMO label provides the properties and axioms assigned to *BodyVessel*. This gloss has two ‘gold’ words² (tube and circulate) and one ‘silver’ (body_fluid) and the last word is monosemous. From the Selectional Preferences acquired from SemCor, we know that the typical events applied to this kind of *vaso* are for instance the corresponding equivalent translations to Spanish for $\langle inject, shoot \rangle$ or $\langle administer, dispense \rangle$. In this case, there are no new relations coming from WordNet 2.0. As before, we must add that this knowledge can be also ported to the rest of languages integrated into the MCR.

vaso_2 04195626-n
 LF: 08-NOUN.BODY
 DOMAIN: ANATOMY
 {SUMO: &%BodyVessel+

TO: 1stOrderEntity-Form-Substance-Solid
 TO: 1stOrderEntity-Origin-Natural-Living
 TO: 1stOrderEntity-Composition-Part
 TO: 1stOrderEntity-Function-Container

EN: vessel vas
 IT: vaso dotto canale
 BA: hodi baso
 CA: vas

04195626-n vessel vas:
 GLOSS: a tube in which a body fluid circulates

eXtended WordNet:
 GLOSS: a tube#NN#4 in which a body_fluid#NN#1 circulate#VBZ#4

DOBJ SemCor

04195626	01781222	0.0334	be occur
04195626	00058757	0.0072	inject shoot
04195626	01357963	0.0068	follow travel_along

²Hand corrected

04195626	00055849	0.0045	administer dispense
04195626	01012352	0.0022	block close_up impede jam obstruct occlude
04195626	00054862	0.0021	care_for treat
04195626	01670590	0.0017	hinder impede
04195626	00401762	0.0011	cognize know
04195626	01253107	0.0005	go locomote move travel
04195626	01669882	0.0003	keep prevent

DOBJ SemCor No-Generalization

04195626	01357963	follow	travel_along
04195626	01781222	be	occur

SUBJ SemCor

04195626	01831830	0.0133	stop terminate
04195626	01357963	0.0127	flood travel_along
04195626	01830886	0.0043	discontinue
04195626	01779664	0.0008	cease end finish terminate
04195626	01832078	0.0003	continue go_along go_on keep keep_on proceed
04195626	01253107	0.0002	go locomote move travel
04195626	01520167	0.0002	transfer
04195626	01505951	0.0002	give
04195626	01590833	0.0002	furnish provide render supply
04195626	01612822	0.0001	act move
04195626	01775973	0.0000	be

Proto-Classes

open	dobj	04195626	0.0006462453
show	subj	04195626	0.0001756852

The last sense of *vaso* is the equivalent translation of *<glassful, glass>*. This ILI record, belongs to the Semantic File QUANTITY and has assigned a different WordNet Domain (FACTOTUM-NUMBER). The Top Concept Ontology in this case, has the following properties *Composition-Part*, *SituationType-Static* and *SituationComponent-Quantity*. The SUMO label provides the properties and axioms assigned to ConstantQuantity. This gloss has only one 'silver' word from the eXtended WordNet (quantity). The other two have label 'normal'. From the Selectional Preferences acquired from SemCor, we know that the typical events applied to this kind of *vaso* are for instance the corresponding equivalent translations to Spanish for *<drink, im-bibe>* or *<consume, have, ingest take, take_in>*. WordNet 2.0 also provides a new

morphological derivational relation: to *glass#v#4* “put in a glass container”. As before, we must add that this knowledge can be also ported to the rest of languages connected.

```
vaso_3 09914390-n
LF: 23-NOUN.QUANTITY
DOMAIN: NUMBER
SUMO: &%ConstantQuantity+
```

```
TO: 1stOrderEntity-Composition-Part
TO: 2ndOrderEntity-SituationType-Static
TO: 2ndOrderEntity-SituationComponent-Quantity
```

```
EN: glassful glass
IT: bicchierata bicchiere
BA: basocada
CA: got vas
```

```
09914390-n glassful glass:
GLOSS: the quantity a glass will hold
```

```
eXtended WordNet:
GLOSS: the quantity#NN#1 a glass#NN#2 will hold#VB#1
```

DOBJ SemCor

```
09914390      00795711      0.0026 drink imbibe
09914390      01530096      0.0009 accept have take
09914390      00786286      0.0009 consume have ingest take take_in
09914390 01513874      0.0001 acquire get
```

DOBJ Semcor No generalization

```
09914390 00795711 drink imbibe
09914390 01530096 accept have take
```

As we can see, we can add consistently a large set of explicit knowledge about each sense of *vaso* that can be used to differentiate and characterize better their particular meanings. We expect to devise appropriate ways to exploit this unique resource in the future.

C.2 The “Pasta” Example

We will continue illustrating the current content of the MCR, after porting, with another simple example: the Spanish noun *pasta*.

The word *pasta* (see tables C.2 and C.1) illustrates how all the different classification schemes uploaded into the MCR: Lexicographer File, WordNet Domain, Top Concept Ontology, etc. are consistent and makes clear semantic distinctions between the money sense (*pasta_6*), the general/chemistry sense (*pasta_7*) and the food senses (all the rest). The food senses of *Pasta* can now be further differentiate by means of explicit Top Concept Ontology properties. All the food senses are descendants of *substance_1* and *food_1* and inherits the Top Concept attributes *Substance* and *Comestible* respectively.

<p>Domain: chemistry-pure.science LF: 27-Substance SUMO: Substance-SelfConnectedObject-Object-Physical-Entity</p> <p>Top Concept ontology Natural-Origin-1stOrderEntity Substance-Form-1stOrderEntity</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>pasta#n#7 10541786-n <i>paste#1</i> gloss: any mixture of a soft and malleable consistency</p> </div>	<p>Domain: money-economy-soc.science LF: 21-Money SUMO: CurrencyMeasure-ConstantQuantity-PhysicalQuantity-Quantity-Abstract-Entity</p> <p>Top Concept ontology Artifact-Origin-1stOrderEntity Function-1stOrderEntity MoneyRepresentation-Representation-Function-1stOrderEntity</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>pasta#n#6 09640280-n <i>dough#2, bread#2, loot#2, ...</i> gloss: informal terms for money</p> </div>
--	---

Table C.1: Food senses for the Spanish word *pasta*

Selectional Preferences can also help to distinguish between senses, e.g only the money sense has the following preferences as object: *1.44 01576902-v {raise#4}, 0.45 01518840-v {take_in#5, collect#2}* or *0.23 01565625-v {earn#2, garner#1}* or *0.12 01564908-v {clear#15, take_in#10, make#10, gain#8, realize#4, pull_in#2, bring_in#2, earn#1}*.

Table C.3 presents the new selectional preferences acquired for the Spanish word *Pasta*. That is, the prototypical verbs associated to each English equivalent translation or their hypernyms.

We can also investigate new inference facilities to enhance the integration process. After full expansion (**Realization**) of the EWN Top Concept ontology properties,

Domain: gastronomy-alimentation-applied_science LF: 13-food Top concept ontology Comestible-Function-1stOrderEntity Substance-Form-1stOrderEntity							
Top Concept ontology Natural-Origin-1stOrderEntity	<table border="1"> <tr> <td> Top Concept ontology Part-composition-1stOrderEntity </td> <td> pasta#n#1 05671312-n <i>pastry#1,pastry_dough#1</i> gloss: a dough of flour and water and shortening </td> </tr> <tr> <td> pasta#n#4 05886080-n <i>spread#5,paste#3</i> gloss: a tasty mixture to be spread on bread or crackers </td> <td> pasta#n#3 05739733-n <i>pasta#1,alimentary_paste#1</i> gloss: shaped and dried dough made from flour and water & sometimes egg </td> </tr> <tr> <td></td> <td> pasta#n#5 05889686-n <i>dough#1</i> gloss: a dough of flour and water and shortenings </td> </tr> </table>	Top Concept ontology Part-composition-1stOrderEntity	pasta#n#1 05671312-n <i>pastry#1,pastry_dough#1</i> gloss: a dough of flour and water and shortening	pasta#n#4 05886080-n <i>spread#5,paste#3</i> gloss: a tasty mixture to be spread on bread or crackers	pasta#n#3 05739733-n <i>pasta#1,alimentary_paste#1</i> gloss: shaped and dried dough made from flour and water & sometimes egg		pasta#n#5 05889686-n <i>dough#1</i> gloss: a dough of flour and water and shortenings
Top Concept ontology Part-composition-1stOrderEntity	pasta#n#1 05671312-n <i>pastry#1,pastry_dough#1</i> gloss: a dough of flour and water and shortening						
pasta#n#4 05886080-n <i>spread#5,paste#3</i> gloss: a tasty mixture to be spread on bread or crackers	pasta#n#3 05739733-n <i>pasta#1,alimentary_paste#1</i> gloss: shaped and dried dough made from flour and water & sometimes egg						
	pasta#n#5 05889686-n <i>dough#1</i> gloss: a dough of flour and water and shortenings						
Top Concept ontology Artifact-Origin-1stOrderEntity Group-Composition-1stOrderEntity	<table border="1"> <tr> <td> pasta#n#2 05671439-n <i>pie_crust#1,pie_shell#1</i> gloss: pastry used to hold pie fillings </td> </tr> </table>	pasta#n#2 05671439-n <i>pie_crust#1,pie_shell#1</i> gloss: pastry used to hold pie fillings					
pasta#n#2 05671439-n <i>pie_crust#1,pie_shell#1</i> gloss: pastry used to hold pie fillings							

Table C.2: Food senses for the Spanish word *pasta*

we will perform a full expansion through the noun part of the hierarchy of the selectional preferences acquired from SemCor and BNC (and possibly other implicit semantic knowledge currently available in WN such as meronymy information).

pasta#n#1	hyper2 05909338	divide 0,0127 wrap 0,0063 pack 0,0045 mix 0,0044 press 0,0025 check 0,0013 pass 0,0007 add 0,0006 eat 0,0006 make 0,0005 prevent 0,0004 remove 0,0004 produce 0,0002 leave 0,0001 like 0,0001
pasta#n#2	hyper-1 05670938 hyper-2 05670374	eat 0,0017 serve 0,0012 choose 0,0007 include 0,0002 leave 0,0002 take 0,0001 dispense 0,0161 crush 0,0137 pop 0,0120 eat 0,0103 bless 0,0102 chew 0,0095 put_out 0,0064 tuck 0,0058 freeze 0,0050 clutch 0,0048 transfer 0,0015 fill 0,0014 try 0,0013 avoid 0,0006 buy 0,0006 in- clude 0,0001 make 0,0001
pasta#n#3	direct	divide 0,0127 wrap 0,0063 pack 0,0045 mix 0,0044 press 0,0025 check 0,0013 pass 0,0007 add 0,0006 eat 0,0006 make 0,0005 prevent 0,0004 remove 0,0004 produce 0,0002 leave 0,0001 like 0,0001
pasta#n#4	direct hyper-1 05844302	mix 0,0065 add 0,0004 mix 0,0142 picture 0,0097 spread 0,0046 accom- pany 0,0017 serve 0,0016 hate 0,0013 prepare 0,0013 pass 0,0007 do 0,0005 keep 0,0005 include 0,0004 love 0,0004 like 0,0003 hold 0,0002 make 0,0001 produce 0,0001
pasta#n#5	hyper-1 105909338	divide 0,0127 wrap 0,0063 pack 0,0045 mix 0,0044 press 0,0025 check 0,0013 pass 0,0007 add 0,0006 eat 0,0006 make 0,0005 prevent 0,0004 remove 0,0004 produce 0,0002 leave 0,0001 like 0,0001

Table C.3: New Selectional Preferences for Food senses of “pasta”

APPENDIX D.

Lexicographer File - Top Concept Ontology

LF	TCO	LF	TCO	LF	TCO
04	Agentive	23	Quantity	36	Existence
05	Animal	24	Relation		BoundedEvent
06	Artifact	25	Physical	37	Experience
07	Property	26	Static		Mental
08	Object	27	Substance	38	Location
	Natural	28	Time		Physical
09	Mental	29	Dynamic		Dynamic
10	Communication		Physical	39	Experience
11	Dynamic	30	Dynamic		Physical
12	Experience	31	Mental		Dynamic
13	Comestible		Dynamic	40	Possession
14	Group	32	Communication		Dynamic
15	Place		Dynamic	41	Social
16	3rdOrderEntity	33	Social		Dynamic
17	Object		Dynamic	42	Static
18	Human	34	Physical		
19	Phenomenal		Location	43	Phenomenal
20	Plant		Dynamic		Physical
21	Possession	35	Location		Dynamic
22	Dynamic		Dynamic		

Table D.1: LF -TCO Equivalences

APPENDIX E.

Senseval-II issues

The SENSEVAL-II English Lexical Sample task contains several inconsistencies. First the misspelling of some WordNet variants (shown in table E.1) and secondly MWEs variants in the test solutions which are not lexicalized in the input text and last but not least, variants which appear in the test corpus but do not appear in the training (shown in tables E.3 and E.4).

Senseval Variant	Correct WordNet Variant
keep_one-s_nose_to_the_grindstone%2:41:00::	keep_one's_nose_to_the_grindstone%2:41:00::
keep_one-s_distance%2:42:00::	keep_one's_distance%2:42:00::
pull_in_one-s_horns%2:32:00:	pull_in_one's_horns%2:32:00:
pull_the_wool_over_someone-s_eyes%2:32:00::	pull_the_wool_over_someone's_eyes%2:32:00::
wash_one-s_hands%2:32:00::'	wash_one's_hands%2:32:00::
free_will%1:26:00::	free_will%1:07:00::
natural_language_processing%1:10:00::	natural_language_processing%1:09:00::
local_post_office%1:14:01::	local_post_office%1:14:01::
vital%5:00:00:alive(p):01	vital%5:00:00:alive:01

Table E.1: Non existing variants and their correct form

draw.048	draw_in%2:35:00::
dress.068	dress_up%2:36:00::
dress.115	dress_up%2:36:00::
dress.128	dress_up%2:36:00::
live.169	live_on%2:42:00::

Table E.2: MWE variant which are not in the text

Word	Variant
art	art_critic#n#1
art	art_exhibition#n#1
art	art#n#4
art	pop_art#n#1
authority	regulatory_authority#n#1
bar	candy_bar#n#1
bar	sushi_bar#n#1
begin	begin#v#8
carry	carry_off#v#1
carry	carry_on#v#3
carry	carry_over#v#1
carry	carry#v#17
carry	carry#v#19
carry	carry#v#8
chair	bath_chair#n#1
chair	feeding_chair#n#1
chair	musical_chairs#n#1
channel	television_channel#n#1
circuit	computer_circuit#n#1
day	order_of_the_day#n#1
detention	house_of_detention#n#1
develop	develop#v#15
draw	draw_close#v#2
draw	draw_in#v#2
draw	draw_in#v#3
draw	draw_in#v#4
draw	draw_in#v#7
draw	draw_out#v#3
draw	draw_up#v#5
drift	drift_away#v#1
drive	drive_around#v#2
facility	docking_facility#n#1
facility	health_facility#n#1
fatigue	battle_fatigue#n#1
feeling	feeling#n#6
find	find_out#v#1
find	find#v#14
fine	fine_print#n#2
free	free_rein#n#1
free	free_trader#n#1

Table E.3: Senses which appears on the test corpus but not on the training I

Word	Variant
green	green#a#7
green	green_woodpecker#n#1
green	yellowish_green#n#1
grip	grip#n#5
keep	keep_going#v#3
keep	keep_off#v#2
keep	keep_out#v#1
keep	keep_up#v#3
keep	keep_up#v#5
leave	leave#v#14
local	local_government#n#1
mouth	mouth#n#7
nation	balkan_nation#n#1
natural	natural_ability#n#1
natural	natural_resource#n#1
natural	natural_theology#n#1
play	play_off#v#1
play	play_possum#v#1
play	play#v#12
play	play#v#33
pull	pull_off#v#2
pull	pull_the_plug#v#1
pull	pull_up_short#v#1
see	see#v#10
sense	sense_of_smell#n#1
sense	sense_organ#n#1
serve	serve#v#5
strike	strike_home#v#1
strike	strike#v#14
strike	strike#v#19
turn	turn_away#v#4
turn	turn_in#v#3
turn	turn_off#v#1
turn	turn#v#26
wander	wander#v#2
wash	wash_up#v#2
work	work_at#v#1
work	work_up#v#1
work	work_up#v#2
work	work#v#24
yew	western_yew#n#1

Table E.4: Senses which appears on the test corpus but not on the training II