Computational Linguistic Text Processing
Lexicon, Grammar, Parsing and Anaphora Resolution

Rodolfo Delmonte

CONTENTS

Preface p. 2
Introduction p. 4

CHAPTER I: Inducing Fully Specified Lexical Representations p. 21

CHAPTER II: Treebanking: from Phrase Structure to Dependency Representations p. 65

CHAPTER III: Parsing Arguments And Adjuncts p. 113

CHAPTER IV: Deep Linguistically-Based Parsing p. 149

CHAPTER V: Grammar and Structure p. 185

CHAPTER VI: Anaphoric Binding p. 215

CHAPTER VII: Quantifiers and Anaphora p. 251

CHAPTER VIII: Discourse Anaphora Resolution p. 267

CHAPTER IX: Linguistic Information Extraction for Text Correction and Summarization p. 335

BIBLIOGRAPHY p. 363
PREFACE

This book is the second in a series of books organized as an experimental exercise: they contain both theoretical background and the output of the system, GETARUNS that enacts and applies the theory. The architecture of the system is strictly related to the structure of the books.

Thus, we can think of the books as being organized around two scientifically distinct but in fact strictly interrelated fields of research:

• sentence level linguistic phenomena
• text or discourse level linguistic phenomena

the former is to be described by means of grammatical theories, the latter requires the intervention of extralinguistic knowledge, i.e. knowledge of the world.

Book 1 – or the current book – addresses sentence grammar or what is usually referred to as such by theoretical linguists. It does it by dividing up – somewhat ideally and sometimes arbitrarily – what must or needs to be computed at sentence level from what needs not or cannot be computed at the same level, and consequently belongs to discourse grammar. In that sense, the subdivision is not totally an arbitrary one, even though overlappings are normal cases and will be discussed where needed.

The book also indirectly does another (un)intended subdivision: the one existing between syntax and semantics. Again, it would be impossible not to deal with semantically related issues when talking about syntax or the lexicon. However, semantics with uppercase S, is only treated in Book 2 – already published – where discourse and text level grammar is tackled.

So eventually, this book deals with lexicon, morphology, tagging, treebanks, parsing, quantifiers and anaphoric or pronominal binding. In other words, all that concerns the level of sentence grammar in a computational environment, i.e.

sentence level parsing.

Sentence level Grammar – as has been purported in linguistic theories – takes care of all grammatical and linguistic relations that belong to that level. Knowledge of the world and semantic disambiguation do not interfere with the rules of sentence grammar, and can be thought of as a separate level of computation, provided that the lexicon be structured in such a way to allow such a subdivision of tasks.

For that reason, the first chapter is devoted to the Lexicon and to a linguistically-based way to derive principled rules that create lexical entries for Out of Vocabulary words when needed on the basis of a fixed number of syntactic, semantic and conceptual lexical types.

The second chapter presents our work on a treebank of Italian and the way to produce a conversion algorithm to Dependency Structures (almost) effortlessly.

In the following three chapters – the central part of this book – Chaps. 3-5 – we are concerned with parsing. In these chapters we present at first a deep parser – very domain and text limited – and then a less constrained, more versatile, derived scaled version. The deep parser only works topdown and has the goal to identify ungrammatical sentences. The other
parser works in both directions: and produces a semantically complete and consistent DAG (direct acyclic graph) like representation. Both parsers use the same lexical resources and rule modules. They also both take advantage of a “shallow” parser where tagging and chunking takes place. The output of the shallow parser is used to help lookahead mechanisms work appropriately at given critical constituency boundaries: to detect where possible structural ambiguity may come up. Its output can also be used as a final backoff strategy to recover from complete failure.

In these three chapters we also develop a comparison with Dependency parsers and discuss their advantages and deficiencies. Then in chapter 4, we discuss at length why we regard statistically based parsing ill-founded and speak in favour of linguistically-based symbolic parsing – where however statistics can play a minor role. Eventually, in chapter 5, we present our deep parser and the rule modules.

The following chapters deal respectively with: quantification – chapt. 6 -, semantic shallow interpretation – chapt. 7 -, pronominally based discourse anaphora – chapt. 8. Chapter 9 show the system used fruitfully in a shallow version for summarization.

We decided to include the system in its various implementations in a CD-Rom attached to the book. GETARUN comes in three versions:

- Version 1. Complete Getarun – performs a complete analysis of a text from tokenization to discourse structure. This version also supports the fully topdown parser for grammaticality checking. It also implements question answering with a generator;

- Version 2. Partial Getarun – also performs a complete analysis but does it in a fully bottomup version, only checking for broad semantic constraints. No temporal reasoning, no logical form, no semantic discourse model. It builds a fully indexed augmented dependency structure which is then used to produce a level of informational structure. This is used to produce discourse relations and discourse structures. It is also used to evaluate entailment relations;

- Version 3. Shallow Getarun – can be used to do sentence extraction on the basis only of tagging and local discourse perusal based on discourse markers. It can also be used to do the same thing on a shallow version of the Partial Getarun, which we called Deep Summarization. It also implements a version of Question Answering based on sentence extraction as best candidate answers.
INTRODUCTION

1. THE SYSTEM GETARUNS AND THE BOOKS

Getaruns (General Text And Reference UNderstander System) is a system for text understanding. The aim of the system is to build a model world where relations and entities introduced and referred to in the text are asserted, searched for and ranked according to their relevance. In addition to that, the system is able to generate text, in the form of answers to queries, and in the form of short paraphrases or summaries of the input text(s). In some cases, it can also generate stories and Questions and Answers randomly from a plan and a Discourse Model.

GETARUNS is a general multilingual text and reference understander which represents a linguistically based approach to text understanding and embodies a number of general strategies on how to implement linguistic principles in a running system. The system addresses one main issue: the need to restrict access to extralinguistic knowledge of the world by contextual reasoning, i.e. reasoning from linguistically available cues.

Another important issue addressed by the system is multilinguality. In GETARUNS the user may switch from one language to another by simply unloading the current lexicon and uploading the lexicon for the new language: at present Italian, German and English are implemented. Multilinguality has been implemented to support the theoretical linguistic subdivision of Universal Grammar into a Core and a Peripheral set of rules. The system is organized around another fundamental assumption: the architecture of such a system must be modular thus requiring a pipeline of sequential feeding processes of information, each module providing one chunk of knowledge, backtracking being barred at intermodular level and allowed only within each single module. The architecture of the system is organized in such a way as to allow feedback into the parser from Anaphoric Binding: however, when pronominals have been finally bound or left free no more changes are allowed on the f-structure output of the parser.

Thus we can think of the system as being subdivided into two main meta-modules or levels: Low Level System, containing all modules that operate at Sentence Level; High Level System, containing all the modules that operate at Discourse and Text Level by updating the Discourse Model.

The books are organized as an experimental exercise: they contain both theoretical background and the output of the system, GETARUNS that enacts and applies the theory. The architecture of the system is strictly related to the structure of the books. To better describe it, we decided to dedicate one book to the lower level part of the system and another book to the higher level system components. In this way, each component or module is presented in at least one chapter of the book.

Thus, we can think of the book as being organized around two scientifically distinct but in fact strictly interrelated fields of research:
• sentence level linguistic phenomena
• text or discourse level linguistic phenomena

the former to be described by means of grammatical theories, the latter requiring the
intervention of extralinguistic knowledge, i.e. knowledge of the world. This distinction is
usually drawn for scientific purposes and is obviously an artificial one: the sentence being at
the same time the smallest domain at which rigorous linguistic analysis can hopefully be
applied; but also the basic complete semantic unit whereby meaning can be conveyed,
depending on the text/discourse context. We are aware of the fact that this subdivision is
mainly wrought out for scientific reasons and does not really imply that such a neat
subdivision of tasks can be actually envisaged in real text processing. As shall be discussed in
detail in the books, semantic issues need to be tackled already at the beginning. This
notwithstanding, the separation has its own “raison d’etre” and we will try to validate it in the
books.

Book 1 – or the current book – addresses sentence grammar or what is usually referred to
as such by theoretical linguists. It does it by dividing up – somewhat ideally and sometimes
arbitrarily – what must or needs to be computed at sentence level from what need not or
cannot be computed at the same level, and consequently belongs to discourse grammar. In
that sense, the subdivision is not totally an arbitrary one, even though overlappings are normal
cases and will be discussed where needed.

The book also indirectly does another (un)intended subdivision: the one existing between
syntax and semantics. Again, it would be impossible not to deal with semantically related
issues when talking about syntax or the lexicon. However, semantics with uppercase S, is
only treated in Book 2 – already published – where discourse and text level grammar is
tackled.

So eventually, this book deals with lexicon, morphology, tagging, treebanks, parsing,
quantifiers and anaphoric or pronominal binding. In other words, all that concerns the level of
sentence grammar in a computational environment.

An important contribution the books make is the argument against the simplistic idea that
texts are a “bag of words” or that they can be processed in a satisfactory way using treebanks
derived statistical approaches. Not that treebanks are useless as sources of grammatical
information: as will be discussed in a chapter of the book, this does not support the statement
that all the grammar there is to learn is contained in a single treebank.

Additionally, it cannot be proven that statistics and “bag of words” approaches are
useless for NLP tasks. On the contrary, in some cases they constitute the only appropriate and
sensible approach – and more than one chapter will discuss at length the pros and cons. The
question is just wrongly posed: statistics cannot be treated as a panacea for all problems
raised by half a century of linguistic studies and represented by a(ny) text.

Sentence level parsing covers in our perspective all the issues tackled in this book. In this
sense, it speaks against those approaches – the majority in nowadays computational
linguistics – that reduce sentence level parsing to a phrase structure parenthesized
representation problem, with word tags and constituency labels in the style proposed and
made into a standard de facto by the Penn Treebank initiative. Nor can it be represented by
Dependency Structure with or without grammatical relation labels.

Sentence level Grammar – as has been purported in linguistic theories – takes care of all
grammatical and linguistic relations that belong to that level. Knowledge of the world and
semantic disambiguation do not interfere with the rules of sentence grammar, and can be thought of as a separate level of computation, provided that the lexicon be structured in such a way to allow such a subdivision of tasks.

For that reason, the first chapter is devoted to the Lexicon and to a linguistically-based way to derive principled rules that create lexical entries for Out of Vocabulary words when needed on the basis of a fixed number of syntactic, semantic and conceptual lexical types along the lines proposed by FrameNet and other similar efforts.

The second chapter presents our work on a treebank of Italian and the way to produce a conversion algorithm to Dependency Structures (almost) effortlessly.

In the following three chapters – the central part of this book – Chaps. 3-5 – we are concerned with parsing. In these chapters we present at first the Partial parser a less constrained, more versatile, derived scaled version of the the Deep parser – very domain and text limited. The Deep parser only works topdown and has the goal to identify ungrammatical sentences. The Partial parser works in both directions: and produces a semantically complete and consistent DAG (direct acyclic graph) like representation. Both parsers use the same lexical resources and rule modules. They also both take advantage of a “shallow” parser where tagging and chunking takes place. The output of the shallow parser is used basically to help lookahead mechanisms work appropriately at given critical constituency boundaries: to detect where possible structural ambiguity may come up. Its output can also be used as a final backoff strategy to recover from complete failure.

In these three chapters we also develop a comparison with Dependency parsers and discuss their advantages and deficiencies. Then in chapter 4, we discuss at length why statistically based parsing is ill-founded and should be dismissed in favour of linguistically-based symbolic parsing – where however statistics can play a minor role. Eventually, in chapter 5, we present our deep parser and the rule modules with examples and code.

The following chapters deal respectively with: quantification – chapt. 6 -, semantic shallow interpretation – chapt. 7 -, pronominally based discourse anaphora – chapt. 8. Chapter 9 has been included to show that the system can be used fruitfully in a shallow version, by doing away with some of its components: indeed, in this version the quantity of semantic processing and the ensuing understanding achieved will be drastically reduced. Here below the chapters are presented in more detail.

We decided to include the system in its various implementation in a CD-Rom attached to the book. GETARUNS comes in three versions:

- Version 1. Complete Getaruns – performs a complete analysis of a text from tokenization to discourse structure. This version also supports the fully topdown parser for grammaticality checking. It also implements question answering with a generator;
- Version 2. Partial Getaruns – also performs a complete analysis but does it in a fully bottomup version, only checking for broad semantic constraints. No temporal reasoning, no logical form, no semantic discourse model. It builds a fully indexed augmented dependency structure which is then used to produce a level of informational structure. This is used to produce discourse relations and discourse structures. It is also used to evaluate entailment relations;
- Version 3. Shallow Getaruns – can be used to do sentence extraction on the basis only of tagging and local discourse perusal based on discourse markers. It can also be
used to do the same thing on a shallow version of the Partial Getaruns, which we
called Deep Summarization. It also implements a version of Question Answering
based on sentence extraction as best candidate answers.

2. **BOOK I: LOW LEVEL SYSTEM**

2.1. **Parser**

The parser is based on LFG theoretical framework and has a highly interleaved modular
structure. Basic grammatical representation modules are the Lexicon, and grammar rules to
bind C-structure and F-structure which is internally represented as a graph. The Parser uses
Prolog Horn Clauses and exploits the properties of Prolog as to its general parsing strategy -
topdown, depth-first - and makes backtracking naturally available.

The internal architecture of the parser is organized on the basic idea of Universal
Grammar, i.e. on the well-known fact that all languages share a common core grammar and
may vary at the periphery: internal differences are predicted by principles. The grammar
allows the specification of linguistic rules in a highly declarative mode: it works topdown and
by making a heavy use of linguistic knowledge may achieve an almost complete deterministic
policy. Principles are scattered throughout the grammar so that they can be made operative as
soon as a given rule is entered by the parser.

In particular, a rule may belong either to a set of languages, e.g. Romance or Germanic,
or to a subset thereof, like English or Italian, thus becoming a peripheral rule. Rules are
activated at startup and whenever a switch is being operated by the user, by means of logical
flags appropriately inserted in the right hand side of the rule. No flags are required for rules
belonging to the common core grammar.

Some such rules include the following ones: for languages like Italian and Spanish, a
Subject NP may be an empty category, either a referential little pro or an expletive pronoun;
Subject NPs may be freely inverted in postverbal position, i.e. preverbal NP is an empty
category in these cases. For languages like Italian and French, PP or adverbial adjuncts may
intervene between Verb and Object NP; adjectival modifiers may be taken to the right of their
head Noun. For languages like English and German, tense and mood may be computed in CP
internal position, when taking the auxiliary or the modal verb. English almost freely allows an
empty Complementizer for finite complement and relative clauses, and negation requires do-
support.

Syntactic and semantic information is accessed and used as soon as possible: in
particular, both categorial and subcategorization information attached to predicates in the
lexicon are extracted as soon as the main predicate is processed, be it adjective, noun or verb,
and are used to subsequently restrict the number of possible structures to be built. Adjuncts
are computed by semantic compatibility tests on the basis of selectional restrictions of main
predicates and adjuncts' heads.

Syntactic rules are built according to some chomskian paradigm with CP-IP functional
maximal projections; however, the general underlying theoretical framework is cast into LFG
theory. Thus, we build and process syntactic phenomena like wh- movement before building
f-structure representations, where only anaphoric binding for pronouns takes place.
The parser makes use of Parsing Strategies and achieves a strong determinism thanks to a Lookahead mechanism, which has been finely tuned for each language. Lookahead may look the input stack for up to three symbols in sequence. The Hybrid architecture makes use of shallow c-structure information to predict when the verb is ready to be processed top-down.

2.2. Quantifier Raising

Since we know that quantifiers and quantified NPs usually take scope at propositional and NP level, we assume f-structure to be an adequate level of representation in which quantifier scope can be computed. In this we partially follow Halvorsen’s proposal, which however requires a further mapping from f-structures to s-structures in order to do that. We proceed as follows: after assigning Q-Markers to quantifiers and quantified NPs and adding this information as attribute-value pair at f-structure, we perform Quantifier Raising by traversing f-structure from the quantifier node up, until we reach a propositional or NP node. At that level we deposit a Quantifier-Operator (Q-Op), in an attribute that has a list as its value. Once Q-Ops have been produced, we are in a position to assign quantifier scope. In case more than one Q-Op is present in the list, the algorithm simply reorders the operators according to their quantifying force, and/or to grammatical function. Otherwise, a search downward is performed in the f-structure for other q-ops. When some q-marker is found another attribute-value pair is added at Pred level indicating a quantified interpretation. Quantifier scope may affect number and ultimately semantic interpretation.

2.3. The Binding Module

The output of grammatical modules is fed then onto the Binding Module (BM) which activates an algorithm for anaphoric binding in LFG terms using f-structures as domains and grammatical functions as entry points into the structure. Pronominals are internally decomposed into a feature matrix which is made visible to the Binding Algorithm (BA) and allows for the activation of different search strategies into f-structure domains. Antecedents for pronouns are ranked according to grammatical function, semantic role, inherent features and their position at f-structure. Special devices are required for empty pronouns contained in a subordinate clause which have an ambiguous context, i.e. there are two or more possible antecedents available in the main clause. Also split antecedents trigger special search strategies in order to evaluate the possible set of antecedents in the appropriate f-structure domain. Special care is paid to pronominals bound by quantifiers or quantified NPs. The output of the BA is then passed on to an Interpretation Module which operates locally in order to spot the presence of conditions for Specific or Arbitrary Reading for pronominal expressions.

Finally, this information is added into the original f-structure graph and then passed on to the High Level System.
2.4. Anaphora Resolution

Anaphoric binding of free pronouns takes as input Discourse level information which is computed by a Module of Discourse Anaphora (MDA) and decides on the basis of semantic categories attached to predicates and arguments of predicates whether to bind a pronoun to the locally available antecedent or to the discourse level one.

Discourse Anaphora is computed by a system which is very close to the ones available in literature on the same topic, and presented by C.Sidner and B.Webber in their publications. Definite descriptions are computed by means of locally available information and, but only when required, by tapping external knowledge sources. However, a set of default rules are activated in lack of such knowledge which work simply on the basis of grammatical and semantic information.

This level of representation works on the basis of a list of candidates or possible arguments of discourse which includes all external pronouns and referential expressions. The algorithm creates a Weighted List of Candidate Arguments of Discourse (WLCAD).

The MDA detects Main, Secondary, Potential and Expected TOPIC in each utterance by weighting the list of external pronouns and of referential expressions made available by the grammatical representation. After the first utterance, the MDA tries to take advantage of discourse level internal cohesion and coherence mechanisms in order to check whether a given topic is reintroduced in the following text as a pronoun, a nominal substitute, a deictic and so on. The MDA is a finite state automaton that works strictly on the basis of two adjacent utterances; it has a set of six states available: Continue, Change, Resume, Shifting, Retaining, Continue_Analyze. Special inferential mechanisms are activated at this local level by the presence of nonthematic functions, i.e. SUBJECTs or OBJECTs of copulative and other verbs which do not assign a semantic role to their arguments. The algorithm takes care of bridging descriptions using the Discourse Model and external world knowledge. The output of the MDA is a list of all Topics in each utterance and a state.

2.5. Computing Information Structure

As a first step towards a semantic representation, the system builds Information Structure at clause level. Each utterance is decomposed into separate clauses to account for coordination and subordination. The output is a set of values associated to labels such as Clause Type, Point of View, Factivity, Change in State of the World, Relevance and Discourse Relations. Point of View may vary between Subjective, Subjective_Extensional, Subjective_Intensional, Objective; Change may be Null, Early, Culminated, Gradual and Setting; Relevance may be Foreground and Background. All these values are computed mainly on the basis of semantic, aspectual and temporal features associated with main verbal predicate at f-structure for each clause.

Information structure is passed, then, on to the Semantic Module and Discourse Relation Reasoning Module. In particular, the latter computes Discourse Relations on the basis of information structure, temporal interpretation and the model. The output of this module is a set of values, which include: Narration, Elaboration, Description, Parallel, Explanation, Cause, Result. The Discourse Model is consequently a database in which inheritance is easily computed at the end of the text. In addition, we created a mechanism of Relevance Score
assignment which takes into account topicality and discourse relations in which a given entity was involved by computing a score with different weights. In this way, all entities appearing in the text are finally sorted in descending order according to their relevance, with the list of facts and situations each one shared.

2.6. Shallow and Partial Text Processing for Linguistic Information Extraction and Summarization

In this chapter we present a morphological analyser for Italian, a Tagger and the accompanying statistical and syntactically based Disambiguation algorithm; a Shallow Multilevel Syntactic Parser and an adaptation of the Higher Level Discourse and Semantic components of GETARUNS to fit into a Partial Semantic Parser. This is both referred to grammar checking and the summarization problem. In the case of summarization, sentence extraction is viewed as a first step in the process of text summarization and as a means to achieve a reasonable reduction of information load when attempting text understanding tasks. Linguistic strategies can be fruitfully applied, as will be discussed, in order to postpone the use of a knowledge representation language for terminological reasoning, until the size of the text has been reduced to a percent of the original value - typically 25%.

3. BOOK II: HIGH LEVEL SYSTEM

3.1. Semantic Interpretation

Semantic interpretation is carried out in two phases: a static phase and a dynamic phase in the Semantic Interpreter.

The static phase starts by building wellformed-formula from a Logical Form where individuals and sets are taken care of to compute quantifier scope and build logical representations which are eventually linked together at propositional level, where temporal indices are added to event/state quantified operators. At this level of computation we are still using syntactic indices; these are then turned into semantic indices by the second phase which takes as input the representation into wellformed-formulas realised by Logical Form. This module turns it into a situation semantics structure which contain relations, their arguments and their modifiers and adjuncts. At this level all syntactic indices are turned into semantic identifiers by accessing first the Discourse Model and taking into account the output of the anaphora resolution algorithm. Empty variables associated to syntactically controlled arguments are discarded and are substituted by their controller’s semantic index. Arguments may also be added in case of passive agentless structures in order to preserve the integrity of predicate-argument structures as required by lexical and semantic forms. Other notable operations carried out at this level of computation regard idiomatic expressions and other locutions which discard pieces of logical representation related to semantically empty lexical elements and conflates them with their head – as for instance in “there_be” verbal predicate.
3.2. The World or Discourse Model

Reasoning is carried out in the Discourse Model into several separate modules: Discourse Relations, Temporal Reasoning and Situation Semantics Representations. Discourse Relations are built for each proposition by means of semantic information associated to main predicates and by the output of the Temporal Reasoner.

These information are eventually passed to the Semantic Module where a representation in terms of situation semantics is computed: each proposition has been translated by the Logical Form algorithm into a fact or a situation according to semantic conditions, and is now completed by a polarity, a spatio-temporal location constituted by semantic indices inherited from the previous modules. The final ontology is made up by locations, individual entities which may also be treated as classes and set with a given cardinality, facts and situations about entities which may be attributes or roles according to their semantics. Basically they are computed from grammatical functions and semantic roles associated to the argument or adjunct by the previous parse.

The Knowledge Base thus coincides with the dynamically built Discourse Model where extensionality and intensionality is computed according to the actual factuality value of the main predicates in a given context. The External World Knowledge Base is tapped only when needed, and in particular whenever a singular definite NP is introduced with a special topicality in the discourse and no antecedent is available in the Model.

3.3. Spatio-Temporal Reasoning

Temporal aspectual interpretation is carried out intrasententially and intersententially on the output of f-structure representation and computes a complete interval logic coindexation of events and states at clause level on the basis of Reichenbach's tripartite model for temporal calculus. This is then passed on to the semantic module to serve for the understanding of Temporal Relations, which are cast on the basis of J.Allen's system for temporal logic notation, the Temporal Reasoner. According to the input received, the system may assign an interpretation in terms of discourse relations by means of a mechanism of Temporal Anaphora and Focus which takes into account information related to semantic roles associated with arguments of the predicates and matches them at first with the ones available in the previous proposition: a temporal anaphora or a new temporal focus is thus the outcome both of semantic and temporal reasoning.

3.4. Discourse Structure, Discourse Relations, and Point of View

Informational structure is used to build up discourse structures at clause level, by assigning each clause to an attachment point at the Same Level, or Down a previous clause. In case a new Topic is introduced or something happens in the world that interrupts the previous sequence or text stretch, an UP movement is used and a new discourse structure is begun. These structures may be used to bind deictic propositional pronouns like "this", "that", "it" in English. Discourse Relations are used to build discourse structure in an efficient way. Point of
view is switched from the Narrator's to one or other character according to Domain information, which may be either Objective or Subjective, and to Topicality: Main Topic and Subjective Domain will trigger the assignment of Point-of-View to that character.

3.5. Causal Discourse Relations

This Chapter is devoted to Causal Discourse Relations, a special subtype of Discourse Relations discussed in the previous chapter. Causal Relations are usually signalled by the presence of specific discourse markers. However when these markers are missing or are highly ambiguous problems may arise as to the correct encoding of the Discourse relation. The chapter discusses in depth such cases and presents a working algorithm again based on a shallow version of GETARUNS, and tested on available annotated corpora.

3.6. Text Generation

In this Chapter we review current unification-based generation systems and present our module for text generation which uses a Planner, Discourse Relations, Temporal Reasoning and Conceptual Representation to generate answers to questions in Italian and English.

3.7. Linguistically-Based Semantic Evaluation for Text Entailment

Semantic processing is usually done in presence of meaning ambiguity, disambiguation processes being usually left for the final reasoning modules. What is being presented in this chapter is our approach to Text Entailment, a technique for deciding whether two text excerpts convey the same meaning or are semantically entailed in one another in an asymmetric way. To come up with such a decision, all lexical semantic inferences are fired in order to establish coherence and cohesion relations between main predicates and arguments of the two texts. The evaluation is carried out by associating penalties or rewards to synonymous and entailed linguistic material and by inducing failure whenever a contradiction ensues. In this chapter we present the shallow or robust complete parser which however will receive a much extended treatment in the dedicated chapters of this book.

3.8. Reasoning from a Text with Conceptual Representations and Discourse Model for Question/Answering and Text Paraphrasing

Eventually, we present the terminological reasoning system which takes as input the Discourse Model built by the Text Understanding System and uses lexical information encoded as Conceptual Representations to allow for queries about spatiotemporal locations of entities in the world of the text being analysed. This level of computation is made feasible by the presence of a fully consistent semantic representation. The important part of this Chapter is constituted by the generalized procedures that allow the system to instantiate all the objects
needed by KL-ONE to build up its internal database for inferencing and consistency checking. These procedures testify to the solidity of the overall project, that of passing from linguistic representations onto knowledge representations automatically.

4. State of the Art and Other Systems

There is a large number of well-documented systems in the literature which compare well with GETARUNS, in particular TACITUS and KERNEL. However we will only comment on KERNEL as presented by Palmer et al. (1993). KERNEL’s architecture is similar to ours in that syntactic, semantic and pragmatic tasks are segregated into separate processing modules but they are allowed to communicate: in particular many linguistic phenomena requiring interaction with knowledge representation and reasoning are channeled though lexical semantics.

KERNEL performs its analysis in two stages: first syntactic parsing “which has limited access to shallow semantic constraints for parse disambiguation” (ibid., 20) and second integrated semantic and pragmatic processing which has constrained access to external knowledge sources. However, syntactic processing is not itself performed by a context-sensitive semantically guided parser: it is basically a context-free grammar with restrictions, a grammar formalism called restriction grammar. In turn each clause is then translated or mapped into a functional-like representation with attribute-value pairs called ISR. Semantic interpretation is performed while building up ISR and requires among other things recovering unexpressed constituents like subjects or non obligatory prepositional phrases, as well as implicit but essential and sometimes obligatory arguments of a given verb predicate when used in its nominalized form. Noun phrase analysis in addition has a separate mechanism from clause analysis in that the former but not the latter allows for referenc e resolution. This requires a search for a likely discourse referent. Here comes another important limitation in the system: since each constituent is interpreted in the order in which it is logically built, there are cases in which a pronoun in the matrix clause has its antecedent in the subordinate which however is not yet available for use in the interpretation process (ibid., 62). This rigidity of the system could be overcome in case the system could choose to delay reference resolution of all nouns as for instance in CANDIDE (see Pollack & Pereira). Actually the real problem with KERNEL is its inability to deal with adjunct semantic interpretation (ibid., 56) but only with the interpretation of arguments of a given predicate. In other words, there is no attempt at disambiguating prepositional phrase attachment for nominal structures, hence the requirement to apply directly pragmatic knowledge and reasoning or inferencing from discourse model and/or knowledge representation for the domain. Very much in the same way is resolved the problem of semantic interpretation: there is a first pass for semantic role assignment which is locally determined by some governing predicate. Then a second pass is performed that looks for inferred role fillers obtained as a by-product of reference resolution (ibid., 29). This way of interpreting noun phrases resembles very closely ours, in that we do a local interpretation first and then the current description may be found coreferring to some entity already in the domain model, thus automatically inheriting all previously asserted properties. KERNEL does not interpret spatial location but only temporal ones which are computed in a manner very similar to ours. Reference resolution is also handled in the same way except that we do not
concentrate pronoun resolution in one single module as they do. They use a focus list which is a list of referents ordered by saliency: in addition, the entire previous utterance is regarded as the preferred potential focus. Reference resolution is also used to look for essential roles which have no syntactic realization in the utterance, which is recognized as a case of semantically implicit reference. In our case, the unexpressed agent role of a passive sentence is filled at f-structure level by an indefinite existential dummy quantifiers which is then used by the semantic interpreter to search the domain model for a suitable description. The conclusion the authors draw is that more inter-module communication is called for in order to cope with ambiguity at syntactic analysis level: but at present there are no choice point at which syntax is encouraged to query the discourse context about its structure. In this way, no contextual reasoning is performed.

5. THE STORIES

The theoretical basis of this book has been tested by means of GETARUNS on a number of texts both in Italian and in English. We chose a selection of them that we report here below. More examples and texts are presented in the following chapters. Story 1 is the story of the three little pigs, an abridged version of the original much longer story, which we report partially in the first 24 utterances alone though - the most interesting part of the story - as Story 3. Then there is an abridged version of a newspaper article on a politician, Avveduti, which we report as Story 2. These are analyzed as Italian texts. We have also reported a number of English texts which are mainly taken from literary works. At the end we report an abridged version of an EC Council directive. We add a rough literal translation of each Italian text into English: translations have not been used to tune the system, however.

5.1. Italian Texts

La Storia dei Tre Porcellini / The Story of the Three Little Pigs

C'erano una volta tre fratelli porcellini che vivevano felici nella campagna. Nello stesso luogo però viveva anche un terribile lupo che si nutriva proprio di porcellini grassi e teneri. Questi allora, per proteggersi dal lupo, decisero di costruirsi ciascuno una casetta. Il maggiore, Jimmy che era saggio, lavorava di buona lena e costrui la sua casetta con solidi mattoni e cemento. Gli altri, Timmy e Tommy, pigri se la sbrigarono in fretta costruendo le loro casette con la paglia e con pezzetti di legno. I due porcellini pigri passavano le loro giornate suonando e cantando una canzone che diceva: chi ha paura del lupo cattivo. Ma ecco che improvvisamente il lupo apparve alle loro spalle. Aiuto, aiuto, gridarono i due porcellini e cominciarono a correre più veloci che potevano verso la loro casetta per sfuggire al terribile lupo. Questo intanto si leccava già i baffi pensando al suo prossimo pasto così invitante e saporito. Finalmente i porcellini riuscirono a raggiungere la loro casetta e vi si chiusero dentro sbarrando la porta. Dalla finestra cominciarono a deridere il lupo cantando la solita canzoncina: chi ha paura del lupo cattivo. Il lupo stava intanto pensando al modo di penetrare
La Storia di Avveduti /The Story of Avveduti

Gli piaceva parlare del suocero come di una facile occasione mancata che chiunque altro avrebbe sfruttato ma che lui, Avveduti, preferiva lasciare perdere. Solo verso il 1950 decise di accettare un posto nella organizzazione della fiera di Verona. Lo nominarono delegato cioè una specie di funzionario viaggiante con incarichi diplomatici di tenere i rapporti con le delegazioni commerciali, curare i produttori stranieri, le grandi ditte, la stampa. Questo era un compito che corrispondeva bene alla sua vocazione e nel quale Avveduti sapeva giostrare con notevole agilità. Quando il suocero morì, egli non perse il posto. A Verona il collegio di Alberti lo aveva ereditato Trabucchi e col collegio aveva ereditato la presidenza della fiera. Trabucchi continuò a valersi della collaborazione di Avveduti. L'ex-ufficiale del Novara-Cavalleria gli era simpatico. La sua distinzione lo impressionava. Lo confermò nell'incarico alla fiera. Avveduti funzionava benissimo come segretario particolare. Sapeva mobilitare prefetti e questori. Tutti gli invidiavano il suo segretario particolare.

And this is a rough literal translation,

Until three years ago, Franco Avveduti never got involved with the world of public administration. As a beaurocrat, he was an immigrant who came from outside. Son of a respected family, when he was 20, he decided to enroll at the Military Academy of Cavalry. He was a good cadet with excellent qualifications. Later on, he became a successful officer. Then in 1945, Avveduti resigned from the army. The military had disappointed him. Laid down his uniform, Avveduti enrolled in the University. In 1947 he graduated and in 1948 he became a solicitor. In the meantime, in Verona, he met Paola, the daughter of Antonio Alberti, powerful Christian Democrat senator, and he married her. The senator might be regarded the most powerful political man in Verona. The voters sent him to parliament covering him literally with preferential votes. At the suite of Alberti, who had become vicepresident of the Senate, Franco Avveduti in the immediate postwar moved to Rome. However at the beginning he kept himself aloof from the sphere of interest of his father-in-law. He liked to talk about his father-in-law as if he were an easy chance missed which anybody else would have exploited but which himself, Avveduti, preferred to let go. Only towards 1950 he grudgingly ended up by accepting a position in the committee of the Verona Fair. They appointed him delegate, i.e. a kind of travelling officer with diplomatic charges, to have relations with commercial committees, take care of foreign producers, the big corporations, the press. This was a task which fitted well with his inclination, and where Avveduti could joust with remarkable ease. In Verona, Alberti's college had been inherited by Trabucchi, and with the college he had inherited the chair of the Fair. Trabucchi continued availing himself of Avveduti's collaboration. To him, the Novara Cavalry officer was nice. His distinction impressed him. He confirmed him in the position at the Fair. Avveduti worked very well as special secretary. He knew how to mobilize prefects and questors. Everybody envied him his special secretary.

La Storia dei Tre Porcellini Rivisitata / The Story of the Little Pigs Revisited

Questa è la storia di tre porcellini che andarono per il mondo a cercare fortuna. I loro nomi erano Timmy, suonatore di flauto, Tommy, violinista e Jimmy, grande lavoratore. Giunti in un bel bosco, decisero di costruire ognuno una comoda casetta. A Timmy non piaceva per niente lavorare così pensò di costruirsi rapidamente una capanna di paglia. In breve la casetta fu pronta e Timmy decise allora di andare a vedere che cosa stavano facendo i suoi fratellini. Incontrò dapprima Tommy il violinista. Anche lui non aveva molta voglia di faticare così costruiva con dei pezzi di legno una semplice casetta. Ben presto anche la casa di
legno fu pronta. Come quella di paglia, non era certo molto resistente. Ma i due porcellini scansafatiche se la erano sbrigata in poco tempo ed ora potevano tranquillamente divertirsi. Mentre Timmy suonava il flauto, Tommy lo accompagnava con il suo violino e insieme se la spassavano allegramente. Poi stanchi di fare baldoria, decisero di andare a vedere che cosa stava facendo il loro fratellino. Si misero in cammino e ben presto raggiunsero Jimmy. Il bravo porcellino stava costruendo anche lui la sua casetta. Ma poiché Jimmy era previdente e non aveva paura di lavorare sodo, la costruiva con mattoni e cemento. Jimmy voleva una casa robusta perché sapeva che il lupo cattivo viveva nel bosco vicino. Quando i due pigri porcellini videro Jimmy impegnato nel suo duro lavoro, si misero a ridere a crepapelle. Ma quei due sciocchi porcellini non pensavano al pericolo. Così continuarono a prendere in giro il saggio Jimmy, canticchiando e suonando con il flauto e il violino. I due porcellini, sempre suonando e ballando, tornarono ciascuno alla propria fragile casetta. Ma appena Timmy aprì la porta, sbucò fuori dal bosco il lupo cattivo. Il porcellino lo vide e tremante di paura si chiuse immediatamente in casa. Il lupo cattivo cominciò a chiamarlo. "Apri la porta e fammi entrare nella tua casetta di paglia."

And this is a rough literal translation,

This is the story of three little pigs who went around the world seeking their fortune. Their names were Timmy, flute player, Tommy, violinist, Jimmy, great worker. As they reached a nice wood, they decided to build each a comfortable little house. Timmy didn't like working at all, so he thought to build quickly a straw hut. Soon the little house was ready and Timmy decided then to go and see what his little brothers were doing. At first he met Tommy the violinist. Also he, himself, did not have much wish to toil, so he was building a simple little house with sticks of wood. Very soon, also the house of wood was ready. Like that of straw, it was not very resistant. But the two little pigs lazy had managed to finish their work in a short time and now they could enjoy themselves freely. While Timmy was playing the flute, Tommy accompanied him with his violin and together they were having a lot of fun. Then, tired to make merry, they decided to go and see what their little brother was doing. They started walking and soon they reached Jimmy. The clever little pig was building a little house. But since Jimmy was farsighted and did not fear working hard, he built it with bricks and cement. Jimmy wanted a sturdy house because he knew that the big bad wolf lived in the woods nearby. When the two lazy little pigs saw Jimmy busy in his hard work, they started laughing out of their wits. But those two silly little pigs didn't think about the danger. So they continued making a fool of the wise Jimmy, singing and playing the violin and the flute. Each of the two little pigs, always singing and playing, went back to its fragile little house. But as soon as Timmy opened the door, the big bad wolf got out of the woods. The little pig saw it and trembling with fear locked himself inside the house immediately. The big bad wolf started calling him. "Open the door and let me in your little house of straw."
5.2. English Texts

*Virginia Woolf's Excerpts*

**Text 1**

John gave Mary a rose. She took it and put it in her hair. She knew that she had been given a present, something precious. When Steve faced them saying, "are you enjoying yourselves?". It was horrible! It was shocking! Not for herself. She felt only hostility and his determination to ruin that wonderful moment. John smiled and went away embarrassed.

**Text 2**

The three friends went all outdoors. As they were walking in the garden, John said to himself, "Sara will marry that man", without any resentment. Richard would marry Sara. He felt strongly about that. She was the right person for a man like Richard. For himself he was absurd. His demands upon Sara were absurd. She would have accepted him still if he had been less absurd. Richard began to sing.

**Text 3**

Mary picked up the phone and called Jason. Her husband, she thought, would have considered such a move as untruthful and utterly base. Perhaps there was something bad in herself that she could not help but do the wrong thing at the wrong time. Jason answered immediately.

*Psychological Statements: At the Restaurant*

**Text 1**

John went into a restaurant. There was a table in the corner. The waiter took the order. The atmosphere was warm and friendly. He began to read his book.

*A Legal Text*

Council directive of July 1985 on the approximation of the laws, regulations and administrative provisions of the Member States concerning liability for defective products. The council of the European Communities has adopted this directive. Having regard to the proposal from the commission. Whereas approximation of the laws of the Member States concerning the liability of the producer for damage caused by the defectiveness of his products is necessary because the existing divergences may entail a differing degree of protection of the consumer against damage caused by a defective product to his health or property. Whereas liability without fault should apply only to movables which have been industrially produced. Whereas protection of the consumer requires that all the producers involved in the production process should be made liable in so far as their finished product, component part or any raw material supplied by them was defective. Whereas, to the extent that liability for nuclear injury or damage is already covered in all member states by adequate special rules, it has been possible to exclude damage of this type from the scope of this directive. Whereas, in situations where several persons are liable for the same damage, the protection of the consumer requires that the injured person should be able to claim full
compensation for the damage from any one of them. Producer means the manufacturer of a finished product, the producer of any raw material or the manufacturer of a component part and any person who, by putting his name, trade mark or other distinguishing feature on the product presents himself as its producer. The injured person shall be required to prove the damage, the defect and the causal relationship between defect and damage. This directive shall not apply to injury or damage arising from nuclear accidents and covered by international conventions ratified by the Member States. Where, as a result of the provisions of this directive, two or more persons are liable for the same damage. They shall be liable jointly and severally, without prejudice to the provisions of national law concerning the rights of contribution or recourse. A product is defective when it does not provide the safety which a person is entitled to expect, taking all circumstances into account. The liability of the producer arising from this directive may not, in relation to the injured person, be limited or excluded by a provision limiting his liability or exempting him from liability. Without prejudice to the liability of the producer any person who imports into the community a product for sale, hire or any form of distribution in the course of his business shall be deemed to be a producer within the meaning of this directive.

Whereas to protect the physical well-being and property of the consumer the defectiveness of the product should be determined by reference, not to its fitness for use, but to the lack of the safety which the public at large is entitled to expect. The liability of the producer may be reduced or disallowed when, having regard to all the circumstances, the damage is caused both by a defect in the product and by the fault of the injured person or any person for whom the injured person is responsible. However, if within three months of receiving the said information the commission does not advise the Member State concerned that it intends submitting such a proposal to the council, the Member State may take the proposed measure immediately. Any member state may provide that a producer's total liability for damage resulting from a death or personal injury and caused by identical items with the same defect shall be limited to an amount which may not be less than 70 million Ecu.

6. ACKNOWLEDGEMENTS


I must thank my colleague Dario Bianchi – who recently (September 2005) died - for providing a lot of ideas in the implementation of the system, and Emanuele Pianta who graduated and worked with me at the semantic interpretation. Ideas and implementation advances would have not been possible without the intervention of students at my Computational Linguistics courses which is the most valuable testbed for any theory.
Chapter 1

INDUCING FULLY SPECIFIED LEXICAL REPRESENTATIONS

SUMMARY

In this chapter we intend to test hypotheses on the nature of lexical knowledge and lexicon acquisition. The questions we pose ourselves in implementing the programs needed to simulate lexical knowledge and acquisition are approximately the following: what is the set of primitives required to build lexical descriptions which we call forms - in accordance with LFG - which make available to native speakers both syntactic and semantic information required to parse and recognize sentences of their mother tongue in the acquisition phase. Also, since we believe that low level inferences are made unconsciously by speakers, what information are required in order to automatically build a conceptual representation to be used by the inferencing mechanism?

In this sense, lexical forms are tightly connected to the nature of the parser and the inference mechanism which uses them. However, we believe the format of lexical forms should be universal and internally motivated, rather than externally conditioned.

The starting point will be FrameNet (hence FN) in its latest release 1.3, which is rightly regarded as the most reliable source of lexical knowledge for semantic and syntactic processing. FN is a lexical knowledge repository containing knowledge representation of the world allowing inference to be fired at different levels, close to a full-fledged ontology partly in the vein of WordNet.

In commenting FN peculiar features we will compare it to a number of other similar lexical resources: LCS (Lexical Conceptual Structures), WordNet, Levin’s WordList, PropBank and VerbNet. Each of these publicly available lexica contains some features that make it comparable to FN, even though they all lack its fine-grained overall data description. In particular, even though they are all handcrafted, they all allow some type of inference to be fired directly; some contain such an induction mechanism at the level of lexical structure, some other as additional superstructure.

In particular we will comment on Dorr’s Lexicon of Conceptual Structures which constitutes a valid companion to FN and other lexical resources. We will introduce the underlying theoretical framework in some detail seen that this is also what we have been using in our work on the lexicon of Italian. One feature in particular will be made object of
further scrutiny: the notion of Causality and the way it is represented at a lexical level. One section of the chapter will be devoted to explore this notion in detail and the way in which it is tackled in FN and LCS.

Eventually, we will describe our effort of building one such resource in late ’80s for Italian. In that effort we tried to reconcile knowledge of the world and linguistic semantic and conceptual information needed to produce highly structured lexical entries as the ones provided by FN and LCS. Besides presenting the algorithm and its linguistic components, we will highlight those features of our approach that make it different and, in some cases, computationally more perspicuous than the others. As will be shown, the premises we adopted can lead in some cases to a more complete and a richer representation.

1. INTRODUCTION

In this paper we will present a computational approach to the problem of lexical access based on FrameNet (hence FN), which may be regarded the lexical repository closest to a description of the world as it is. In this sense, FN is a database containing knowledge representation of the world which allows inference to be fired at different levels, close to a full-fledged ontology.

However, from a computational point of view, FN is considered a too sparse source of lexical data to serve as a useful tool in open domain, unrestricted text processing (see Green et al., 2004). The current version of FN is useful for a certain number of different text processing tasks, like using Frames and Frame Elements to produce more perspicuous semantic roles associated to the output of a dependency parser. Eventually there is always the possibility to extend it manually – if needed - to other lexical units which share/evoke the same concept/frame. This is certainly beneficial in case small texts in highly restricted domains are the target of the analysis.

In our perspective, however, lexical knowledge associated to a given predicate or lexical entry, is gathered from sentence level or propositional level syntactic/semantic structure and is constituted of the following linguistic items:

1. Entities, Properties and Events
2. Events or Situations
3. Participants (arguments)
4. Semantic Roles (the situation)
5. Perspective or Point of View
6. Temporal Extension of the Event

As a consequence of that, there is a number of issues that are strictly related to lexical acquisition that need explanation:

7. Meaning of lexical entry is related to the actual world or not
8. Events carry consequences on the state of affairs described
9. Relations of events to spatiotemporal locations of arguments may change or not
10. Complexity of lexical meaning contained in the lexical entry
Differently from the theory put forward by Fillmore in FN, we assume that lexical knowledge is not attached to single lexical units, which “evoke Frames”. Fillmore’s underlying theory is Construction Grammar (hence CxG) (see Croft, Kay et al., Fillmore et al., Hudson). CxG assumes that linguistic knowledge is all lexically fixed by grammatical constructions which are syntactic templates where form and content are coupled: no derivations and no semantic de-compositional processes are needed. Phonology, prosody, semantics, pragmatics and indeed syntax are part of grammatical constructions which are organized in a hierarchical set of models. However, if this theory were really computable and mentally or cognitively plausible, one would have to assume an inventory of some million different such grammatical constructions covering all possible semantically and pragmatically relevant combinations of atomic morphemic units with their corresponding sounds. CxG uses notions such as Default Inheritance and Complete Inheritance to induce generality in the network constituted by grammatical constructions: however, the model of CxG proposed by FN and more accepted seems the Usage-based model. This model is based on inductive learning, i.e. linguistic knowledge is acquired bottom-up from real sentences. Generalization on similar structures is obtained through use of recurrent constructions. In such a framework there is no Universal Grammar, no Typological Universals seen that mapping from form to content is strictly language and construction dependent. Eventually, no formal distinction between lexical (semantic) and syntactic structures.

Rather, we assume that computability is a cognitively sound gauge of the psycholinguistic plausibility of any theory: human languages are characterized by “Recursivity”, and make use of semantic types and compositionality of meaning allows all possible combinations to be computed smoothly. Words need to be computed w.r.t. their linguistic context which alone can guarantee the semantic representation to match with predicate-argument structure (hence PAS) in the lexicon. From PAS Frames can be evoked and Frame Elements be associated to arguments of the governing predicate.

Principles underlying CxG are listed in Holmes and Hudson (2001:2) and reported here below, where the authors match and combine principle formulated in Kay & Fillmore (the CG manifesto) and in Hudson’s Word Grammar:

- The goal of linguistic theory is "to account for the entirety of each language", including "noncore" patterns as well as the central core.
- No distinction is assumed (or found) between 'rules' and 'lexical items', so a linguistic theory must include "an explicit system of representation, capable of encoding economically and without loss of generalization all the constructions (or patterns) of the language, from the most idiomatic to the most general".
- The list of constructions is the database of "an explicit, nonderivational (constraint based) grammar", so the grammar is generative (explicit) but not derivational (transformational).
- Syntactic patterns are intimately bound to semantic ones so that "syntactic and semantic information is represented within a single feature structure"; each grammatical construction is "a conventional association of linguistic form and content".
- Complex patterns in sentence structure are generated by the interaction of a multiplicity of individually much simpler patterns. In CG the simpler patterns are called 'constructions', so the grammar must be able to integrate "both constructions
and the words, phrases and sentences of the language which they license - which we call 'constructs' ..." (It is true that the terms "construction" and "construct" have not been generally used in WG, but they apply perfectly to the very simple basic patterns of WG and the more complex patterns that they license. In both theories the term 'inherit' is used for the relation of a construct to its licensing constructions.)

- Semantic structures must show the fine grain of lexical semantics as well as the broader structures due to syntax; for example, the analysis of GIVE must include "a set with four members, each ... representing a minimal predication, consisting of a frame plus its participants or arguments ...". The semantic structure must accommodate pragmatic information such as illocutionary force (e.g. request for information) and presupposition (e.g. that the scene described is "incongruous" as in the famous What's X doing Y? construction).

In our perspective, syntactic structure is independently built and has independent properties and principles that help in the semantic interpretation process. Even though we are also lexicalist – as LFG is one such theory – we believe in the usefulness of lexical and syntactic rules and in their independence of semantic rules. Some of the rules depend on lexical representations, some others are part of any semantic theory and are applied at propositional level through some Logical Form representation.

This will be discussed in the following sections: Section 2 presents details of Frame theory and its main features. In the same section we also present similar resources available which describe other important elements of lexical semantic representation. Section 3 is devoted to discussion of the underlying linguistic framework motivating our approach. Section 4 of this chapter is dedicated to a preliminary proposal for Lexicon induction from semantic primitives and grammatical categorization. In this proposal we will point to shortcomings of FN and address them by referring to the Italian lexicon we built in the past.

We will assume that in order to construct a lexicon containing the information listed above, we should address the following linguistic categories:

- grammatical categories – derived from a categorization of reality into entities – nouns –, events – verbs and nominals -, attributes – adjectives, adverbials, and nouns;
- semantic categories, like negation, quantifiers;
- discourse level categories, like deictics, definiteness, conjunctions for coordination and subordination at propositional level;
- syntactic categories - encoding the arity of predicate-argument structures as they are interpreted in situations;
- aspectual categories – encoding the internal temporal structure of events (as expressed both by verbal and deverbal nominals);
- semantic conceptual categories – classifying types of events in relation to the (un)reality they encode;
- selectional restrictions – encoding the typicality of event participants in inherent semantic features as they are represented in an ontology or connected encyclopedic database of entities and their semantic interrelationships;
- grammatical constraints – encoding so-called syntactic and anaphoric binding constraints on arguments of predicate and dependent predicates only for propositional arguments, though.
This set of primitives has been used for lexical induction in a program that automatically produces full-fledged lexical representations in Italian. In Delmonte (1989) - but see also Carrier et al. 1993 - we presented a mapping algorithm that starting from a fixed number of classes and a syntactic encoding of argument structure could derive automatically via a certain number of linguistic rules, both grammatical function, semantic roles and conceptual representations. In conclusion, starting from FrameNet and its companion lexical resources available online, we want to test hypotheses on the nature of lexical knowledge acquisition and its function in language understanding and generation, by simulating its creation and usage at a computational level.

2. Frames and Syntactic-Semantic Representations

FrameNet is designed as an ontology of frames, i.e. representations of prototypical situations. Each frame provides a set of predicates (nouns, verbs or adjectives) by which it can be evoked and a set of semantic roles – or frame elements - which correspond to categories of entities or concepts that occur in the situation. Like other projects, FrameNet has predominantly concentrated on building a large manually annotated corpus. The corpus, a subset of the British National Corpus, currently contains about 135,000 instances of 795 frames. However, for role semantics to become relevant for language technology, robust and accurate methods for automatic semantic role assignment are needed. In recent years, a number of studies has investigated this task on the FrameNet corpus (Gildea and Jurafsky, 2002; Fleischman et al., 2003; Chen and Rambow, 2003). For each frame, the following information is given:

1. the name of frame,
2. a description (usually including example sentences),
3. a list of the lexical units (LUs) in the frame (if any), and
4. the names and descriptions of each frame element (FE), usually including example sentences.
5. Example sentences which are either taken from the Brown Corpus or the Penn Treebank and come with POS, Syntactic constituency labels and Grammatical Functions.

In turn Frame Elements

• may belong to the Core or the Periphery;
• they may be Extrathematic, Core-Unexpressed;

Of all the FEs or Semantic roles associated to example sentences, limited though to verb LUs, we have the following distribution in terms of Core, Periphery and Extrathematic:

• 61390 roles are core
• 6560 roles are extrathematic
• 16100 roles are peripheric
• finally, 1091 verbal entries are empty (there are no examples associated)
Semantic Roles which appear in all three types are the following ones,

ADDRESSEE, CARRIER, CAUSE, CONTAINER, DIRECTION, DISTANCE, EVIDENCE,, GOAL, INITIAL_STATE, INSTRUMENT, LOCATION, MEANS, MEDIUM, OUTCOME, PATH, PATH_SHAPE, PURPOSE, ROLE, SOURCE, STATE, VEHICLE

Semantic Roles which appear in only two of such types – either core/extrathematic, or core/periiphery, or periphery/extrathematic – are the following ones,

Area, Category, Characterization, Charges, Circumstances, Components, Configuration, Containing_Event, Cotheme, Court, Degree, Depictive, Difference, Dimension, Duration, Event, Experience, External_Cause, Feature, Final_Category, Final_Value, Firearm, Form, Goods, Group, Handle, Initial_Category, Intermediary, Internal_Cause, Judge, Legal_Basis, Lessor, Locus, Manipulator, Manner, Material, Message, Mode_Of_Transportation, Money, Occasion, Parameter, Perceiver_Passive, Place, Point_Of_Contact, Purpose_Of_Theme, Rate, Reason, Resource, Result, Resulting_Action, Seller, Sleep_State, Speaker, Standard, Support, Time, Topic

Finally Semantic Roles which may only appear in one such syntactic type are the following ones where we divide Core/C from Extrathematic/E and Periphery/P:

C
Arguer, Authority, Body_Part, Cognizer, Entity, Impactor, Interlocutor, Item, Part, Participant, Partner, Party, Performer, Phenomenon, Side, Theme
E
Beneficiary, Concessive, Duration_Of_End/Final_State, Iteration, Ground, Recipient, Subregion
P
Criteria, Value

FN is accompanied by examples which are represented as constituent structures. Constituency labels used in FN are the following:

2nd, 3rd, a, ajp, apos, avp, cni, dni, inc, ini, n, np, poss, pp, pping, ppinterrog, quo, sabs, sbrst, sfin, sforto, sing, sinterrog, srel, sto, sub, swether, unknown, vpbrst, vpfnt, vping, vpto

where in some cases we see the use of a lexical category (a, n, poss). Grammatical Functions labels are only 4,

appositive, DEP, EXT, OBJ

where EXT is associated to what is usually called the SUBject, and they may have Indefinite Null Instantiation (INI), Definite Null Instantiation (DNI) and Constructional Null Instantiation (CNI). Constructional Null Instantiation, CNI is used to mark omitted subjects in passives, imperatives, gerunds, infinitives and objects of some imperatives. DNI is used to mark arguments omitted when they are presupposed as known. INI is used with existential
cases with intransitivized objects etc. Inchoative uses of a LU is distinguished from the causative use. But passive, middle and ergative uses are not present.

In addition to that, FN has built a network of hierarchical Frame-to-Frame relations with super frames called “scenarios” – currently 23 have been created - to allow for inferencing to take place. They are listed here below:

Inherits From:, Is Inherited By:, Subframe of:, Has Subframes:, Precedes:, Is Preceded by:, Uses:, Is Used By:, Perspective on:, Is perspectivized in:, Is Causative of:, See Also:

However not all the inferential functions are filled as they should be in a real ontology. Scenarios constitute the highest or top level of the hierarchy, which may be dubbed as complex Frames and connect together a certain number of Frames. Semantic inheritance relations are encoded through the two basic relations, “Inherits From” and “Is Inherited By”. The currently encoded Scenarios are the following one,

Attempting_scenario, Causation_scenario, Change_of_phase_scenario, Change_of_state_scenario, Commerce_scenario, Crime_scenario, Employee's_scenario, Employer's_scenario, Employment_scenario, Getting_scenario, Giving_scenario, Knot_creation_scenario, LosePossession_scenario, Measure_scenario, Motion_scenario, Obligation_scenario, Receiving_scenario, Requirement_scenario, Resolve_attempt_scenario, Safety_scenario, Searching_scenario, Shooting_scenario, Transfer_scenario

There are 107 Frames directly connected to the Scenarios. Producing an handcrafted highly reliable deep lexical representation as FN is very labor-intensive and time-consuming. The number of Frames of the current release has increased from version 1.2 by a 62% - from 488 to 795 – thus indicating that the possibility to cover all facets of reality by producing frames and linking them to the evoking predicates and the example sentences is an effort which gives slow but steadily increasing results. However this is still to be regarded as a small inventory of all possible frames and the FN team itself does not expect to have a full inventory of frames until a substantial proportion of the general-purpose vocabulary of English has been analyzed.

FN constitutes the last and more perspicuous level of representation of knowledge of the world and the least generalizing one, on top of strictly domain-related pieces of reality expressed in natural language. The frames are organized into hierarchies where the more specific frames inherit some properties from the more general ones. In general, some semantic areas are covered only by a general frame, some others by a combination of specific frames and some by a combination of specific and general frames.

As said above, the current version of FN is useful for a certain number of different text processing tasks, like using Frames and Frame Elements to produce more perspicuous semantic roles associated to the output of a dependency parser. Eventually there is always the possibility to extend it manually – if needed - to other lexical units which share/evoke the same concept/frame. However this task, when based completely on FN and its set of examples, is poorly executed due to the small number of lexical units listed for each Frame (see Pado & Boleda, 2004). The same researchers remark the lack of uniformity in the argument structures associated to each Frame, which adds up to the poor predictability that
can be associated to the evoking predicates of a given frame. Other problems related to FN are due to the lack of a one-to-one mapping between evoking LUs, their syntactic realization in example sentences and semantic roles (Frame Elements). In fact, on the one side FN proposes a general scheme for frame-evoking LUs by associating Frame Elements to portions of text – actual words - in example sentences; on the other side it tries to reconcile generality with specificity by increasing the number of example sentences. This should cover the problem of lack of selectional restrictions associated to Frame Elements. Example sentences are however only ancillary or subsidiary pieces of information in the overall theoretical framework: their syntactic structure, grammatical relations and grammatical categories play no restrictive role on the way in which Frame Elements – for instance – have been encoded in the corresponding frame evoked by the governing predicate.

In this respect, one of the basic problems presented by the use of examples for the purpose of machine or automatic learning of semantic roles, is the presence of a great number of Null elements as indicated by the grammatical representation, which makes the mapping from Frame Elements to actual sentences and viceversa, poorly executable and consequently poorly predictive/predictable. Null elements are the result both of syntactic Alternations, in the sense of Levin, and syntactic transformation or derivation. Most common structures are,

- Passivized structures where the Agent has been deleted;
- Intransitivized structures where the Object has been left unexpressed and the subject is an agent;
- Middle structures where the Object has been left unexpressed and the sentence is non factual;
- Inchoativized structures are represented in the FEs;
- Ergativized structures where the Object has been raised to Subject and the deep subject is left unexpressed.

As Green et al. remark in their paper, using FrameNet for open texts processing reveals its limits as far as coverage is concerned: low recall affects both the number of semantic frames, but also the frame-evoking capacity of each list of verb units, which is very low. No verbs are listed for over 30% of all frames, while another 10% or so list only 1 or 2 verbs. This must be regarded as a secondary or side-effect of the overall theoretical framework, which takes Lexical Units and not the Predicate Argument Structure or eventually the sentence in which LUs are contained as the primary source of lexical knowledge representation.

2.1. Lexical Resources and Ontologies

To cope with the problem of sparcity, a number of other lexical resources have been proposed in the past and others are being proposed now. We start by commenting the paper by Green R. et al.(2005) where the authors comment in their Conclusions that “... sets of verbs evoking a common semantic frame can be induced from existing lexical tools.” However as the authors have to admit, the lexical resources available on English suffer from the problem of sparseness which affects both Recall and Precision in the last resort.
Many recent annotation efforts for English have focused on pieces of the larger problem of semantic annotation, rather than producing a single unified representation like the Prague Dependency Tectogrammatical Representation (Hajicova & Kucerova, 2002).

Other important resources are the following ones, which will be commented below:

- **PropBank** (Palmer et al, 2005) annotates predicate argument structure anchored by verbs.
- **NomBank** (Meyers, et al., 2004a) annotates predicate argument structure anchored by nouns.
- **TimeBank** (Pustejovsky et al, 2003) annotates the temporal features of propositions and the temporal relations between propositions.
- The **Penn Discourse Treebank** (Miltsakaki et al 2004a/b) treats discourse connectives as predicates and the sentences being joined as arguments.

### 2.1.1 PropBank

The Penn Proposition Bank focuses on the argument structure of verbs, and provides a corpus annotated with semantic roles, including participants traditionally viewed as arguments and adjuncts. An important goal was providing consistent semantic role labels across different syntactic realizations of the same verb, as in *the window* in *[ARG0 John] broke [ARG1 the window] and [ARG1 The window] broke*. Arg0 and Arg1 are used rather than the more traditional Agent and Patient to keep the annotation as theory-neutral as possible, and to facilitate mapping to richer representations. Finally, the corpus contains 5 different types of Arg. Coarse-grained sense tags, based on groupings of WordNet senses, are being added, as well as links from the argument labels in the Frames Files to FrameNet frame elements.

### 2.1.2 NomBank and NomLex

The NYU NomBank project can be considered part of the larger PropBank effort and is designed to provide argument structure for instances of about 5000 common nouns in the Penn Treebank II corpus (Meyers, et. al., 2004a). PropBank argument types and related verb Frames Files are used to provide a commonality of annotation. This has lately been upgraded by the creation of NomLex which includes over 8000 entries fully classified and subcategorized, with the release of NomBank 1.0.

### 2.1.3 TimeBank

The Brandeis TimeBank corpus focuses on the annotation of all major aspects in natural language text associated with temporal and event information (Day, et al, 2003, Pustejovsky, et al, 2004). Specifically, this involves three areas of the annotation: temporal expressions, event-denoting expressions, and the links that express either an anchoring of an event to a time or an ordering of one event relative to another. Identifying events and their temporal anchorings is a critical aspect of reasoning, and without a robust ability to identify and extract events and their temporal anchoring from a text, the real aboutness of a text can be missed. The core of TimeBank is a set of 200 news reports documents, consisting of WSJ, DUC, and ACE articles, each annotated to TimeML 1.2 specification. It is currently being extended to AQUAINT articles. The corpus is available from the timeml.org website.
2.1.4. **WordNet**

WordNet is a handcrafted lexical database that is based on the hyperonymy-hyponymy relation – but comprises also other semantic relations like antonymy, meronymy, a causal relation, etc. – to shape the structure of an English lexicon and build a lexical semantic network where each entry is organized as a synset, i.e. a list of close synonyms. Each entry is also accompanied by a gloss which however, contrary to what happens in LDOCE, uses a non-restricted vocabulary. The use of a restricted vocabulary would enable its computability in terms of word sense disambiguation. The same problem affects another important lexical resource, FrameNet, which is enriched with glosses which however don’t use a restricted vocabulary.

2.1.5. **Levin’s Verb Classes**

According to the hypothesis stated in (Levin, 1993), syntactic features of verbs are semantically determined and thus syntactic behavior of verbs can be used for their semantic classification. Levin describes syntactic behavior of verbs with respect to possible syntactic alternations and semantic classes are built from verbs that undergo a certain number of alternations. An alternation means a change in the surface realization of the argument structure of a verb. Levin’s list contains 258 semantic verb classes.

2.1.6. **The LCS database**

(Dorr, 2001) was designed as a semantic representation of predicates and propositions. It describes the semantics of verbs as a combination of semantic structure and semantic content – semantic structure is characteristic for all verbs from one semantic group whereas particular verbs can differ in their semantic content. The lexical item is an oriented rooted graph that bears information on its subject, its objects (arguments) and its 'modificators' and on their obligatoriness / optionality. In addition, their thematic roles are stated as well as restrictions on conceptual categories (also called conceptual POS, as e.g. 'thing', 'event', 'state', 'place', 'purpose', 'manner', 'time').

LCS uses the following 12 THETA-ROLES labels for Logical Arguments:

AG, EXP, TH, SRC/SRC(), GOAL/GOAL(), INFO, PERC/PERC(), PRED/PRED(), LOC/LOC(), POSS, TIME/TIME(), PROP

and the following 7 role labels for Logical Modifiers:

MOD-POSS, BEN, INSTR, PURP(), MOD-LOC, MANNER(), MOD-PROP, PARTICLE

Apart from Purpose and Manner all other roles modifiers are attested both with and without preposition. Particle indicates other particle handled via collocations. The presence of () indicates that the roles can be preceded by preposition. Together they amount to 19 different labels. However when composed together with preposition and obligatoriness or optionality marked by comma or underscore, we reach the number of 250 different Thematic Grids, which have been mapped to the theta roles of PropBank.
2.1.6.1. Relation of LCS to PropBank

The 250 thematic grids of LCS have been mapped to the theta roles of PropBank which contain the following role labels:

- **ARG0**
- **ARG1/ARG1()/ARG1-PRD(THAT/TO)/ARG1-REC**
- **ARG2/ARG2()**
- **ARG3/ARG3()**
- **ARG4()**
- **ARGM-LOC()**
- **ARGM-TMP**

<table>
<thead>
<tr>
<th>Role Description</th>
<th>Arguments</th>
</tr>
</thead>
<tbody>
<tr>
<td>_ag.goal(to)_info(that,about)</td>
<td>arg0 arg2(to) arg1-PRD(that/about)</td>
</tr>
<tr>
<td>_exp.goal.src(from)</td>
<td>arg0 arg1-REC arg2(from)</td>
</tr>
<tr>
<td>_exp,mod-poss(with),mod-perc(about)</td>
<td>arg1-REC arg2(with) arg1(about)</td>
</tr>
<tr>
<td>_exp_perc_mod-prop(to)</td>
<td>arg0 arg2 arg1-PRD(to)</td>
</tr>
<tr>
<td>_exp_prop(that)</td>
<td>arg0 arg1-PRD(that)</td>
</tr>
<tr>
<td>_th_time()</td>
<td>arg1 argm-TMP</td>
</tr>
</tbody>
</table>

2.2. Lexicalized Causal Meaning: LCS and FrameNet

As noted above, one of the problems that faces FN lexical representations is the lack of a direct mapping into Levin’s classes. Syntactic lexical alternations are very useful to detect Frame Elements which are not primitives but are related to passivized, intransitivized, or inchoativized structures derived from an underlying basic lexical entry. Causal meaning is however encoded in FN. Clear cases of this situation include causative-inchoative pairs which is encoded in the hierarchical link “Is Causative of”. So although causative-inchoative pairs are not in the same frame, the FrameNet database provides an explicit link between the paired frames via a frame-to-frame relation Causative of.

Generally speaking, causes need causative verbs to be expressed or else discourse markers. From a conceptual point of view, all predicates belonging to the following aspectual classes may contain an abstract CAUSE operator:

- accomplishments
- non reversible accomplishments
- gradual accomplishments
- achievements
- punctual achievements

Predicates belonging to these classes may also be regarded as containing two additional features:

- intentionality
- animacy
However, it may be proven that Causative Relations may be non-intentional and have a non-animate causer as it would happen with all natural events. FN does not provide aspectual information for single LUs but only what can be referred to Frames as a whole, i.e. relations intervening between the arguments (optional and obligatory ones) of the governing predicates. In this sense, Causality in FE is only a property of the overall Frame. However, from a computational point of view the lexical semantic notion of causality is very important to help define Causality Relations in sentences and texts – but see the chapter dedicated to that specific issue in Book 2.

In this perspective, we follow Ray Jackendoff by postulating the existence of a certain number of primitive lexical operators among which is the CAUSE operator. For instance, the sentence, 

- John built a house

may be conceptually decomposed and represented into the following representation,

```plaintext
CAUSE(John-SUBJ,
    BE-IN-THE-WORLD/EXISTENCE(house-OBJ),
    Evs_{i,j}(Ev_i … Ev_j))
```

Which can be paraphrased as follows,

John caused the house to be in existence/world in the time span intervening between Ev\_i and Ev\_j, where the final event represent the Accomplished Result State. More on these topics below.

Predicates belonging to STATE or ACTIVITY cannot be used to express Causes and consequently cannot be decomposed into the lexicalized conceptual causal operator. On the contrary they may be regarded the preferred target of RESULT clauses. In the LCS lexicon, on the total number of lexical items present, which amount to 9000 entries, over 5000 have a CAUSE operator incorporated in their lexical meaning representation.

FrameNet uses the following FRAMEs related to Causation to classify verb predicates:

- **Causation**, which is used for Verbs, Nouns and one Adjective, ‘causative’;
- then another 26 Frames which are used to define Verbs and Nouns:

  ```plaintext
  Cause\_begin\_motion,  Cause\_change,  Cause\_change\_of\_consistency,
  Cause\_change\_of\_phase,  Cause\_change\_of\_position\_on\_a\_scale,  Cause\_confinement,
  Cause\_expansion,  Cause\_fluidic\_motion,  Cause\_harm,  Cause\_impact,  Cause\_motion,
  Cause\_of\_shine,  Cause\_temperature\_change,  Cause\_to\_amalgamate,  Cause\_to\_be\_dry,
  Cause\_to\_be\_sharp,  Cause\_to\_be\_wet,  Cause\_to\_continue,  Cause\_to\_end,
  Cause\_to\_experience,  Cause\_to\_fragment,  Cause\_to\_make\_noise,
  Cause\_to\_make\_progress,  Cause\_to\_move\_in\_place,  Cause\_to\_resume,  Cause\_to\_start,
  Cause\_to\_wake
  ```

All predicates thus described – evoked by each frame - will have a Causer or an Agent as Subject. The number of predicates thus classified in FrameNet is however very small, less than 548 over 10195 predicates, 349 of which are constituted by verb predicates.
LCS entries contain cross reference to Levin verb classes, to WordNet sense, to PropBank argument list which have been mapped to a more explicit label set of Semantic Roles which can be regarded more linguistically motivated than the ones contained in FrameNet which are more pragmatically motivated. However, for our purposes, LCS notation is more perspicuous because of the presence of the CAUSE operator, and is more general: on a total number of 9810 lexical entries – and 4868 different linguistic forms -, 5000 contain the CAUSE operator. On the contrary, in FrameNet on a total number of 10000 lexical entries, only 333 are related to a CAUSE Frame; if we search for the word “cause” in the definitions contained in all the Frames the number increases to 789 but is still too small compared to LCS. Here below is one example of LCS entries:

`:DEF_WORD "prevent"
:CLASS "059"
:WN_SENSE ("1.5" 01387332
 ("1.6" 01669882))
:PROPBank ("arg0 arg1 arg2(from)"
 :THETA_ROLES ((1 "_ag_th_prop(from)"))
 :LCS (cause (* thing 1)
 (go circ (* thing 2)
 (from 3) circ (thing 2) (at circ (thing 2) (* nil 27)))
 (prevent+ingly 26))
 :VAR_SPEC ((27 (aspect prog))) )

`:DEF_WORD "purge"
:CLASS "10.4.1.a"
:SOURCES (LEVIN)
:WN_SENSE ("1.5" 00509207 00045781
 ("1.6" 00612875_purge%2:32:00:: 00051406_purge%2:29:00::)
 ("1.7.1" 00716629_purge%2:32:00:: 00057833_purge%2:29:00::)
 ("2.0" 00875519_purge%2:32:00:: 00071623_purge%2:29:00::))
:PROPBank ("arg0 arg1 arg2(from - up.)"
 :THETA_ROLES ((1 "_ag_th_src()"))
 :LCS (cause (* thing 1)
 (go loc (* thing 2)
 (from 3) loc (thing 2) (onloc loc (thing 2) (thing 4)))
 (purge+ingly 26))
 :VAR_SPEC ((1 (animate +)))
)

To recover the same information from FrameNet, either the definition or the Frame Elements had to be inspected. In particular, Prevent is part of the Frame PREVENT which has Preventing Cause as Frame Element. “Purge” is part of the Frame Removing and in order to know that it is a causative verb one needs to read the definition which paraphrases the content of the Frame Elements by saying that The Agent is a person (or other force) that causes the Theme to move.
3. FROM FRAMENET TO A LINGUISTICALLY-BASED FULLY SUBCATEGORIZED LEXICON

As said above, we intend to test hypotheses on the nature of lexical knowledge and lexicon acquisition. We assume that the representation of lexical knowledge in a system for text understanding is equivalent to that present in the mind of an individual, i.e. it is psycholinguistically relatable to the one possessed by a speaker of the language. As said above, differently from what is assumed by FN theory, where Frames are evoked by LUs, we believe that Frame or frame-like representations and more generally any lexical representation at all, can only be derived from sentence or propositional level interpretation. It is thus a type of knowledge which is not merely word-related but profoundly structurally-related both syntactically and semantically.

Another important issue is that it is impossible to list all lexical entries for a given predicate, due to the non predictability of all contextual determined variations that can impinge on meaning and on structure.

There are two basic components or subtheories onto which our lexical representations are built: one is the syntactic-semantic component, the other is the semantic-conceptual component. We will address them separately in the subsections below.

3.1. Linguistic Theories and Computational Issues

The questions we pose ourselves in implementing the programs needed to simulate lexical knowledge and acquisition are approximately the following ones: what is the set of primitives required to build lexical descriptions which we call forms - in accordance with LFG - which make available to native speakers both syntactic and semantic information required to parse and recognize sentence of their mother tongue in the acquisition phase. Also, since we believe that low level inferences are made unconsciously by speakers, what information are required in order to automatically build a conceptual representation to be used by the inferencing mechanism?

In this sense, lexical forms are tightly connected to the nature of the human and computational parser and the inference mechanism which uses them. However, we believe the format of lexical forms should be universal and internally motivated, rather than externally conditioned.

The initial hypotheses verify whether Grammatical Functions rather than Semantic Roles can be taken as primitives, or if the opposite applies (see Wilkins, 1988). The two hypotheses are derived from Chomsky's and Bresnan's theories: both take for granted the existence of syntactic categories and an X-bar system to build main constituents: however, in Chomsky's system thematic roles are mapped directly on to s-structures which are specified in terms of syntactic constituents, whereas in Bresnan's model thematic (or rather semantic) roles are only subsidiary on f-structures which act as interface onto c(onstituent)-structures.

Consider at first Chomsky's projection principle:
"Representations at each syntactic level (LF, and D- and S-structure) are projected from the lexicon, in that they observe the subcategorisation properties of lexical items." (1981: 29, see also Chomsky 1986).

According to both theories, words, when they are first heard by the child, are categorized into lexical categories and concatenated into major constituents to form the first structural representation, onto which lexical forms are projected in order to proceed to the semantic interpretation. Whereas Bresnan introduces grammatical functions as a restriction on c-structure wellformedness and as an interface to lexical projections made up both of grammatical functions and semantic roles, Chomsky uses an enriched notion of c-structure, i.e. S-structure. This is meant to capture the same generalizations of Bresnan's two levels: annotated c-structure and f-structure, through the principle-and-parameter approach. In addition, Chomsky introduces the level of D-structure, which is "...directly associated with the lexicon. It is a 'pure' representation of theta-structure, expressing theta relations through the medium of the X-bar-theoretic conditions in accordance with the projection principle" (1988:2).

Furthermore, a lexical entry must specify "enough of its properties to determine its sound, meaning, and syntactic roles... it should not contain redundant information, for example, about the quality of the vowel, properties of action verbs generally, or the fact that together with its complement it forms a VP." (ibid.,2). In Chomsky's terms then, the lexicon is separated from D-structure, and both are related to S-structure by means of iterated application of Affect(Move)-Alpha. This segmented view or modular view of the grammar is completely opposite to the unitary view proposed in Bresnan, in which the lexicon is the repository of all information relating c-structure to its interpretation. The lexical form of an item includes its phonetic, syntactic and semantic properties all in one single format, the only exception being constituted by derived forms (the Affect-Alpha application) which are built once and for all by Lexical Redundancy Rules and as such are listed in the lexicon in the same form as the underived or original one, thus receiving the same interpretation in terms of predicate argument structure and semantic roles assignment.

In Movement theories then, in order to address a level of representation like D-structure or LF-structure one must go through the interaction with S-structure, or phrase structure representation; on the contrary, in LFG, c-structure is separated from f-structure and both are separated from lexical structure and may be addressed separately. Also, Thematic Structure is separated from syntactic functions and both are separated from discourse functions. Syntactic functions are associated with semantic roles and discourse functions according to principles of Lexical Mapping (see Bresnan & Kanerva, 1989:23-28). Subject and object correspond virtually to any semantic role and may even be nonthematic; oblique arguments, on the contrary, have fixed semantic roles, as Bresnan & Kanerva note (ibid.,25), implying by this that there must be some additional information which is contextual in kind: in particular, this may be marked by the preposition used, obviously when no ambiguity may arise - which as we shall see in more detail below is far from unfrequent. The only fixed fact is the possibility to classify certain semantic roles relatively to the coupling with syntactic function: for instance, Agent may never be [+objective], like Locative. However, the main difference between the two theories lies in the fact that in Chomsky's theory theta-roles are syntactically relevant elements; on the contrary, in Bresnan's framework, semantic roles are just semantic objects related to Predicate Argument Structure, or simply Argument Structure.
We shall follow the second position. In our system, Semantic Roles are partly derived from a mapping with Grammatical Function, as in Bresnan's system, and partly reflect the wealth of information contained in semantic and conceptual classes which are associated to each verbal predicate. These classes contain information about the meaning of the verb, meaning which can be represented partly in terms of Semantic Roles and partly in terms of Conceptual Representations, as we shall see in more detail below. Even though we think it sensible to assume that it is impossible to transpose all world or encyclopaedic knowledge into classes or roles, we believe it a scientifically interesting hypothesis trying to define these levels of representation within a global system of text understanding in order to see what kind of information must be made available to the semantic interpreter. In this sense, we think FN can be deemed the most important attempt in that direction.

4. TRANSITIVITY AND THE LIFVE

Theoretically speaking, we can assume that transitivity is the main syntactic class we want to characterize in our lexicon, the remaining classes being only derivative on it. In fact, from our work on the main general lexicon of Italian, we found that transitive verbs constitute the great majority of verbs, and their number is by far superior to intransitive verbs. Over a number of 13,000 verb entries we see that intransitive classes are only approximately 3,000, the remaining belonging to one of the transitive classes, as follows:

\[ Tr = \text{transitive} [6700]; \ tr\_cop = \text{transitive+predicative argument} [112]; \ tr\_perc = \text{transitive_perceptive} [24]; \ ditr(+preps) = \text{ditransitive} [386]; \ psych1 = \text{psychic 1} [59]; \ psych2 = \text{psychic 2} [251]; \ psych3 = \text{psychic 3} [19]; \ inac = \text{unaccusative} [935]; \ inerg = \text{unergative} [1612]; \ rifl = \text{reflexive} [890]; \ rifl\_rec = \text{reflexive reciprocal} [203]; \ rifl\_in = \text{reflexive inherent} [304]; \ erg\_rifl = \text{ergative reflexive} [1742]; \ imp = \text{impersonal} [30]; \ imp\_atm = \text{impersonal atmospheric} [32]; \ cop = \text{copulative} [8]; \ mod = \text{modal} [5]; \ C\_mov = \text{motion verb + another class} [255]; \ C\_prop = \text{propositional verb + another class} [210]; \]

Most of the classes have labels which are self-explanatory. Some of the classes have specific values like the one of psychic verbs, which we divided into three subclasses both for syntactic and semantic reasons (but see Delmonte, 1989):

- psych1 verbs assign the Experiencer role to the subject;
- psych2 verbs assign the Experiencer role to the direct object;
- psych3 verbs are intransitive and assign the Experiencer role to an oblique argument;

Another class that needs clarification is the one of, erg(ative)-refl(exive) - reflexive verbs which have as subject, the object of the transitive or psychic corresponding verb.

Latey, we did additional lexical research on the properties of specific lexical classes: in particular, we focussed our attention on verbs governing sentential complements, including both tensed and untensed propositions. The final classification ended up with over 100
different syntactic-semantic classes. Differences regarded mainly the following linguistic items:

a. optionality of the complementizer;
b. type of the complementizer;
c. obligatoriness of subjunctive mood;
d. obligatoriness of negation;
e. propositional type;
f. optionality of the sentential complement;
g. idiomatic lexical forms.

From basic syntactic classes, the 19 listed above, we posited the need to introduce more fine-grained labels where we specialized the following ones:

h. ditr(ansitive)1 - <SUBJ, OBJ, COMP>
i. ditr(ansitive)2 - {<SUBJ, OBJ2, COMP>; <SUBJ, OBL, COMP>}

We then added a specific class for negative transitive verbs:

l. ntr_propint(negative transitive + question + subjunctive mood)

Examples of idiomatic lexical forms are the following ones:

o. avere, aux_prop_a-[ebbi, ebbe, ebbero] / have
p. dare, ditr2_prop-[a_bere] / give
q. far_si, tr_prop / do as if
r. valere_la_pena, inac_propx / be worth the trouble

where prop stands for proposition and propx for infinitival lacking a complementizer.

Going back to the general remarks on transitivity, if we look at nontransitive predicates, we might regard them as conflated forms of either transitive or copulative corresponding predicates. In line with what Hale and Keyser (1991) discuss in their paper, we could also assume that a lot of verbs are conflated forms in which a light verb (fare, stare, avere, mettere etc.) and either an adjective or a noun are made into a verb. We can have verbs like “figliare” (avere un figlio / have a baby) for animals; verbs which allow cognates object as in “vivere/live, correre/run” from “fare una vita” or still “vivere una vita”/live a life with the repetition of the internal object; or verbs like “pranzare/lunch, cenare/dinner, etc.”, or “pescare/fish, bagnarsi/bathe, tuffarsi/dive, etc.” presumably made up of a light verb “fare”/make and the internal noun denoting the kind of activity or event; weather verbs “piovere/rain, nevicare/snow, lampeggiare/flash, tuonare/thunder, etc.”

Besides, we know that derivational rules build verbs from adjectives, as discussed for Italian in Scalise (1984) with parasynthetic derivation - “annerire/blacken, arrossire/redden, appiattire/flatten, etc.” with the obvious meaning “cause to become ADJ” where ADJ is the underlying adjective base form. The same thing could be said for an extended number of inherent reflexive verbs like “arrabbiarsi/enrage, ammalarsi/fall ill, indurirsi/harden, etc.” which could be treated as a form of a copulative verb, using “diventare”/become as a light verb and the corresponding adjective as a predication of the subject.
Finally, we have psychic verbs which are decomposable into a light verb and a noun, as in “addolorare/distress, interessare/interest, divertire/amuse, meravigliare/amaze, stupire/astonish, piacere/like etc”; this can be explained by the fact that in the argument structure of the underlying noun there is always the presence of a beneficiary, or an emotional causer and an experiencer. Thus we can say:

i. il dolore/l’interesse/l’ostupore/il piacere di Gianni per Maria/*di Maria
the pain/the interest/the astonishment/the pleasure of John for Mary/ *of Mary

which is equitable to,

ii. Maria addolora/interessa/stupisce/piace a/ Gianni
Mary distresses/interests/amazes/likes/ John

4.1. The Computational Lexicon of Italian

Work on a Lexicon of Italian – the LIFVE started in the ‘80s thanks to a contract with DIGITAL Eq. to produce a Text-to-Speech (TTS) system for unrestricted text. At first we actually produced a fully subcategorized lexicon of Italian with reference to most frequent 5000 lemmata as registered in current Frequency Lexicons of Italian (FLI). Followed by the first Treebank worldwide on the basis of 40,000 word corpus, BUT fully manual. Subsequently, from the beginning of the ’90s we worked on a more extended version of the LIFVE. We classified a list of 18,000 Italian verbs entries (approximately 10,000 different lemmata, and 5,000 different roots) for computational purposes, producing a highly fine-grained categorization of syntactic and semantic features of arguments of the predicate. At a national level, an initiative called “The Ilex project” was started by three research centres ITC-IRST, University of Tourin, University of Venice and the goals were formulated in a very ambitious way, by setting ourselves as goals the following incremental sizes of lemmata: 1000 lemmata end of 1996; 10000 lemmata end of 1997; 30000 lemmata end of 1998.

Lexical Information for ILEX was constituted by POS and syntactic-semantic class information; morphological and phonological information; semantic information. However, work stopped after the first year, 1995. Each research unit, we too, continued working on the peculiarities of Italian verb classes.

4.1.2. LIFVE - a Lexical Transducer

With LIFVE we wanted to produce a lexical encoder, a fully automatic annotation tool, which allowed the linguist annotator to easily encode all syntactic and semantic aspects of lexical knowledge. In particular, we organized the algorithm into a sequence of operations with the following properties:

- Hierachy of syntactic-semantic choices
- Highly efficient and rigorous encoding
- All information in one single record
- Linguistic rules for the multilevel transducer
Semantic and Conceptual representation are not be explicitly represented and are derived automatically by rule from lexical and syntactic information with the addition of semantic and aspectual classes. In our lexicon, then, both semantic roles and conceptual representations are derivative notions.

Suppose now that we want to characterize a computational lexicon: this should be made in reference to what a parser needs when analysing a text. For instance, in order to produce predictions as to what the internal constituents of a given predicate might be - verb guidance as is commonly called, and consequently the structure of the VP governed by a given verb, we want to have access to the verb predicate argument structures which might correspond to the following information:

1. predicate; 2. syntactic class; 3. aspectual class; 4. semantic class; 5. the list of arguments contains for each argument:
   a. syntactic constituency; b. grammatical function; c. semantic role; d. selectional restrictions or a control equation for open functions; e. no information at all for propositional arguments.

The final result for the verbs "give" and "give_up" is the following record of information:

i. give, ditrans, achiev, transf_poss, [np/subj/owner/[human], np/obj/poss/[object], pp/obj2/recip/[human]].
ii. give_up, trans, achiev, abort, [np/subj/agnt/[human], np/obj/theme/[activ, object]].

As discussed above, and also proposed by FN, a number of derived structures can reasonably be expected to descend from such a lexical representation of verbs belonging to the class of trans(itive) or ditrans(itive). However, none of these structure will be represented explicitly in our lexicon seen that they can all be derived automatically from lexical rules, as indicated in LFG theoretical framework. Among these derivative structures we note the following:

- passive constructions for transitive verbs;
  i. The present was given to Mary (general passive structure)
  iv. Mary was given a present (special passive rule for English)
- dative shift;
  v. John gave Mary a present (specialized for Germanic languages)
- generation of a quantified oblique argument in case of agentless passive;
  vi. The book was sold to help the students
- ergative or inchoative constructions again from transitive verbs;
  vii. The book sold (*to help the students
- object intransitivized constructions from transitive verbs that allow it and are therefore marked as such in terms of object argument NP optionality;
  viii. John is reading
- direct object NP deletion or indirect object2 PP deletion from psychic verbs, adequately marked in the lexicon, inducing thus the interpretation of generic assertions;
  ix. La musica rende _ felici / Music makes (everybody) happy
- postverbal subjects for focusing purposes with unaccusatives, freely in Italian;
  x. E’ arrivato Gino / *has arrived John
postverbal subjects for topicalizing purposes, (almost) freely in Italian and English – only with nouns and emphatic pronouns as SUBJECTs;

xi. ... disse Gino / ... said John

preverbal sentential argument dislocation with impersonal predicates;

xii. That John is happy results from what I heard yesterday

controlling secondary predication with transitive verbs is expressed in the lexicon in Italian where it is limited to a restricted class of predicates - “dipingere”/paint for resultatives, “considerare/consider, ritenere/maintain” with individual level predication and “rendere”/render for stage level predicates; in English it is constrained by aspectual class of the predicate - only change of state verbs allow freely resultatives, i.e. accomplishments and achievements (see Rapoport);

xiii. John hammered the metal flat (*happy, *tall)

John painted his house red (*high)

middle constructions are not freely generable from the lexicon. In particular, aspectual classes are relevant - stative verbs cannot form middles; only certain adverbs are allowed; only affected themes may be used, however (see Hoekstra & Roberts); the same applies to impersonal constructions;

xiv. *This lesson knows easily

Bureaucrats bribe easily (*evidently)

This truck loads easily

4.1.3. Lexical Types

Alongside the specification of optionality for certain arguments - the feature “lexs” - our computational lexicon lists a number of interesting transitive predicates. Here below we list some examples with special argument specifications for double object constructions, adverbial objects, and secondary predication: all these cases require the parser to search in the argument list for a given functional or semantic type. In the case of adverbial object NP no passive is allowed, nor can it constitute a legal argument for wh- movement; for secondary predication we shall have to build an open function which predicates on the object NP, as happens in verbs like “considerare”/consider.

We will start from verbs. Here below, we list one representative for each verb class characterized by syntactic criteria in second slot:

4.1.3.1. Verbs Syntactic Types

<table>
<thead>
<tr>
<th>Verb</th>
<th>Arguments</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>abbagliare, tr, achievite, propr</td>
<td>[np/subj1/actor/+abst, np/obj1/theme_aff/+hum, lexs]</td>
<td></td>
</tr>
<tr>
<td>andare, unacc, accompling, exten</td>
<td>[np/subj1/theme_affect/+ani, -ani, +hum, obj2/goal/a/-ani, -abst]</td>
<td></td>
</tr>
<tr>
<td>abbaiaire, unerg, achieviter, inform</td>
<td>[np/subj1/actor/+ani, obl/address/a/[]]</td>
<td></td>
</tr>
<tr>
<td>abbassare, refl, achiev, measu_min</td>
<td>[np/subj1/agent/[rifl]/[+hum]]</td>
<td></td>
</tr>
<tr>
<td>apparire, cop, statv, hyper</td>
<td>[np/subj1/theme_bound/+ani, -ani, +hum, -hum, +abst, ap/acomp/prop/[subj = subj1]]</td>
<td></td>
</tr>
<tr>
<td>accadere, vimp, statv, factv, [s/subj/prop/[subj=subjx], obj2/goal/a/[+hum, +ani, lexs]]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>addolorare, psych1, statv, eval, [np/subj1/causer_emot/+hum, -ani, +abst], np/obj1/theme_unaffect/[+ani, +hum]]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dispiacere, psych2, achieviter, eval, [np/subj1/theme_emot/[+ani], obj2/exper/a/[]]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.1.3.2. Verbs specific syntactic types

persuadere, tr, activ, manip, [np/subj1/actor/+hum, np/obj1/goal/+hum, vcomp/prop/a/\[subj=obj1\]]
pigliare, tr, pos, [np/subj1/actor/+hum, np/obj1/+hum, pcomp/prop/per/\[subj=obj1\]]
pesare, tr, stat, measu_maj, [np/subj1/causer/-ani,+hum], np/spec/nadverb/[+abst,lexs]
passare, unerg, activ, posit, [np/subj1/actor/+ani,+hum], np/spec/nadverb/[+abst], obl/locat/in/[+abst]
trattare, tr, activ, process, [np/subj1/actor/+hum, advp/manner/[, np/obj1/theme_aff/[+ani,-ani,+hum]]

Then we list Lexical Types for Adjectives:

4.1.3.3. Adjective syntactic types

abile, restr, statv, intens, [np/head/[+hum], obl/theme/in/[+ani]].
certo, attr, statv, quant, [np/head/[all]].
acido, pred, statv, state, [np/subj1/head/-ani,+hum]].
brutto, tough, statv, intens, [np/subj1/head/[all]], fcomp (f’ topic (topic=subj1 (sv’ vcomp
goal/obj =
subj1 [subj=obl] )) obl/benef/per/[+ani,+hum,lexs]].
ambizio, tr, eventintens, [np/subj1/head/[+hum], obl/theme/di/[+ani,+hum,+abst]].
ammirevole, fact, unacc, intens, [subj[subj=subjx], s'/scomp/quest].

4.1.3.4. Adjective semantic types

coperto, restr, statv, state, [np/head/-ani,+hum], obl/theme/di/[+ani]]
capace, restr, statv, meas, [np/head/-ani]]
confessionale, restr, statv, orig, [np/head/-ani,+abst]]
conforme, restr, statv, quant, [np/head/-ani,+abst], obj2/goal/a/[+ani,+abst]]
contemporaneo, restr, statv, temp, [np/head/[+ani,+hum,+abst]]
vio, restr, statv, eval, [np/head/[+hum]]
buio, pred, statv, col, [np/subj1/head/[+ani,+hum,-abst]]
coraggi, restr, event, intens, [np/head/[+ani,-ani,+hum,+abst]]
falso, restr, statv, hypint, [np/head/[+ani,+abst]]
carico, tr, statv, proc, [np/subj1/head/[+ani,-ani,+hum], obl/theme/di/[+ani,-ani,+hum,+abst]]
esclusivo, restr, statv, exort, [np/head/[+ani], obl/theme/di/[+ani,+hum]]

As for nouns, we only look at syntactic classes, where we aim at specifying which optional argument must be computed as subject, object, adjunct etc.:
4.1.3.5. **Noun syntactic types**

abitazione,nl,com,[locat,locat_exten],[np/subj1/specfn/[+hum],obl/theme/in/[–abst]]
ricordo,tr,com,[event,abst],[np/subj1/+hum],np/obj1/[–ani, +abst]]
abbondanza,unacc,com,[measure,relaz],[np/subj1/theme_unaffect[–ani, +abst, +num],obl/theme/in/[–ani, +hum, –abst]]
abuso,iner,com,[event,relaz],[obl/theme/di/[–ani, +abst],np/subj1/agent/[–ani, +hum]]
abitudine,refl,com,[state,relaz],[np/subj1/theme_affect[rifl]/+[hum],obj2/goal/a/[–ani, +abst]]
abitato,cop,com,[locat,locat_exten],[np/subj1/[appost]/[–ani]]
distrazione,refl_in,com,[event],[]

4.1.4. **Lexicon and Semantics**

As said above, the relations which we assume as relevant from the point of view of semantic interpretation and which can reasonably be encoded in a computational lexicon are specified as follows:

- the interpretation of the structure of the event denoted by the sentence in which the predicate is analysed as a whole, depends compositionally on the basic lexical aspectual class, tense specifications, definiteness and quantification of the object NP, the presence of temporal adjuncts.
- relations intervening between aspectual class and the presence of a certain tense morpheme;
- relations intervening between aspectual class and the presence of temporal adjuncts;
- relations intervening between aspectual class and object NP quantificational properties;

4.1.4.1. **Verb Aspectual and Semantic Types**

aspirare,unerg,activ,eval,[np/subj1/actor/[+hum],obj2/goal/a/[+abst]]
avvolgere,tr,achiev,touch,[np/subj1/actor/[+hum, -hum],np/obj1/[–ani, +hum],obl/theme/in/[{}]]
assordire,unacc,statv,propr,[np/subj1/theme_unaffect/[+ani, +hum],obl/source/da/{}]]
affettare,tr,accompingr,divid,[np/subj1/actor/[+hum],np/obj1/theme_unaff[–ani, +abst]]
accarezzare,refl,achieviter,touch,[np/subj1/actor[rifl]/actor/[+hum],np/obj1/theme_unaff[–ani, +hum, lexs]]
accingere,refl_in,achiev,process,[np/subj1/theme_affect[rifl]/+[hum],vcomp/prop/a/[subj = subj1]]
arrivare,unacc,result_st,posit,[np/subj1/theme_affect/[+hum],obl/locat/a/{}]]
ammazzare,tr,activ,exten_neg,[np/subj1/actor/[+hum],np/obj1/theme_effect/[+ani, +hum]
mutare,tr,accomp,propr,[np/subj1/causer/[–ani, +hum],np/obj1/theme_aff[–ani, +hum]]
abortire,unerg,achiev,process,[np/subj1/causer/[+ani, –ani, +hum]]
Semantic roles are derived from argument structure and semantic field information as well as from aspectual class; the following is a complete list of roles derivable in our lexical representations:

Argument semantic roles associated to NP and to non NP
- Agent, Head (only for adjectives), Theme_affect, Appost, Poss, Theme_unaffect, Theme_emot, Causer_emot, Agent, Exper, Theme_bound (subj of predication), Trans_obj, Istigat, Address, Source_info, Goal. Specfn (only for nouns), Receiv, Owner, Perciv, Possesn, Locat, Causer (non-human), Informtn, Patnt, Nattrib, Recipnt, Ex_owner, Comit, Instr, Source, Med_exch, Ratio, Benef, Temp, Direc, Malef, Mattr, Locat/manner, Locat_exten, Asp_progr, Manner, Adjunct, Prop, Propint, Propf, Quest, Excl.

4.2. The Classification Algorithm

As described in detail below, we start with a description of lexical entries which takes into account their categorial status and reflects their syntactic behaviour in terms of major constituents, according to the predictions made by an X-bar system (see Jackendoff, 1977). The result is a set of fully specified Subcategorization Frames - in the sense that the categorial status of the subject or external argument is also included - with selectional restrictions expressed in terms of inherent semantic features.

However this is not sufficient to account for Functional Structures, which in our model are derivative on syntactic constituency; also, since functional control had to be specified at a lexical level (we chose to include this explicitly for each controlled argument rather than deriving it from a default rule and specifying control for exceptions, because this introduced a lack of uniformity in lexical formats) - this is specified by an additional feature on the controlled argument.

As we said, in LFG grammatical functions are specified in lexical forms to produce new lexical entries by means of Lexical Redundancy Rules. Some such rules, for instance Passivization, are activated by the presence of two arguments, with function SUBJECT the first and OBJECT the second. Given a parser which implements general conditions of functional completeness and functional coherence, each lexical form for a verb thus defines a set of grammatical contexts in which the verb can be lexically inserted.

Finally, since syntactic constituency does not make available information on the semantic and logic nature of arguments of predicates, we specify syntactic classes like Unaccusative, Unergative, Copulative and so on, to tell predicative from non predicative arguments apart.

What is more important, the level of predicate argument structure containing information relative to Theta-roles - which is derivative on lexical forms containing grammatical functions - requires information as to the nature of semantic classes, like Aspectual classes. Theta-roles assignment precedes the creation of Conceptual Representations used to work out inferences.

In sum, there is a set of computer programs building on a primitive description of lexical items based on categorial status, main constituents subcategorized, inherent features, syntactic and semantic classes. It produces a number of representations divided up into four levels: a syntactic level, or Level 1, a functional level Level 2, a thematic level Level 3 and a conceptual level Level 4. Each level builds on top of the lower one using restrictions based on
the information present at that level and class information available. For instance, syntactic classes are only available at Level 2 in order to assign grammatical functions; semantic classes are only available at Level 3 and 4 in order to build semantic roles and conceptual representations. As to conceptual representations, they are an expansion of semantic roles, which are abstractions and reductions on the finer grained semantic structures of predicates, and can likewise be produced with some additional information constituted by semantic classes.

The classification regards an enlarged version of the frequency lexicon of written Italian which we call LIFVE, consisting in approximately 2000 verbs, 4000 nouns and 1000 adjectives which however amount finally to 12,000 entries when different contexts of use are taken into account. In addition, we have a list of about 500 function words (see Delmonte, 1989). The output of LIFVE is used by our system for text understanding GETA_RUN and for the generation of conceptual representations which will be commented further on in this paper (but see also Delmonte, 1995).

The basic classification (see also Carrier et al. 1993) is intended to provide information to be used by more than one level of representation. Level 1 is a complete syntactic representation with a rich number of features partly used by Level 2, partly by Level 3, partly by lexical redundancy rules, and partly by Level 4. Some of these features are associated with the predicate and some with its arguments: the former are features such as [+RAIS], [+INAC], [+PERC], [+FACT]; the latter are inherent semantic features and include the following, [+ANIM, -ANIM, +HUM, -HUM, +ABST, -ABST]. Only three features can have agentive and causative meaning, [+ANIM, +HUM, -HUM], where +Anim, stands for animal kind, +Hum for human kind, and -Hum for natural powers. These three features individuate then possible causers, the first two intentional, and the latter non-intentional. Level 1 lexical classification is obtained by an interactive program written in C language, which presents a series of windows with multiple choice menus. The classifier - a linguist - is thus required to provide what we define as full subcategorization frames which basically expand the classical notion of syntactic subcategorization and extend it to include explicit reference to the subject argument, a distinction between argumental XP arguments and predicative XP arguments, plus a number of syntactic and semantic features which are meant to allow the translation of each level of representation. They are also used by the parser to impose restrictions on the class of semantically compatible and appropriate modifiers and adjuncts.

Level 2 is an annotation in lexical-functional grammar (LFG) terms of complete grammatical functions onto major syntactic constituents: this level is coupled with a number of lexical redundancy rules, which are expressed as rules in the parser, in order to build derived structures such as passivized and pronominalized structures, inchoativized and intransitivized structures and so on.

Level 3 is a translation into the semantic roles augmented by a system of aspical features. Aspectual classes are accounted by features such as STATES, ACTIVITIES, ACHIEVEMENTS, ACCOMPLISHMENTS, and others. They are derived from the more primitive features: ±imperative, ±temporal dependent, ±ingressive, and the interaction with syntactic classes. In turn, both syntactic classes and aspectual classes contribute to the labelling of arguments in terms of Theta-roles as appearing in Level 3 representations. In terms of thematic relations, the subject of activity verbs cannot be a Theme and can be either an Actor or an Experiencer. In the class of activity predicates, achievements can be distinguished from accomplishment in that the latter class requires an agentive subject
whereas the former does not. It is part of the nature of accomplishments that they involve a result or an end, and these can be expressed by the direct object or by another argument strictly subcategorized by the verb. Thus, an activity verb like "run" becomes an accomplishment if there is a specific distance to overcome or a specific goal to attain. This implies that verb aspectual classes are closely connected with the semantics of the arguments of a predicate. As to syntactic classes, PSYCHic verbs receive a labelling of their arguments in terms of Theta-roles that assigns THEME to the Subject argument and EXPERIENCER to the Object one. Level 4 is a classification in conceptual representation derived strictly from Level 3 plus information on semantic categories. Level 1 lexical classification is listed below where we show the basic items included in our system of representation:

4.2.1. Verb Classes

1.1. verbal class:
   1. transitive, 2. unaccusative, 3. unergative, 4. reflexive, 5. copulative, 6. inherent reflexive, 7. impersonal, 8. psych1, 9. psych2, 0. raising;
1.2. aspectual features:
   0. ± imperative, 1. imperative, 2. non-imperative, 3. accomp-ingr, 4. abortive, 5. inchoative, 6. completive, 7. continuative, 8. achievement-irreversible, 9. achievement-iterative;
1.3. conceptual-semantic classes:
   0. extensional, 1. subjective/intensional, 2. hyperintensional, 3. manipulative, 4. factive, 5. emotional/affective, 6. properties, 7. mental activities, 8. measure, a. perceptive-pol=0, b. perceptive-pol=1, c. extensional-pol=1, d. extensional-pol=0, e. perlocutive, f. information-pol=1, g. information-pol=0, h. possession-pol=1, i. possession-pol=0, l. positional-pol=1, m. positional-pol=0, n. permissive, o. coercitive, p. ask_poss, q. at_posit, r. not_exten, s. not let;
1.4. arguments of the predicate:
   1.4.1 syntactic classes: NP, AP, PP, VP, S, PPpred, NPPred, APpred, AdverbSN;
   1.4.2 preceding particles: all prepositions
   [1. A 2. DA 3. DI 4. IN 5. SU 6. CON 7. PER 8. TRA 9. CONTRO a. VERSO b. DA_ PARTE_ (DI) c. DAVANTI A d. INTORNO A e. (MATTER) IN, DI, f. (LOCAT/MANNER) IN, DI, A), g. (LOCAT), IN, A, SU, SOPRA, SOTTO.
1.5 semantic features of predicate-argument structures:
   1.5.1. inherent features of obligatory arguments (NPs, APs, PPs) of the predicate: [+ANIM], [-ANIM], [+HUM], [-HUM], [+ABST], [-ABST], [LEXICALLY SPECIFIED], [NONE], [-NUM], [ALL], [PLUR] (either one or another in a set);
   1.5.2. VP= subject control {SUBJ, OBJ, OBJ2};
   1.5.2.1. syntactic feature [+ restruct];
   1.5.3. S= subject control {SUBJ/X, SUBJ, X};
   1.5.3.1 propositional {excl, quest, prop};
   1.5.4. XPPred = subject control by a lexical redundancy rule associated with each predicative argument

4.2.2. Adjective Classes

2.1. adjective class: 1. predicative, 2. attributive, 3. predicative/attributive, 4. "tough" predicates, 5. transitive, 6. factive;
2.2. syntactic classes: 1. imperative, 2. non-imperative, 3. quasi-imperative, 4. unaccusative;
2.3 semantic classes: 0.state, 1.measure, 2.origin, 3.quantity/modality, 4.time, 5.evaluating, 6.colour, 7.intensional, 8.hyperintensional; 2.4. arguments of the predicate: as for verbs

4.2.3. Noun Classes
3.1. nominal classes: 0.pure nominal, 1.transitive, 2.unaccusative, 3.unergative, 4.reflexive, 5.copulative, 6.ambiguous(active/passive), 7.impersonal, 8. psych, 9.passive
3.2. semantic typology: 0.proper nouns(only +human), 1.geographical, 2.common
1.locative, 2.quantity/modality, 3.temporal, 4.legal institution, 5.profession, 6.general, 7.species, 8.relations, 9.events, 0.subjective;
3.3. inherent features, as for arguments of the predicate;
3.4. arguments of the predicate as for verbs

4.2.4. Rules for the transduction from Grammatical Functions into Semantic roles
Here below we indicate some of the rules for decoding from functional labels into theta-roles. The translation from functional specifications into semantic roles works according to the following scheme:

• all SUBJECT arguments are translated into AGENT role
• all OBJECT arguments are translated into THEME role

This rule constitutes the default value assigned to Subject and Object arguments. This default rule however is applied after all exceptions have been computed. The exceptions make use of a number of features available in Level 1 representation and can be divided up into syntactic and semantic exceptions to the default rule.

1. Exceptions to Subject decoding:
a. Syntactic
- the verb is classified as Psychic1 ----> CAUSER_EMOTIONAL
- the verb is classified as Psychic2 ----> EXPERIENCER
- the verb is classified as Inherent Reflexive ----> THEME_AFFECTED
- the verb is classified as Perception verb ----> PERCEIVER
- the verb is classified as Raising verb ----> THEME_BOUND
- the verb is classified as Unergative ----> ACTOR
- the verb is classified as Unaccusative ----> THEME_AFFECTED
- the verb is classified as Copulative ----> THEME_UNAFFECTED
- the verb is classified as Copulative and Reflexive ----> THEME_AFFECTED
- the noun is classified as Passive ----> THEME
- the noun is classified as Ambiguous and is a Pure Nominal -> POSS/CREAT
- the noun is classified as Copular ----> APPOST
- the noun is classified as a Pure Nominal ----> POSS
- the adjective selects an NP ----> HEAD
b. Semantic
- the features associated to the Subject are NOT [+HUMAN] ----> CAUSER
- the verb is classified as Unergative and is STATIVE ----> THEME_UNAFFECTED
- the verb is classified either as Inherent Reflexive or as Psychic and the feature associated to the Subject is [+HUMAN] ----> THEME_EMOTIONAL
- the verb is classified as Copulative and stative ----> THEME_BOUND
- the verb is classified as stative and [+human] ----> EXPERIENCER
- the verb is classified as "possessional, pol=0" ----> RECIPIENT
- the verb is classified as "possessional, pol=1" ----> OWNER
- the verb is classified as "possessional" and is stative ----> POSSESSION
- the verb is classified as "possessional, pol=0" and is reflexive ----> TRANSFER_OBJECT
- the verb is classified as "subjective" ----> INSTIGATOR
- the verb is classified as "informational, pol=1" ----> SOURCE_INFORMATION
- the verb is classified as "informational, pol=0" ----> ADDRESSEE
- the verb is classified as "manipulative" and is reflexive ----> GOAL
- the verb is classified as "possessional", is stative and is not reflexive ----> LOCATIVE

2. Exceptions to Object decoding
   a. Syntactic
   - the verb is classified as agentive ----> THEME_AFFECTED
   - the verb is classified as Inherent Reflexive or Psychic1 ----> EXPERIENCER
   - the verb is classified as Psychic2 ----> THEME_EMOTIONAL
   - the verb is classified as Raising ----> THEME_BOUND
   - the noun is classified as pure common Noun ----> AGENT
   b. Semantic
   - the verb is classified as "accomplishment, pol=1" ----> THEME_EFFECTED
   - the verb is classified as "informational" ----> THEME_UNAFFECTED
   - the verb is classified as stative ----> THEME_UNAFFECTED
   - the verb is classified as "manipulative" and the selection is [+human] ----> GOAL
   - the verb is classified as "manipulative" ----> THEME_UNAFFECTED
   - the verb is classified as "subjective" ----> ADDRESSEE
   - the verb is classified as "possessional" ----> POSSESSION
   - the verb is classified as Unaccusative and the Aspectual feature is ACTIVITY ----> AGENT
   - the features associated to the Object are either [+HUM] or [+ANIM] ----> PATIENT

The remaining decoding instructions are quite simple and are derived straightforwardly from functional labels. Some difficulties derive from ambiguous prepositions for Oblique argument decoding, and need manual intervention. In particular,

a. CON (with) PP argument can either be computed as COMITATIVE or as INSTRUMENTAL; the default value is INSTRUMENTAL and exceptions are both syntactic and semantic, as follows,
   - the verb is Reflexive ----> COMITATIVE
   - the feature associated to the PP argument is either [+HUM] or [+ANIM] ----> COMITATIVE
b. another exception derives from PP with preposition VERSO(towards) which can be computed either as BENEFACTIVE or as DIRECTION; the default value is BENEFATIVE and the other value is derived as follows,  
- the verb is classified as Reflexive  

-----> DIRECTION


c. another exception derives from PP with preposition PER(for) which can be computed either as PLACE or as BENEFACTIVE; the default value is BENEFACTIVE and the other value is derived as follows,  
- the verb is classified either as Psychic or as Reflexive inherent  

-----> PLACE

d. nouns' oblique argument PP with DA(from) preposition is not computed as SOURCE but as GOAL if the noun is a Pure Nominal.

More complex and inherently semantic disambiguation procedures are required for the following classes of noun predicates:

e. nouns' oblique argument PP with DI(of) preposition is computed as SPECIFICATION if the noun is a Pure Nominal and it is not EVENTive;

f. nouns' oblique argument PP with PER(for) preposition is computed as GOAL if the noun is a pure nominal and is neither SUBJECTive nor EVENTive;

g. nouns' oblique argument PP with PER(for) preposition is computed as GOAL if the noun is a reflexive and is EVENTive.

Among the conceptual/semantic classes there two classes which are underspecified: EXTENSIONAL and PERLOCUTORY. An extensional predicate may have to be further specified in case, for instance, the domain requires a notion of ingesting a solid vs drinking a liquid. The perlocutory or performative class includes all predicates which ensue in an event which has a social outcome, which require a certain social status in order to be adequately performed, which have a legal domain as inherent and so on.

The need to increase the number of theta/semantic roles is determined by the requirements of both conceptual representations and syntactic and semantic parsing. The basic idea is that whenever an internal argument/object is actually to be computed as an Affected Theme there must be an Agent as Subject. Real Affected Themes are those which undergo a change of state as a result of the action denoted by the predicate. There is a great number of rules of grammar which are sensitive to the partition of OBJects into Affected Themes versus non-Affected Themes. In particular, all predicates involving some kind of change in the OBJect NP may have a Resultative as semantically adequate secondary predication of the OBJect. The same applies to predicates of change of position, of possession, of information which do not activate Affected Themes being usually intransitive verbs – but may have Resultative. This does not apply to stative predicates, as for instance psychic predicates. Also consider the partition of Agents vs. Recipients where the former give rise to CAUSE as a conceptual template, while the latter requires LET. Finally all alternates like go/come, sell/buy, and others discusses by B.Levin(1986) including the anti-causative alternation. For this reason we have increased the number of theta-roles which map onto SUBJect, OBJect and other grammatical function on the basis of conceptual fields. Alternates are accounted for by the presence of the semantic attribute of polarity: kill, has polarity=0, while create has pol=1; buy has polarity=0, while sell has polarity=1 because the seller possesses the object at time t1 whereas the buyer does not possess it, and so on and so forth.
4.3. Some Rules from the Transducer Algorithm

```c
if (c == '0') {
    if (act) {strcat(output, "STAYexten(AT"); }
    else if (td || ingr) {strcat(output, "GOexten(TO"); }
    else strcat(output, "IDENTexten(AT");
}
```

```c
abbassare[CAUSE(<agent>(STAYmeas(PLACE-AT<theme_affect>))){(e1(t1,t2,t3))}]}
```

```c
credere[BE(<exper>(IDENT-IN SUBJ-MIND(REP<prop_nonpred>)){(en,tn})]
```

5. FROM LINGUISTIC THEORY TO CONCEPTUAL REPRESENTATION

Conceptual representations produced by our transducer are derived from theoretical work done by Jackendoff and Gruber. In this section we describe the theory behind it. In accordance with the principle of meaning decomposition, we assume that concepts denoted by lexical items are made up of primitive concepts which can be expressed by the use of a very limited number of templates. The granularity of the description depends strictly on the (sub)domain and the aim of the task at hand. For instance, abstract concepts like “responsible” or “responsibility” when dealt with in a legal subdomain require a specification of preconditions which is different from what is expected in a generic domain.

A method for the decomposition of lexical information should represent a principled way to organize a taxonomy of the concepts in a language, categorized by sets of features, which however are tightly interleaved with argument structure and the syntactic nature of each argument.

Basic constituents for our conceptual representations are spatial primitives on the basis of analogical relations existing in the spatiotemporal realm which is at the heart of the meaning of all verbs and deverbal nouns and adjectives. According to Jackendoff and Gruber, human beings seem to base their descriptions of any kind of experience on some crucial concepts drawn directly from what might be called the spatial semantic field. Similarly, temporal sequence is both perceived and expressed on the model of spatial sequence. Events and states are located in time - on a timeline - just as things and entities are located in space. The same prepositions are used both for spatial and temporal expressions.
In theory, it should be possible to describe the basic conceptual components of meaning of any verb given a finite number of spatial primitives and of modalities attached to them. Modalities describe a bit/portion of meaning of a lexical item when decomposed into conceptual primitive functions, and adds to them a certain modality. This is not to be confused with negation, which is itself an operator preceding and having scope over conceptual functions, as implied by the meaning of lexical items – more on this below.

When analysing utterances, we are interested in interpreting their meaning: this implies three basic things,

A. knowing if the meaning has to be related to the actual world or not, and if so, where should it pick up its reference from;
B. knowing what consequences the action or state of affairs described by the predicate-argument structure of the predicate(s) appearing in the utterance brought about, i.e. what can be inferenced. This is partly contained in the conceptual decomposition; and partly extractable from aspectual classes, tense, and the presence of time adverbials;
C. knowing the internal structure of temporal representation which varies basically according to lexical representation, but as we saw, can undergo dramatic structural modification in case of the intervention of surface level constituents of meaning, since we know that temporal interpretation must be built compositionally.

As for the conceptual level of representation, we take Jackendoff's theoretical framework as the basis of our work. However in our framework, conceptual representations are not primitives, they are derivative on low level representations and depend heavily on the semantic/pragmatic class of the verb. Also theta or semantic roles are derivative in our system.

To discuss this point briefly, let us consider a few predicates. Copulative verbs like BE and HAVE in our lexicon are assigned different lexical forms in case they may appear in different structural contexts: BE may be inserted or may appear with an open complement, a generic XCOMP, or it may have an infinitive as SUBJECT/prop. On the contrary, the verb HAVE only appears with NCOMP/prop.

The result of this choice is that there is no way to map the semantic and conceptual interpretation directly onto a specific lexical form at word level, these being dependent on the kind of lexical complement analysed in a particular sentence. For instance, the existential-locative meaning of the verb BE would have to be mapped once a PCOMP with locative meaning is analysed: however, this meaning does not depend on the presence of a particular preposition alone. It is both geared on selectional restrictions of the head noun and the preposition. In "John is in a hurry" we just want to say that John is an EXPERIencer and that there is a STATE in which he is in, and this state is a PCOMP whose head is "hurry". This would correspond to the Italian counterpart which uses HAVE rather than BE, "Gino ha fretta", and an NCOMP rather than a PCOMP.

Thus, there is no way to map the "possessive meaning" of HAVE onto a specific lexical form, since this meaning depends strictly speaking on the kind of nominal heads being analysed in the SUBJECT and the NCOMP. Thus "John has a fever" means again that John is EXPERIencing a particular STATE; on the contrary, "John has a car" will have to be computed with John as OWNER and the NCOMP as indicating POSSESSION. More on this topic below.

At the same time, we want to be able to map different conceptual representation onto separate syntactic lexical forms in case it is needed: take for instance, the Italian "chiedere" for “ask” which has only one underlying lexical form made up of the same combination of functional labels but with different semantic roles:
i. \text{pred(chiedere \{LET(<address>(GO(REP(<informtn><sourceA>)))))\})}

ii. \text{pred(chiedere \{CAUSE(<agent>(REP(GO-POSSinf(<from<theme>>(GO-circ(<from<ownerA>))))))\})}}

where we see in i. the conceptual structure for ask/inquire for information and in ii. the structure for ask as demand or request someone to be allowed or given. These representations derive from the corresponding syntactic representations where we see that no difference exists at the level of constituency and functional labels:

\begin{align*}
\text{pred(chiedere,tr,achiev,inform,[np/subj1/addres/[+hum],np/obj1/informtn/} & \text{[+abst],obj2/source/a/[/])]} \\
\text{ii. pred(chiedere,tr,achiev,ask_poss[np/subj1/agent/[+hum],np/obj1/theme_affect/} & \text{[-ani],obj2/owner/a/[/])}}
\end{align*}

Thus semantic role and ontological categories are determined by the semantics of the head, its aspectual class and selectional restrictions which in turn depend on syntactic and semantic classes of the predicate.

Conceptual representations are of paramount importance when we consider the case of predicates containing negation or some other modality, like "difficult" in a predicate like \text{MANAGE}, or else "guilt" in \text{MURDER}. Predicates belonging to the class of "performatives" may be decomposed into primitives which involve the assignment of properties or social roles as the main meaning of the verb. To this aim, consider a predicate like \text{MARRY}, which could be assigned a conceptual representation as the following one,

\begin{align*}
\text{pred(marry, \{CAUSE(<actor>(GO[to_exist(Property(x)]) (PATH-TO-IDENT <actee>(GO [to_exist (Property(y)])))) \{(Gender=mas, Property=husband ; Gender=fem, Property=wife), Property(x) \neq Property(y) \})}}
\end{align*}

In other words, to marry someone corresponds to cause a new property to be existent in the world and to associate this property to the actor or actee of the event according to their gender. Consequently, a predicate like \text{DIVORCE} would inherit from process verbs like \text{END} its basic meaning, i.e. to divorce a wife or husband means to put an end to a property or social role, basically:

\begin{align*}
\text{pred(divorce, \{CAUSE(<actor>(GO[to_end(Property(x)]) (PATH-TO-IDENT<actee>(GO[to_end (Property(y)])))) \{(Gender=mas, Property=husband ; Gender=fem, Property=wife), Property(x) \neq Property(y) \})}}
\end{align*}

Consider also predicates like \text{PREVENT}, \text{HINDER}, \text{PROHIBIT}, \text{FORBID}, \text{BAN}, \text{DESTROY}, \text{END}, \text{LACK}, \text{INTERRUPT}, \text{BEGIN} which contain information related to negation and permission rather than simply causation, a semantic component which is important in the semantic interpretation of the sentence. We associate the following descriptions to some of these predicates,

\begin{align*}
\text{pred(distruggere[CAUSE(<agent>(GO[nonexist]<theme_aff>))])/ destroy}
\end{align*}
Another fundamental property of our conceptual representations is the event structure which is automatically generated from aspectual information. Differently from what happens in Jackendoff’s representation, we assume temporal properties of events and states to constitute a separate layer of information from conceptual primitives. This will be further commented in the following section. In our conceptual representations, there are basically five types of event structure:

- **states**, represented as \{ (e\textsubscript{n}, t\textsubscript{n}) \}
- **activities**, represented as \{ (e\textsubscript{1}, t\textsubscript{n}) \}
- **achievements**, represented as \{ (e\textsubscript{1}, t\textsubscript{n}) \}
- **gradual accomplishments**, represented as \{ (e\textsubscript{1}, t\textsubscript{n}), (e\textsubscript{2}, t\textsubscript{2}) \}
- **accomplishments**, represented as \{ (e\textsubscript{1}, t\textsubscript{1}), e\textsubscript{1}, t\textsubscript{n}, (e\textsubscript{2}, t\textsubscript{2}) \}

Each representation contains two indices, one E for subevent time, and another T for the actual temporal interval in which the current event takes place. The subscript may indicate: beginning subevent, 1 and ending subevent, 2 into which the overall event described by the lexical entry is subdivided. The presence of a N subscript indicates openness or lack of temporal closure of the event: this applies to all types of actions apart from Achievements which are punctual and have no temporal extension whatsoever.

The following is a list of the most important types taken from the 50 semantic and conceptual classes making up our inventory together with event structures contained in our lexicon: their relevance in meaning representation and text understanding will be fully explained below.

\[
\begin{align*}
&\text{domostrare} \{ \text{LET}(<\text{agent}>(\text{GO(REP(<theme_unaff>(TO<percv_a>))))\{(e1,tn)\})} \\
&\text{costruire} \{ \text{CAUSE}(<\text{agent}>(\text{GO(TO[exist]<theme_eff>)))\{(e1,t1),e1,tn,(e2,t2)\})} \\
&\text{ascollare} \{ \text{CAUSE}(<\text{percv}>(\text{REP(\text{GO(TO<theme_unaff>))}))\{(e1,tn)\})} \\
&\text{descrivere} \{ \text{CAUSE}(<\text{source_info}>(\text{GO(REP(<informtn>(TO<address_a>))))})\{(e1,tn)\}) \\
&\text{derubare} \{ \text{LET}(<\text{receive}>(\text{GO(<possesn>(FROM<ex_owner>))))})\{(e1,tn),(e2,t2)\}) \\
&\text{chiedere} \{ \text{LET}(<\text{address}>(\text{GO(REP(<informtn>(FROM<goal_a>))))})\{(e1,t1)\}) \\
&\text{contraddire} \{ \text{CAUSE}(<\text{agent}>(\text{GOreact(TO<theme_eff>))))\{(e1,t1)\}) \\
&\text{dimagrire} \{ \text{CAUSE}(<\text{causer}>(\text{GOmeas(TO[minor])))\{(e1,t1),(e2,m)\}) \\
&\text{andare} \{ \text{CAUSE}(<\text{agent}>(\text{GOposit(FROM[x](TO<locat_a/in/da/sopra/sotto>))))\{(e1,t1),e1,tn,(e2,t2)\}) \\
\end{align*}
\]
Inducing Fully Specified Lexical Representations

The full inventory of conceptual primitives produced for our lexicon is included here below and is represented in table 1. where we also associate one conceptual representation and some example verbs:

- "costringere" [CAUSE(<agent>(GOcoerc(<theme_aff><prop/v_pred_a>)))]{(e1,t1)}
- "domandare" [CAUSE(<agent>(REP(GO(<theme_aff><prop/v_pred_a>)))](e1,t1)]
- "esistere" [BE(<theme_unaff>(STAYposit(AT)))]{(en,tn)}
- "mancare" [BE(<theme_affect>(NOT(STAYexten(<theme_a>))))]{(en,tn)}
- "vietare" [CAUSE(<agent>(GOcirc(<prop/v_pred_di>))))]{(e1,t1)}
- "seguire" [CAUSE(<agent>(GO(AFTER<theme_aff>)))]{(e1,tn)}
- "toccare" [CAUSE(<agent[plur]>)(STAYtouch(AT<comit_con>[refl]))]{(e1,t1)}
- "dimezzare" [CAUSE(<agent>(GOsegmnt(<theme_aff>))))]{(e1,t1)}
- "acchiappare" [CAUSE(<agent>(STAYhold(AT<theme_aff>))))]{(e1,t1)}
- "inseguire" [CAUSE(<causer>(GO(AFTER<difclt><theme_aff>))))]{(e1,tn)}
- "unire" [CAUSE(<theme_aff[plur]>(STAYunite(AT<theme_in>[refl])))]{(e1,tn)}
- "contrastare" [CAUSE(<agent>(GO(AGAINST<theme_aff>)<instr_con>))]{(e1,tn),(e2,tn)}
- "inghiottire" [CAUSE(<agent>(GOingest(<theme_aff>))))]{(e1,tn)}
- "cedere" [CAUSE(<agent>(NOT(STAYreact(<theme_a>))))]{(e1,t1)}

The full inventory of conceptual primitives produced for our lexicon is included here below and is represented in table 1. where we also associate one conceptual representation and some example verbs:

- exten, subject, hyper, manip, factive, evaluat, proprty, ment_act, process, measu_maj, percut, perfect(end/exist), perform, inform(at/to), possess(to), possess(from), inform(at/from), measu_min, positi(at/to), react, positi/origin(from), exten_neg, let, coerc, ask_posi, at_posi, not_exten, not_let, dir, touch, divide, hold, hole(into), dir_difclt, unite, go_against, ingest, perform_to, rep_contr, not_react, dir_tow, over, go_through, color, quantity, follow.

5.2. A General Introduction to CRs

In this section we comment extensively on Jackendoff’s theory and highlight differences from ours. As said above, the main feature of conceptual representations is the close relationship existing between the temporal field and the spatial field, which highlights the ground similarity existing between all the domains which can be detected in the mind's organization of input from experience. The number and precise features of all conceptual fields have not yet been thoroughly investigated, nor do the 46 classes above intend to cover all facets of reality. Since conceptual structures pertaining to any domain of experience are closely related to those concerning the spatial field, a preliminary analysis of sentences that express spatial location and motion is required.

The content of conceptual representations is as follows:

A. a set of primary functions which are, GO, BE, STAY, CAUSE, LET, ORIENT, IDENT and might all be preceded by negation NOT;
B. a fixed finite number of semantic fields distinguishing common areas of meaning in real knowledge of the world, like INFORM, POSSESS, EVAL, SUBJ, HYPER, PERCPT, MANIP, FACTV, MENT.ACT, PROPR, MEASU, POSIT, COERC, ASK, REACT, TOUCH, HOLD, HOLE, DIR, DIVID, UNIT, LET, etc.

C. a small number of directions indicators, FROM, TO, INTO, AGAINST, AT, TOWARD

D. a small number of secondary functions which are REP, TR

E. a finite set of modality operators with scope on the verb meaning and its complements, which include the following: [exist, nonexist, major, minor, violent, difficult, perf]

Table 1. Semantic Classes And Conceptual Representations

<table>
<thead>
<tr>
<th>Class</th>
<th>Conceptual Representation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 = exten</td>
<td>(STAY/GO/IDENTTexten(AT/TO) lavorare</td>
</tr>
<tr>
<td>1 = subjct</td>
<td>(STAY!GO/IDENTin_subj_mind(REP credere</td>
</tr>
<tr>
<td>2 = hyper</td>
<td>(ORIENThyper(TOWARD fingere</td>
</tr>
<tr>
<td>3 = manip</td>
<td>(STAY/GO/IDENTcirc(AT/TO minacciare</td>
</tr>
<tr>
<td>4 = factive</td>
<td>(IDENTfactv(REP/TR sapere</td>
</tr>
<tr>
<td>5 = evaluat</td>
<td>(ORIENTeval(TOWARD piacere</td>
</tr>
<tr>
<td>6 = proprt</td>
<td>(STAY/GO/IDENTpropr(AT/TOWARD divenire, assomigliare</td>
</tr>
<tr>
<td>7 = ment_act</td>
<td>(STAY/GO/IDENTcirc(AT/TO dimenticare, comprendere</td>
</tr>
<tr>
<td>8 = process</td>
<td>(STAY/GO/IDENTcirc(REP(AT/TOWARD continuare</td>
</tr>
<tr>
<td>9 = measu_maj</td>
<td>(STAY/GO/IDENTmeas(AT/TO-[major] crescere</td>
</tr>
<tr>
<td>10 = percept</td>
<td>(STAY/GO/IDENT(REP(AT/TO mostrare, esibire</td>
</tr>
<tr>
<td>11 = perfect</td>
<td>(GO(TO[end]) - (GO(TO[exist] finire, creare</td>
</tr>
<tr>
<td>12 = perform</td>
<td>(GOcirc[perf] battezzare</td>
</tr>
<tr>
<td>13 = percept</td>
<td>(REP(STAY/GO/IDENT(AT/TO ascoltare, udire</td>
</tr>
<tr>
<td>14 = inform</td>
<td>(STAY/GO/IDENT(REP(AT/TO c apire, comunicare</td>
</tr>
<tr>
<td>15 = possess</td>
<td>(STAY/GOposs(AT/TO dare, vendere, possedere</td>
</tr>
<tr>
<td>16 = possess</td>
<td>(GOposs(FROM comprare, ricevere</td>
</tr>
<tr>
<td>17 = inform</td>
<td>(STAY/GO/IDENT(REP(AT/TOWARD domandare, rispondere, sapere</td>
</tr>
<tr>
<td>18 = measu_min</td>
<td>(STAY/GO/IDENTmeas(AT/TO-[minor] diminuire</td>
</tr>
<tr>
<td>19 = posit</td>
<td>(STAY/GOposit(AT/TO andare, stare</td>
</tr>
<tr>
<td>20 = react</td>
<td>(STAY/GOreact(AT/TO resistere, contraddire</td>
</tr>
<tr>
<td>21 = posit</td>
<td>(GOposit(FROM venire, arrivare</td>
</tr>
<tr>
<td>22 = exten_neg</td>
<td>(GO(TO[nonexist] distruggere, uccidere</td>
</tr>
<tr>
<td>23 = let</td>
<td>(LET(GOcirc permettere, aiutare</td>
</tr>
<tr>
<td>24 = coerc</td>
<td>(GOcoerc(TO[exist] costringere</td>
</tr>
<tr>
<td>25 = ask_poss</td>
<td>(REP(GOinf(FROM chiedere</td>
</tr>
<tr>
<td>26 = at_posit</td>
<td>(STAY(AT essere - locativo</td>
</tr>
<tr>
<td>27 = not_exten</td>
<td>(NOT(GOexten(TO[exist] mancare, escludere</td>
</tr>
<tr>
<td>28 = not let</td>
<td>(NOT(LET(GOcirc ostacolare, vietare</td>
</tr>
<tr>
<td>29 = dir</td>
<td>(GO(AFTER seguire, corteggiare</td>
</tr>
<tr>
<td>30 = touch</td>
<td>(STAY/GOtouch(AT/TO toccare, accarezzare</td>
</tr>
</tbody>
</table>
31 = divide (GOsegmnt(TO separare
32 = hold (STAY/GOhold(AT/TO tenere, acciappare
33 = hole (GOhole(INTO perforare, bucare
34 = dir_dfc (GO(AFTER[difclt] inseguire
35 = unite (STAY/GOunite(AT/TO unire
36 = go_against (GO(AGAINST attaccare assalire
37 = ingest (GOingest(INTO ingerire, ingoiare
38 = perform_to (GOperf(TO incaricare
39 = rep_contr (REP(STAY/GO/IDENTcontr(AT/TO controllare
40 = not_react (NOT(GO(STAYreact(TO - cedere, subire
41 = dir_tow (GO(TOWARD dirigere, deviare
42 = go_over (GO(OVER sorpassare, vincere
43 = go_through (GOexten(FROM (TO andare, tradurre
44 = colour (GOexten(FROM (TO dipingere
45 = quantity (GOexten(FROM (TO andare, tradurre
46 = follow (GOexten(FROM (TO andare, tradurre

Finally, there is a generic evaluative polarity which simply accompanies the concept and encodes the way in which its meaning is perceived in a default manner as having a positive or a negative import: kill, die, destroy are computed as GO(TO[nonexist] X ) negative); on the contrary create, be born, heal are computed as GO(TO[esist] X) positive).

5.2.1. Spatio-Temporal Locations

The canonical form of a sentence describing spatial location or motion is SUBJ PRED OBL, where the subject refers to a #thing#, the OBL refers to a #place# or #path#, and the verb specifies the relationship between the #thing# and the #place# or #path#. The sentence as a whole refers to a #state# or an #event#, according to the semantic properties of the verb. (PLACE) concepts may consist of spatial location adverbs or have the internal structure:

- (PLACE PLACE-FUNCTION (THING))

where the place-function is the preposition in the Locative and the (THING) is its object. Place-functions impose conceptual constraints on the reference objects expressed by the arguments they strictly subcategorize. The reference object of the place-function IN, for example, has to be conceived as a bounded area or volume. (PLACES) project into #points# or #regions#, and are occupied by (THINGS) or specify the location of the event or state expressed by the sentence.

(PATH) concepts may consist of a path-function and a reference object or a reference place. (PLACES) can thus be embedded in (PATHS). A few English prepositions express both a path-function and a place-function: this is the case of "onto" and "into", which are decomposed in conceptual structure into

- (PATH TO (PLACE ON (THING))) and (PATH TO (PLACE IN (THING)))
respectively. Other prepositions (e.g. "under") express either a path-function or a place-
function, depending on the context. Three types of paths can be distinguished with respect
to their relationship to the reference object or place: bounded paths, that include source
paths and goal paths and in which the reference object or place is an endpoint of the path;
directions, in which the reference object or place would fall on the path only if the latter were
somewhat extended; and routes, in which the reference object or place is connected to some
point in the path. In Jackendoff's model for conceptual structure representations TO and
FROM are used as basic functions for bounded paths, TowARd and AWAY-FROM for
directions and VIA for routes.

The basic possible representations for (PATHS) therefore are:

- TO/FROM(THING Y), TOWARD(PLACE Y), AWAY-FROM, PATH/VIA

where the (PLACE) constituent is further expanded according to the general structure of
(PLACES). All sorts of spatial sentences can be formally represented by means of five basic
conceptual structure frames which make use of five kinds of constituents, i.e. (EVENT),
(STATE), (THING), (PLACE) and (PATH) – three in ours -, and four semantic primitives
that capture the most relevant features of the meanings of verbs expressing spatial concepts:
GO, STAY, BE and ORIENT. Event and States appear as conceptual primitive constituents
here, differently from what happens in our representation. Jackendoff (1983:173) employs the
primitive GO modified by the subscript "Ext" in the representation of predicates of extent.
Subscripts are written in block letters and joined to the primitives by a hyphen in the present
work. The five conceptual structures are:

a. (EVENT GO ((THING X) (PATH Y)))
b. (EVENT STAY ((THING X) (PLACE Y)))
c. (STATE BE ((THING X) (PLACE Y)))
d. (STATE ORIENT ((THING X) (PATH Y)))
e. (STATE GO-EXT ((THING X) (PATH Y)))

X stays for the subject – which may be a Theme or an Actor, and Y for location.
Causative functions are represented by means of two more primitives: CAUSE and LET. The
canonical form of spatial causative sentences is one in which the subject refers to the Agent
and the object to the Theme. Such sentences are formally represented by the structures:

a. (CAUSE ((THING X) (EVENT Y)))
b. (LET ((THING X) (EVENT Y)))

Since Agents cause things to happen, (STATES) are not eligible to appear as the second
argument of CAUSE. LET, on the other hand, may also be paraphrased roughly as "cause to
persist in a state" and can consequently take a (STATE) as its second argument.

Also, we use theta/semantic roles rather than Thing/Event arguments of the main
conceptual primitive predicates GO/STAY/BE/CAUSE/LET, and the noun predicate itself,
which we do not include here in the representation not to complicate it too much. Thing,
Time, Event etc. are treated as semantic inherent features which are associated to the lexical
form and which do not enter in the conceptual representations.
Only minor changes are necessary in order to adapt this model to the representation of utterances which do not concern the spatial field, since the various semantic domains display similar grammatical and lexical patterns. Jackendoff (1983:188), elaborating work by Gruber (1965), captures the relationship between semantic fields in the Thematic Relations Hypothesis (TRH). In any semantic field, the principal event-, state-, path-, and place-functions are a subset of those used for the analysis of spatial location and motion. Fields differ in only three possible ways:

a. what sorts of entities may appear as Theme; b. what sorts of entities may appear as reference objects; c. what kind of relation assumes the role played by location in the field of spatial expressions". In the temporal field, for instance, (TIMES) appear as reference objects and the time of occurrence plays the role of location. The semantic parallelism between spatial and temporal sentences is preserved in conceptual structure, where spatial functions take the "suffix" TEMP (which works as a modifier) to represent temporal functions. Thus, "The boy is at school" receives the conceptual structure representation:

- (BE ((boy) (PLACE AT (school))))

and "The lecture is at 10:00" receives the representation:

- (BE-TEMP((lecture) (PLACE AT-TEMP (10:00)))).

Only nouns labelled as (+TEMP) in the lexicon can appear in a (TIME) constituent. To this rather limited set of nouns belong the days of the week, the months and all the analogous substantives used to refer to time (e.g. "hour, morning, day, week, year" etc.). It is precisely the presence or absence of such a noun in the PP which makes the difference between the two sentences compared above and which brings about the selection of the appropriate conceptual structure. In order not to multiply the possible conceptual structure frames in the lexicon, when a verb can appear both in spatial and in temporal sentences only a "spatial structure" is given; the temporal one is derived by means of rules which insert the TEMP modifiers beside the primitives if a (+TEMP) noun is embedded in a strictly subcategorized PP.

As all nouns map into (THINGS) in the representational model proposed by Jackendoff, the only way to generate representations which meet the requirements stated in the conditions defining the temporal field - i.e. that only nouns or phrases which express (EVENTS) or (STATES) can appear as Theme in it - is to take into consideration the class to which the nouns belong. State and event nouns are a subclass of common abstract nouns and refer to existing or possible states or to things that happen (e.g. "destruction, birth, lecture", etc.). (STATE) and (EVENT) phrases, on the other hand, are normally represented by gerundive clauses as in "Her screaming at the least noise lasted all evening".

The parallelism between spatial and temporal expressions is very strong also as far as inference is concerned. Jackendoff (1983:190) points out that "the function GO-EXT [...] maps a (THING) and a (PATH) into a (STATE) and asserts that the (THING) occupies every point of the (PATH). When shifted in the temporal domain, [...] GO-EXT maps an (EVENT) and a temporal (PATH) into a (STATE) and asserts that the (EVENT) occupies all points in time within the temporal (PATH)" (sub-scripts and brackets have been modified in order to comply with the version of the formalism we are working out). Such a remark sounds in fact somewhat too optimistic, since although a sentence like, "The speech went from 2:00 to
4:00”, implies an extension of an (EVENT) over a time span, it does not necessarily mean that the speech included every single instant "from 2:00 to 4:00”. One may feel that "The speech went from 2:00 to 4:00" describes a state of affairs which entails that the activity took place in that time span without any major interruption, that is in all the relevant points of the time-line segment concerned. The precise determination of what such relevant points are is left to pragmatic factors. In the spatial field, on the other hand, a sentence like, "The pipeline went from Algiers to Paris", asserts that no interruption whatsoever affects the extension of the [+thing] concerned.

5.2.2. Possession

Another major semantic field regards possession. It is not a homogeneous domain, since it can be subdivided into families, among which are inalienable and alienable possession, ownership and temporary control. Jackendoff (1983) treats only alienable possession in some detail, but the way in which this family satisfies the TRH can apply to the whole field. In the possessional field a [+thing] appear as Ncomp/Poss, and "y has/possesses x" is conceptually parallel to "x is at y" in the spatial field. Verbal primitives and place- and path-functions are modified by POSS in the representations that describe sentences expressing possession. The modifier POSS is specified by the transducer in the lexical entries of the verbs. This does not involve a proliferation of conceptual frames, since only a few verbs, like "keep" and "belong", may also have spatial readings. In such sporadic cases, two separate entries are associated with the verbs in the lexicon. The parser may select the appropriate reading on the basis of the syntactic argument structure or some lexical features of the [+thing] involved. It is worth noticing that the two primitives which have been adopted for causative functions prove extremely useful in the possessional domain. The rather subtle semantic difference between, for instance, "obtain" and "accept" can be adequately represented by using CAUSE in the first case and LET in the second. The sentences:

a. John obtained the book
b. John accepted the book

c. (CAUSE ((john) (GO-POSS ((book) (PATH TO-POSS (john))))))
d. (LET ((john) (GO-POSS ((book) (PATH TO-POSS (john))))))

respectively.

The conceptual structures are related to the thematic relations in the sentences. In the first case "John" is an Agent who has done something to bring about the realization of an action, in the second he simply consents to be the goal of a change-of-possession event. The role played by the grammatical subject in the second sentence is often called "Theme" in the literature, but it is preferable to reserve such label for the [+thing] whose motion or location is described (i.e. "the book"). An alternative choice might be to call it "Patient", which is the label traditionally attributed to the [+thing] affected by the action, but this definition does not describe appropriately a person accepting a book. In addition, the subject "John" fails the tests for the Patient role indicated by Jackendoff (1987:394), since it can not appear in either of the following sentences:
a. "*What happened to John was he accepted the book"
b. "*What John did to the book was accept it".

A tentative solution to this problem may be to exploit the Actor role to cover such cases, that is the first arguments of the conceptual causative function LET, and to adopt some other label to specify the role played by inanimate first arguments of the CAUSE function in sentences like:

a. "The wind pushed me forward"
b. (CAUSE ((wind) (GO ((me) (PATH FORWARD)))))

In each of the representations of the sentences, "John obtained the book" and "John accepted the book", the conceptual constituent (john) appears twice. This accounts for the intuition that "John" is both the Agent/Actor and the Goal of the action. As was already mentioned, there may be conceptual constituents in the meaning of a sentence which do not correspond to any syntactic constituent. This shows up frequently in the possessional field, where the semantics of various verbs encode constituents which are not realized syntactically. "Buy" and "sell", for instance, imply transfer of money from the buyer to the seller. The presence of this feature in conceptual structure representations even when it does not appear in the syntax makes it possible to distinguish these verbs from the semantically overlapping "obtain" and "give".

The semantic field that concerns categorization and the attribution of properties is called identificational. In it, a [+thing] appear as reference objects, and "x is an instance of y" or "x has the property y" plays the role of spatial "x is at y". The NPs representing the reference objects appear grammatically as predicate nominals.

This field includes such verbs as "become" and "change", and the readings of some verbs like "be" and "stay" which correspond to the syntactic argument structure NP1 V NP2 or NP V AP. The identificational acception is expressed in conceptual structure by a modifier IDENT. A sentence like: "Grass is green", for example, is represented as:

- BE-IDENT ((grass) (PLACE AT-IDENT (green))).

There are also extensional identificational functions. These represent particular readings of verbs which already have an extensional meaning in the spatial field. One such verb is "range", which receives the conceptual structure frame:

- GO-EXT ((X) (PATH (FROM-EXT (Y)) (TO-EXT Z)))

when it appears in the spatial field and:

- (GO-EXT-IDENT ((X) (PATH (FROM-EXT-IDENT (Y)) (TO-EXT-IDENT (Z))))

when it is part of a sentence like: "Prices here range from $1 to $10,000".

To the identificational field also belong comparative adjectives. They express in fact an "identificational direction, a path whose endpoints are not specified" (Jackendoff 1983:196).
This provides an adequate treatment of sentences in which they appear. "The balloon became smaller", for instance, is represented as:

- (GO-IDENT ((balloon) (PATH TOWARD-IDENT (small)))).

In the circumstantial field (EVENTS) and (STATES) appear as reference objects and "x is a character of y" is analogous to spatial "x is at y". The reference (EVENTS) and (STATES) are syntactically represented by the subordinate clauses which are strictly subcategorized by the verbs belonging to this domain. A sentence like: "The rat made Mary scream" receives the representation:

- (CAUSE (rat) (GO-CIRC ((mary)(PATH TO-CIRC (scream))))).

The verb "be" can be ascribed to the circumstantial field when it is used to form the progressive tenses. A sentence like: "Mary is singing" corresponds to the conceptual structure representation:

- (BE-CIRC ((mary) (PLACE AT-CIRC (sing)))).

"BE-CIRC takes a snapshot of a state in the middle of an event" (Jackendoff 1983:200), it singles out an instant in the course of an activity. As states already refer to a point in time, Jackendoff argues that it makes no sense to "freeze" them "in mid-course". This explains why progressive tenses normally apply only to nonstative verbs.

5.2.3. Existential and additional Fields

A rather restricted semantic domain is the existential field, in which (THINGS) and (STATES) appear as Theme and there is a single reference location, (EX), expressed by existence. Among the few verbs which belong to it are "create" and "make", in the sense of "cause to come into existence", and "destroy", which can be roughly paraphrased as "cause to go out of existence".

A significant number of English verbs cannot be accounted for in terms of the semantic fields considered so far. Among the "outsiders" are verbs such as "want", "like" and "love", and "see", "feel" and "hear". The first can be defined as "emotional" or "volitional" verbs, the second are usually called "perception verbs". In order to incorporate them into the representational model we assume that there are at least two more semantic domains: the emotional field and the perceptional field.

In the emotional field (THINGS) appear as Theme, (THINGS), (EVENTS) and (STATES) appear as reference objects and "x is emotionally oriented towards y" plays the role of spatial "x points at y". Sentences like, "I love marshmallows" and "I want you to sing" receive the representations:

- (ORIENT-EMOT ((I) (PATH TOWARD-EMOT (marshmallow)))), and
(ORIENT-EMOT ((I) (PATH TOWARD-EMOT (GO-CIRC((you) (PATH TO-CIRC (sing))]))))

5.2.4. Perceptual and Belief/Opinion Verbs

The perceptual field appears more difficult to define. (THINGS) appear as Theme, (THINGS), (EVENTS) and (STATES) as reference objects, and "y perceives x" plays the role of spatial "x goes to y". The senses that convey the perceptions may be referred to in conceptual structures. "Ian saw the cat", for instance, can be represented as:

- (GO-PERC (((cat) (PATH TO-PERC (ian's sight))))).

"Ian looked at the cat", on the other hand, corresponds to:

- (GO (((ian's gaze) (PATH TO (cat))))

and "Ian watched the cat" to

- (STAY (((ian's gaze) (PLACE AT (cat))))).

The difference in conceptual structures mirrors the subtle semantic difference between these verbs: although they all describe a similar kind of phenomenon, only "see" is a perception verb. A last category of verbs which have not yet been considered are the so-called "verbs of opinion", or "belief verbs". This class is exemplified by "think", "believe", "suppose", etc. It is also closely related to verbs like "say", "ask" and "suggest". Jackendoff (1983) notices an ontological analogy between the utterances describing beliefs and those referring to representations of the world as, for instance, pictures. He then argues that beliefs are mental representations of the world. The clause: "Mary has brown eyes" is therefore conceptually more or less the same in both of the following sentences:

a. John thinks that Mary has brown eyes.
b. In John's picture Mary has brown eyes.

A "Representational Rule" helps to mould conceptual structures for these sorts of expressions: "If a sentence S (or sequence of sentences) expresses or implies Y(BE ((X) (IN REPRESENTATIONAL ))), optionally replace every occurrence of (X) in the interpretation of S by (REP (X))" (Jackendoff 1983:225). Verbs that encode "representations of the world" have to display the feature (REP) in the lexicon. A verb like "think" is thus associated to a conceptual structure frame like:

- (BE ((REP (S)) (PLACE (in X's mind))))

which yields the representation:

- (BE ((REP (BE-POSS ((brown eyes)(PLACE AT-POSS (mary)))))) (PLACE (in john's mind))))

for the sentence "John thinks that Mary has brown eyes".
A relevant side-effect of this treatment of belief contexts is that it provides the means to represent verbs like "seem", which are not "verbs of opinion" but nevertheless describe representations, or "verbal pictures" of the world. "Mary seems to have brown eyes" is represented as:

- (REP (BE-POSS ((brown eyes) (PLACE AT-POSS (mary)))).

The sentence "Mary has brown eyes", on the other hand, receives the same representation as the sentence with "seem" except for the absence of (REP). The semantic relationship between such pairs of sentences is brought forward by conceptual structures.

A statement can also be considered as a verbal picture of the world. This justifies the use of (REP) in the conceptual structures representing what is said, asked, suggested, etc. All the clauses strictly subcategorized by these verbs are therefore embedded in (REP) in conceptual structure frames. "John suggested that Mary should sing", for instance, receives the representation:

- (GO ((REP (GO-CIRC ((mary) (PATH TO-CIRC(sing))))) (PATH AWAY-FROM (john)))).

When the representation of the world corresponds to the speaker's projection of the world, it can be said to be a "transparent description". This can be specified by an operator (TR). Jackendoff(1983) uses (TR) in the representation of a sentence like "In John's picture, the blue-eyed girl has brown eyes", which becomes:

- (BE ((REP (BE-POSS ((brown eyes)(PLACE AT-POSS (TR (blue-eyed girl)))))) (PLACE IN (john's picture))).

The application of the operator (TR) involves knowledge of the world. The only contexts in which it can be determined a priori are those related to such verbs as "know". A sentence like:"Mary knows that John's cat is black", implies that the subject's mental representation of the situation described corresponds to reality. The appropriate representation therefore is:

- (BE ((REP (TR (BE-IDENT ((john's cat)(PLACE AT-IDENT (black)))))) (PLACE (in mary's mind))).

6. MAPPING INTO THE KB FROM THE DM BY MEANS OF CRs

In Delmonte(1990) we considered CRs as the link from semantics as specified in a Discourse Model to world knowledge by means of a mapping algorithm with inference rules that allowed reasoning to be performed. As shown above, each conceptual representation carries an event structure which is used for temporal reasoning reported here below,

arrivare[BE(<th_unaff>)(STAY_{pos}(AT<locat>))]{(e_{n,t_{n}},(e_{1},t_{1}))} / arrive
Temporal Reasoning works on interval semantics and the inferential procedure has a set of basic rules which decompose CRs on more elementary predicates like BE, as follows:

\[
\text{[STAY ([X],[AT Y]) from t1 to t2] } \Rightarrow \text{ [BE ([X],[AT Y]) at t3] cond = t1 < t3 < t2}
\]

where the restriction on spatial location of the argument of the main predicate states that in order to Arrived at a given location Y at a given time t1, the participant entity X needs to Be there at time t3 which precedes t1.

Knowledge representation is fed dynamically from the list of facts produced by our system for text understanding stored in a Discourse Model, where anaphoric processes are taken care of. Facts listed in the DM only carry information related to Semantic Roles associated to a given entity in a given situation and governed by a given verb predicate. In addition to that, properties may be associated to the same entity in the same situation, where also a polarity may appear – index 1, argument slot 4 in fact. Verbal predicates are associated to a specific semantic index, K, to which the overall situation is finally linked by the predicate linkverb.

\[
generate :-
\]
\[
\text{fact(\_,isa,[arg:K,arg:ev],1),}
\]
\[
\text{fact(K,Pred,[Agent:X,Locat:Y],1),}
\]
\[
\text{associate\_primary\_function\_and\_roles(K, Pred, Agent:X, Locat:Y).}
\]

Primary function and roles is activated by unification with lexical semantic information associated to main predicate – GO.

\[
\text{associate\_primary\_function\_and\_roles(K, go, Agent:X, Locat:Y):-}
\]
\[
\text{role\_saturation(go, Agent, Locat, SuperFunct, PrimFunct, ExtraRoles),}
\]
\[
\text{db\_mapping(SuperFunct, PrimFunct, \[X, ExtraRoles,Y\], K).}
\]

Role saturation is the actual transfer of conceptual information and is where the mapping from semantic roles to conceptual information takes place:

\[
\text{role\_saturation(go, th\_aff, locat, SuperFunct, PrimFunct, [ExtraRoles]):-}
\]
\[
\text{go[GOexten(<th\_aff>(FROM<source>(TO<locat>))))\{en,tm\}],}
\]
\[
\text{SuperFunct=CAUSE, PrimFunct=GO, ExtraRoles=source,!.}
\]
\[
\text{db\_mapping(MainPrimitive,PrimaryFunction,[Agent,ArgumentRoles],SemanticIndex) :-}
\]
\[
\text{R::MainPrimitive and theme:Agent and primitiveIntroduced:Primitive,}
\]
\[
\text{introduce\_specific\_primitive(PrimaryFunction, [Agent |ArgumentRoles], Primitive),}
\]
\[
\text{linkverb(SemanticIndex,R),!.}
\]

Finally the specific primitive is associated to the conceptual meaning representation which contains reference to the “spatial” semantic field:

\[
\text{introduce\_specific\_primitive(GO, [Agent,From,To], Primitive):-}
\]
Primitive=(S::primGO and theme:Agent and from:From and to:To and semanticfieldgo:spatial),!.

\[ \text{linkverb}(A,B) : A::\text{anything} \text{ and prim}:B. \]

These rules have been used to elaborate questions and answers reported in Book 2.
Chapter 2

TREEBANKING: FROM PHRASE STRUCTURE TO DEPENDENCY REPRESENTATION

In this chapter we describe VIT (Venice Italian Treebank), focusing on the syntactic-semantic features of the treebank that are partly dependent on the adopted tagset, partly on the reference linguistic theory, and, lastly - as in every treebank - on the chosen language: Italian. By discussing examples taken from treebanks available in other languages, we show the theoretical and practical differences and motivations that underlie our approach. Eventually, we discuss the quantitative analysis of the data of our treebank comparing them to other treebanks. In general, we try to substantiate the claim that treebanking grammars or parsers is dramatically dependent on the chosen treebank; and eventually this process seems to be dependent both on substantial factors such as the adopted linguistic framework for structural description and, ultimately, the described language.

1. INTRODUCTION

The VIT Corpus consists of 60,000 words of transcribed spoken text and of 270,000 words of written text. In this chapter we will restrict our description to the characteristics of written texts of our Treebank.

The first version of the Treebank was created in the years 1985-88 with the contribution of Roberto Dolci, Giuliana Giusti, Anna Cardinaletti, Laura Brugè, Paola Merlo who also cooperated in the creation of the first Italian subcategorized frequency lexicon where the first 4,000 words in the frequency list of LIF were chosen. These procedures had been promoted by means of a research program financed by DIGITAL Equipment that was interested in building an Italian version of its voice synthesizer DECTalk, i.e. a system of vocal automatic synthesis from a written text in Italian based on the one realized for American English. To this end, it was necessary to recreate the same linguistic tools of the original version: that is a robust syntactic parser for unrestricted text, a morphological analyser and a lexicon that could work with unrestricted Italian texts without vocabulary limitations. The treebank created at that time was only in paper form, because of the lack of other samples available worldwide – the one created by the University of Pennsylvania was a work-in-progress – and also for the lack of adequate software to produce annotation interactively and consistently.
The paper documents – that are still kept in the Laboratory of Computational Linguistics where they were produced – were used for the creation of a probabilistic context-free grammar of Italian, i.e. a list of all the rewriting rules produced by manual annotation and for every different rule the frequency value of the rule itself in the corpus. The chosen corpus consisted of 40,000 words taken from newspaper or magazine articles pertaining to politics, economics, current events and bureaucratic language: the texts were digitized and available on mainframe computers, but not annotated as for PoS. This phase of the work is documented in a paper (Delmonte R. & R.Dolci, 1989). Work for the creation of the treebank was then discontinuously carried on reusing the above-mentioned texts and gradually expanding the sample. This went on until the approval of the national project SI-TAL in 1998 which was also the right prompt to achieve a normalization of the overall syntactic annotation. The actual treebank uses those texts and others elaborated for the national project SI-TAL and the projects AVIP/API/IPAR as well as texts annotated on a number of internal project - as for instance one with IRST concerned with literary Italian texts.

The creation of a treebank is the last step in a long and elaborated process during which the original text undergoes a total transformation. The texts have been digitized and, if necessary, corrected – in case of orthographic or other sorts of errors, which have been removed in order to avoid unwanted and malformed syntactic structures. Subsequently, by employing the suite of automatic annotation programs by Delmonte et al. (2004), we proceeded to the tokenization of the texts, providing each word with a line or record and one or more indexes – in case the word was an amalgam or a cliticized verb. In this stage, we verified that those words consisting of a combination of letters and digits, letters and graphical signs, dates, formulas and other orthographic combinations that are not simple sequences of characters had been transformed appropriately and that no word of the original text had gone missing during the process.

From the resulting tokenized text we move on to the creation of Multiwords – more details in the following sections. This operation is accomplished using a specialized lexicon which has been created on purpose and in which one could add other forms or idiomatic expressions that have to be analyzed syntactically as one word because they constitute a single meaning and no semantic decomposition would be allowed. Inflected versions of each multiword had to be listed if needed.

In this stage we created a lexicon specialized to particular domains. This has been done in the case of the spontaneous dialogue texts of the national projects AVIP/API/IPAR (see Delmonte et al., 2004) where coding of semi-words, non-words and other forms of disfluencies has taken place; where possible the specific lexicon also contains reference to the lemma of the wordform.

Tagging is performed by assigning to each token previously found the tags or PoS labels on the basis of a wordform dictionary and of a morphological analyser that can proceed to do “guessing” in case the corresponding root cannot be found in the root dictionary – but see also Chapter 9. This operation is done by decomposing a word in affixes, inflections and derivational ones, in order to identify an existing root; in lack of such information, a word will be categorized with the temporary tag “npro” (proper noun) if uppercase or “fw” (foreign word) if lowercase. In this stage amalgamated words (e.g. DEL = Di/prep, lo/art_mas_sing), are split and two separate words are created; in addition to that, an image of the text in the form of sentences is created and these sentences will then be used for syntactic analysis which assumes the sentence as the ideal span of text. As already stated above, all steps of
morphological analysis and lemmatization together with the creation of specific lexica and phase in which we built and analysed the multiwords have required one or more cycles of manual revision.

Tagging was completed by the semi-automatic phase of disambiguation, i.e. choice of single tag associated to every word according to context. The texts we analyzed showed on average 1.9 ambiguity level: this means that every word was associated to two tags on average. To solve the problem of word disambiguation we used hybrid algorithms that are in part statistical and in part syntactical and converge in a program that has an interface for the annotation which allows the annotator to take quick decisions as to which tag to assign in the actual context even when the correct tag differs from the ones displayed by the automatic analysis. In this way, the annotator has also taken care of those cases in which the system did not have enough lexical or morphological information to process the current word.

Eventually, parsing takes place. The automatic analysis of the parser is submitted to a manual check and in the end to the collation from a supervisor who is responsible of the eventual unification of the structural “variants” suggested by different annotators for the same structural type (there were only two). This operation was critical and has required in some cases a total revision of some parts of the treebank itself, as has been the case with comparative and quantified structures in the project SI-TAL (see Delmonte, 2004).

2. DETERMINING FACTORS IN TREEBANK CONSTRUCTION

The following is a list of factors which we assume are of fundamental importance in deciding how the treebank and the underlying corpus should be organized. These factors are at the same time conditions of wellformedness of treebank and may constitute an obstacle against the usability of the same treebank for machine learning purposes. According to us, a treebank should be endowed with:

- Representativeness in terms of text genres
- Representativeness in terms of linguistic theory adherence
- Coherence in allowing Syntactic-Semantic Mapping
- Ability to highlight the distinctive linguistic features of the chosen language

Each factor can impact negatively on the linguistic texture of the treebank, and may undermine its utility in terms of general linguistic reference point for studies of the chosen language. In more detail, we assume that the factors above would have to be substantiated on the basis of the following choices:

- Corpus (Balanced) and representative of 6/7 different text genres vs. Unbalanced/Mono genre
- Strictly adherent to linguistic principles vs. loosely/non adherent (e.g. more hierarchical vs. less hierarchical)
- Constituency/Dependency/Functional structures are semantically coherent vs. incoherent
- Language chosen is highly canonical and regular vs. almost free word order language
The final item is inherent in the language chosen and not to be attributed to responsibilities of the annotators. However, as will be shown and discussed at length below, it may turn out to be the main factor in determining the feasibility of the treebank for grammar induction and probabilistic parsing. At the end of the chapter we have listed in Appendix III all main treebanks, tools to produce them and other useful websites for those interested in producing a treebank from scratch.

In the following we will discuss the linguistic framework we chose and then will make some comparisons with other treebanks.

2.1. The Theoretical Framework

Schematically speaking, X-bar theory (we refer here to the standard variety presented in LFG theory) prefigures an organization of the type head and head-projections where each head – Preposition, Verb, Noun, Adjective and ADVerbial - is provided with a bar in hierarchical order: in this way the node on which a head depends is numbered starting from 0 and the subsequent dominant nodes have a bar, two bars and if necessary other bars (even though a two-bar projection is universally considered to be the maximum level). The hierarchical organization of the theory consists of the following abstract rewrite rules, where X is used instead of one of the heads (P,A,V,N,ADV), and there is an additional functional level CP, based on Complementizer. The preterminal C0 thus corresponds to X0, Xbar is another term for X1, and XP stands for X2:

2.1.1. Theoretical Scheme of X-bar Rules

\[
\begin{align*}
CP & \rightarrow \text{Spec}, \text{Cbar} \\
\text{Spec} & \rightarrow C0 \\
C0 & \rightarrow \text{Complementizer} \\
\text{Cbar} & \rightarrow \text{Adjuncts}, \text{XP} \\
\text{XP} & \rightarrow \text{Spec}, \text{Xbar} \\
\text{Spec} & \rightarrow (\text{Subject}) \text{NP} \\
\text{Xbar} & \rightarrow X0, \text{Complements}/\text{Adjuncts} \\
X0 & \rightarrow \text{Verb}, \text{Adjective}, \text{Noun}, \text{Adverb}, \text{Preposition}
\end{align*}
\]

Spec – Specifier - is a nonterminal including constituents preceding the Head, usually modifiers, or intensifiers. In the case of sentence level, Spec contains the Subject NP for SVO languages. This rule schemata is however too weak to be of some use for practical purposes in a real corpus annotation task, because it conflates all sentence types into one single label CP. So we operated a series of tuning and specialization of the X-bar schema while at the same time trying not to betray its underlying foremost principle, which is the need that each constituent or higher projection should have only one and a single head.

Some decision were due to the need to include under the same constituent label linguistic material belonging to the specifier which in our representation is only constituted by a positional variant: i.e. all constituents coming before the head are in the specifier of that constituent. The first choice we operated had to do with the internal organization of the
specifier of NP that, in case of non-phrasal constituents, can consist of one or more linguistic elements belonging to different minor syntactic categories as reported below:

2.1.2. Atomic vs Structured Specifier

NP Spec --> Determiners, Quantifiers, Intensifiers
Verb Complex --> auxiliary verbs, modals, clitics, negatives, adverbials (also in a PP form), Verb

The choice to have a Spec structure was too difficult an option to pursue because it introduced an additional level of structure which was not easy to formalize in real texts, so we decided to leave minor non-semantic constituents that stood before the head in an atomic form, unless it required a structure of its own, which in some cases applies for quantifiers. Besides, semantic heads such as adjectives and adverbs always have their own constituent structure. As to the Verb Complex, it contained a number of atomic minor categories which we did not want to give a separate structure to if not required specifically, again in case we had a PP or an ADVerbial was preceded by a modifier. So, tensed verb takes a separate structure we have called IBAR - or IR_INFL (“unreal” verb) when the verb is either in future, conditional or subjunctive form - and that can consist of more elements added to the constituency level of the tensed verb. In view of the above, we came up with the following less generic X-bar-like scheme:

2.1.3. X-bar rules for sentence level

CP --> SpecCP, Cbar
SpecCP --> Adjuncts, Fronted Complements, Focussed Arguments, Dislocated Constituents
Cbar --> C1, IP
Cbar --> C0, CP
C0 --> Complementizer
C1 --> Wh+ word

Here IP appears and I stands for Inflection of the Inflected or tensed Verb. However, it is apparent that the rules must be specialized: Cbar in case of wh+ words can never be followed by a CP, i.e. a subordinate clause starting with a subordinator. On the contrary, when a complementizer is instantiated, CP may appear. The remaining rules are below:

IP --> SpecIP, Xbar, Complements, Adjuncts, Dislocated Constituents
SpecIP --> (Subject) NP
Complements --> COMplementTransitive-COMPT/COMplementIntransitive-COMPIN/
COMplementCopulative-COMPC/ COMplementPASsive-COMPPAS
Xbar --> VerbalComplex
Spec --> Adverbials, Quantified Structures, Preposed Constituents
2.2. Syntactic Constituency Annotation

Eventually what we wanted to preserve was the semantic transparency of the constituency representation in order to facilitate the syntax-semantics mapping if needed. In particular we wanted the CLAUSE or IP to remain the semantically transparent syntactic nucleus corresponding to a Semantic Proposition with PAS. To that purpose we introduced a distinction between Tensed and Untensed Clauses, where the second need the unexpressed Subject to be bound to some Controller in the matrix clause. Untensed clauses are Participials, Infinitivals and Gerundives which lack an expressed NP Subject universally. For that reason, linguistic theories have introduced the notion of Big PRO, as representing the unexpressed Subject of these clauses. A big PRO needs a controller – a grammatically or lexically assigned antecedent - in order for the clause to be semantically consistent. It is called controller (and not antecedent) because the syntactic structure licenses its structural location in a specific domain. Big PROs may also end up without a specific controller, thus being a case of Arbitrary or Generic reading. Antecedents are only those individuated by rules of pronominal binding or anaphora resolution.

We were also obliged to introduce specialized constituency label by the specific features of the corpus we analysed: in particular, the texts are full of Fragments or Non-verbal sequences of constituents making a sentence. Other specialized structures will be discussed further on, but now it is important to note that our representation does not employ a VP structure level: in fact, we preferred to analyse verbal group as directly positioned on the same level of S, where there will also be a NP-Subject, if syntactically expressed. We also decided to introduce a label for each of the three main lexical types specifying the syntactic category of the verbal governor to the complement structure which would thus be subcategorized according to different types of complements, among which we also introduced Voice or Diathesis to specialize the complements of a passive verb – COMPPAS, in order to allow an easy automatic conversion in case of the presence of an adjunct containing an agent in SPDA form. By doing this, VIT partially followed NEGRA, the German treebank, also in the sense of specializing major non-terminal constituents, as discussed in the sections below. While on the contrary PennTrebank (hence PT) differs for a less detailed and more skeletal choice, as specified in the PT guidelines (Marcus et al.):

“Our approach to developing the syntactic tagset was highly pragmatic and strongly influenced by the need to create a large body of annotated material given limited human resources. The original design of the Treebank called for a level of syntactic analysis comparable to the skeletal analysis used by the Lancaster Treebank... no forced distinction between arguments and adjuncts. A skeletal syntactic context-free representation (parsing).” (p. 23)

We show two examples below of how a structure in PT could be better represented using our rule schemata:

(1) In exchange offers that expired Friday, holders of each $1,000 of notes will receive $250 face amount of Series A 7.5% senior secured convertible notes due Jan. 15, 1955, and 200 common shares.
As can be easily noticed, the sentence S begins with an Adjunct PP – an adjunct NP would have been treated the same way – which is then followed by the NP subject always at the same level. In our representation, the adjunct would have been positioned higher, under CP,
Also notice that we add an abstract COORD node that in this case is headed by punctuation conjunction AND, and in other cases will be added by punctuation marks.

An interesting question is constituted by the role played by Auxiliaries in case they are separated from the main verb by the NP Subject, has happens in English and Italian with Aux-To-Comp structures, and in general in German with Verb Second phenomena which are very frequent. NEGRA treebank has solved the problem by inserting a special label at S and VP level as shown here:

```plaintext
(S (S-MO
 (VMFIN-HD Mögen)
 (NP-SB
 (NN-NK Puristen)
 (NP-GR
 (PIDAT-NK aller)
 (NN-NK Musikbereiche) ))
 (ADV-MO auch)
 (VP-OC
 (NP-OA (ART-NK die)
 (NN-NK Nase) )
 (VVINF-HD rümpfen) ))
 (S, .)
 (NP-SB (ART-NK die)
 (NN-NK Zukunft)
 (NP-GR (ART-NK der)
 (NN-NK Musik) ))
 (VVFIN-HD liegt)
 (PP-MO (APPR-AC für)
 (PIDAT-NK viele)
 (ADJA-NK junge)
 (NN-NK Komponisten) )
```
Having a more specialized inventory of constituents was done also in view of facilitating further conversion projects into dependency structure which will be illustrated below. It also allows for easy searches and better specification of the structure searched. In particular, having a specialized node for tensed clauses, which is different from the one assigned to untensed ones, allows for better treatment of such constituent, which, as will be shown below, allows for some of its peculiar properties to be easily detected. Moreover, by assuming that the tensed verb complex – IBAR/IR_INFL - is the sentence head is in line with a number of theoretical frameworks and allows a much easier treatment in the LPCFG (Lexicalized Probabilistic Context-Free Grammars) scheme, where the head of the VP is also the head of S. Differently from what happens with PT, in VIT it doesn’t have to be extracted from a substructure because it’s already at S level: on the contrary, in PT the head could be the leaf of many different VP nodes depending on how many auxiliaries or modals precede the main lexical verb. In our case, for every further operation of transduction in dependency structure, the number of levels to keep under control is lower when the task of detecting Head-root and Head-dependent relations.

Adding a VP node that encompasses the Verbal complex and its complement was not a difficult task to carry out. We have then produced a script that enables the transformation of the entire VIT without a VP node into a version that conversely has it, but only in those cases where it is allowed by the grammar. In this way we successfully removed all those instances where the verbal group IBAR/IR_INFL is followed by linguistic material belonging to the S level, such as phrasal conjunctions, PP adjuncts or parenthetical structures. By doing this we were able to identify about 1000 clauses out of the total 16000 where the VP node hasn’t been added.

The following section describes work carried out to produce an algorithm for the automatic conversion of VIT, which uses traditionally bracketed syntactic constituency structures, into a linear word-based head-dependent representation enriched with grammatical relations, morphological features and lemmata. We are also still trying to produce a machine learning parsing algorithm that performs better than the current accuracy results which range below 70%.

3. FROM CONSTITUENT STRUCTURE TO HEAD-DEPENDENT FUNCTIONAL REPRESENTATION

3.1. Introduction

This section describes work carried out to produce an algorithm for the automatic conversion of VIT, which uses traditionally bracketed syntactic constituency structures, into a
linear word-based head-dependent representation enriched with grammatical relations, morphological features and lemmata.

Dependency syntactic representation consists of lexical items – the actual words - linked by binary asymmetric relations called dependencies. As Lucien Tesnière formulated it in (1959):

La phrase est un ensemble organisé dont les éléments constituant sont les mots. Tout mot qui fait parti d’une phrase cesse par lui-même d’être isolé comme dans le dictionnaire. Entre lui et ses voisins, l’esprit aperçoit des connexions, dont l’ensemble forme la charpente de la phrase. Les connexions structures établissent entre les mots des rapports de dépendance. Chaque connexion unit un terme supérieur à un terme inférieur. Le terme supérieur reçoit le nom de régissant. Le terme inférieur reçoit le nom de subordonné. Ans dans la phrase “Alfred parle” ... parle est le régissant et Alfred le subordonné.

If we try to compare types of information represented by the two theories we end with the following result:

- Phrase structure explicitly represent Phrases (nonterminal nodes); Structural categories (nonterminal labels). Possibly some functional categories (grammatical functions)
- Dependency structures explicitly represent Head-dependent relations (direct arcs); Functional categories (arc labels). Possibly some structural categories (POS).


In short, we can define dependency syntax to have to the following distinctive properties:

- It has direct encoding of predicate argument structure
- dependency structure is independent of word order
- for that reason it is suitable for free word order languages (Latin, Walpiri, etc.)
- however, it has limited expressivity
  - every projective dependency grammar has a strongly equivalent context-free grammar but not vice-versa
  - impossible to distinguish between phrase modification and head modification in unlabeled dependency structure

To obviate to some of the deficiencies of the dependency model, we designed our conversion algorithm so that all the needed linguistic information was supplied and present in the final representation as discussed in the section below.
3.2. The Conversion Algorithm

Input to the Algorithm for Head-Dependency Structures (hence AHDS) are the original sentence-based bracketed syntactic constituency structures, which are transformed into Head-Dependent column-based Functional representation by a pipeline of algorithms or rather scripts. These scripts produce a certain number of intermediate files containing the Tokenization, the Head Table, and the Clause Level Head-Dependency Table (hence CLHDT). The final output should be a file that contains the following items of linguistic information – for the word competitività/competitivity - in a column-based format:

<table>
<thead>
<tr>
<th>id_num.</th>
<th>word</th>
<th>POS</th>
<th>role</th>
<th>id_head</th>
<th>const.</th>
<th>lemma</th>
<th>[semantic/morphological features]</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>competitività</td>
<td>N(noun)POBJ</td>
<td>4</td>
<td>SN</td>
<td>competitività</td>
<td>[sems=invar, mfeats=f]</td>
<td></td>
</tr>
</tbody>
</table>

In the Tokenization file VIT is represented as a vertical list of words in the form of word-tag pairs. In addition, all multiword expressions have been relabeled into a set of “n” words preceding the head tagged as “MW”. The Head Table defines what category can be head to a given constituent and also the possible dependents in the same structure. The Head Table differentiates dependents from heads and has been used together with the Tokenization file to produce the CLHDT file. The current Tokenization includes information as to the constituent label the category belongs to. It also differentiates between simple POS labels and rich labels with extended linguistic (syntactic, semantic, morphological) information.

The fully converted file also includes Grammatical Relation labels. In order to produce this output, we had to relabel NP SUBJECTs, OBJECTs and OBLiques which are placed in a non canonical position. A similar question is related to the more general need to tell arguments and adjuncts apart for ditransitive and intransitive constructions. In Italian, prepositional phrases can occur quite freely before or after another argument/adjunct of the same predicate. Our strategy was at first that of marking as OBLique the first PP under COMPIN, and of course PPby under COMPPAS (more on this in a section below). But there is no possibility to mark ditransitive PP complement without subcategorization information, nor for that matter would PPs marked OBLiques be fully compliant without lexical information.

The solution to this problem was on the one side the use of our general semantically labeled Italian lexicon which contains 17000 verb entries together with a lexicon lookup algorithm, where each verb has been tagged with a specific subcategorization label and a further entry for prepositions subcategorized for. The use of this lexicon has allowed the automatic labelling of PP arguments in canonical position and reduced the task of distinguishing arguments from adjunct to the manual labeling of arguments in non canonical position.

On the other side, seen that nominal heads have been tagged with semantic labels – see the tagset in the appendix -, we proceeded at first by labeling possible adjuncts related to space and time. In case of verb of movement, where the subcategorization frames required it, and the preposition heading the PP allowed it, we marked the PP as argument. We also
relabeled as arguments all those PPs which were listed in the subcategorization frames of Ditransitives, again where the preposition allowed it.

We organized our work into a pipeline of intermediate steps which were incrementally turned to the full conversion task. In this way we also managed to check for consistency at different levels of representation.

3.2.1 Tagging and Multiwords

Checking consistency at the level of categories or PoS, was the work done with the first step, the tokenization of the VIT. At this level, we had to recover full consistency with multiwords as they had been encoded in the current version of VIT. We are aware of the fact that the lack of such an important annotation has caused serious problems in the PennTreebank where the same problem has been solved by assigning two different tags to the same word: e.g. the word “New” is tagged NNP and not JJ if it is followed by another NNP - “York” for example –, to convey the fact that “New” has to be interpreted as part of the proper name “New York”. However this has no justification from a semantic point of view seen that “New York” as a geographical proper name needs to use both words in order to access its referent not just one. Perhaps the original meaning of the word “New” in “New York” was that of adjective (hence JJ), seen that “York” in the new continent was “new” in relation to the British corresponding city name. But of course, all those words that encode their meaning in more than one wordform, will not be captured as such in the Penn treebank. For sure, the use of NNP for the non semantically independent portion of proper names will only contribute ambiguity to the same wordform that in other context will be tagged with their “natural” literal meaning. The conversion process starts with The script takes the parenthesized VIT as input file and creates a treebank version with indices without words and then the complete head table where every constituent is associated to its head with word id. To get that we differentiate nonterminal symbols from terminal one and assign an incremental index number to the latter.

Ex:
f-[ibar-[vin-restano],
compin-[sa-[ag-valide], sn-[art-le, n-multe, savv-[avv-già],
sv3-[ppas-irrogate], punt-',',
f2-[sp-[p-per, sn-[art-le, rel-quali]],
f-[ibar-[vin-pende],
compin-[sn-[art-il, n-giudizio,
sp-[php-davanti_al, sn-[npro-Tar]]]]], punto-.]

The same sentence rom VIT_without_words:
#ID=sent_01144  c1_f[c1_ibar-[0_vin], c1_compin-[c1_sa-[1_ag], c1_sn-[2_art, 3_n, c1_savv-[4_avv], c1_sv3-[5_ppas], 6_punt, c1_f2-[c1_sp-[7_p, c2_sn-[8_art, 9_rel]], c2_f-[c2_ibar-[10_vin], c2_compin-[c3_sn-[11_art, 12_n, c2_sp-[13_php, c4_sn-[14_npro]]]]], 15_punto],

#ID=sent_01144
 c1_f c1_ibar
As shown in Table 2, we eventually produced a verticalized version which contains PoS labels and their fully specified meaning, followed by constituent label in which the word was contained. In addition, PoS labels have been commented and whenever possible, morphological features have been added.

**Table 2.** TAGGING — verticalized text with categories and constituent labels

<table>
<thead>
<tr>
<th>#ID=sent_0002</th>
<th>art, article_definite_singular_masculine</th>
<th>sn</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>il, article_definite_singular_masculine</td>
<td>sn</td>
</tr>
<tr>
<td>1</td>
<td>raggiungimento</td>
<td>n, noun</td>
</tr>
<tr>
<td>2</td>
<td>e</td>
<td>cong, conjunction</td>
</tr>
<tr>
<td>3</td>
<td>il</td>
<td>art, article_definite_singular_masculine</td>
</tr>
<tr>
<td>4</td>
<td>mantenimento</td>
<td>n, noun</td>
</tr>
<tr>
<td>5</td>
<td>di</td>
<td>pd, preposition_di</td>
</tr>
<tr>
<td>6</td>
<td>posizioni</td>
<td>n, noun</td>
</tr>
<tr>
<td>7</td>
<td>competitive</td>
<td>ag, adjective</td>
</tr>
<tr>
<td>8</td>
<td>sono</td>
<td>vc, verb_copulative</td>
</tr>
<tr>
<td>9</td>
<td>sempre</td>
<td>avv, adverb</td>
</tr>
<tr>
<td>10</td>
<td>più</td>
<td>in, intensifier</td>
</tr>
<tr>
<td>11</td>
<td>il</td>
<td>art, article_definite_singular_masculine</td>
</tr>
<tr>
<td>12</td>
<td>risultato</td>
<td>n, noun</td>
</tr>
<tr>
<td>13</td>
<td>della</td>
<td>partd,preposition_di</td>
</tr>
<tr>
<td>14</td>
<td>interazione</td>
<td>n, noun</td>
</tr>
<tr>
<td>15</td>
<td>tra</td>
<td>p, preposition</td>
</tr>
<tr>
<td>16</td>
<td>le</td>
<td>art, article_definite_plural_feminine</td>
</tr>
<tr>
<td>17</td>
<td>azioni</td>
<td>n, noun</td>
</tr>
<tr>
<td>18</td>
<td>dei</td>
<td>partd,preposition_di</td>
</tr>
<tr>
<td>19</td>
<td>singoli</td>
<td>ag, adjective</td>
</tr>
<tr>
<td>20</td>
<td>soggetti</td>
<td>n, noun</td>
</tr>
</tbody>
</table>
3.3. Head-Constituent Relations

As a second step in our work we produced the table of Heads/constituents relations according to the rules formulated below. This step obliged us to look into every relation carefully so that no category was left without a function: it could either be a dependent or a head. No dangling categories would be allowed. We discovered that in the case of comparative constructions there was the need to separate the head of the phrase from the second term of comparison which did not have any specific constituent label. Working at constituent level we have been then obliged to introduce a new constituent label SC for comparative nominal structures, a label which is also used for Quantified related constructions. Rules are specified in the table below. The head extraction process was carried out following a list of head rules – which are better specified in an Appendix at the end of the chapter - according to Collins’ model for English. In particular, Direction specifies whether search starts from the right or from the left end of the child list dominated by the node in the Non-terminal column. Priority gives a priority ranking, with priority decreasing when moving down the list:

Table 3. Head-Constituent relations

<table>
<thead>
<tr>
<th>Constituent</th>
<th>Direction</th>
<th>Priority list</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUXTOC</td>
<td>Right</td>
<td>ause, auag, aueir, ausai, vsup</td>
</tr>
<tr>
<td>SN</td>
<td>Right</td>
<td>n, npro, nt, nh, nf, np, nc, sect, fw, relq, relin, relob, rel, pron, per_cent, int, abbr, num, deit, date, poss, agn, doll, sv2, f2, sa, coord</td>
</tr>
<tr>
<td>SAVV</td>
<td>Right</td>
<td>part, partd, avvl, avv, int, rel, coord, fw, neg, f2</td>
</tr>
<tr>
<td>SA</td>
<td>Right</td>
<td>ag, agn, abbr, dim, poss, neg, num, coord, ppre, ppas, fw, star, f2</td>
</tr>
<tr>
<td>IBAR</td>
<td>Right</td>
<td>vin, vii, vit, vgt, vgin, vgc, vppt, vppin, vppc, vcl, vcg, vc, vgprog, vgsf, virin, vt, vrt, vpr, vprin, vprogir, vprog, vprt, vsf, vsupir, vsup, vci, coord</td>
</tr>
<tr>
<td>IR_INFL</td>
<td>Right</td>
<td>vt, vin, vci, vii, vit, vgt, vgin, vgc, vppt, vppin, vppc, vcl, vcg, vgprog, vgsf, vgs, vrg, vrt, vpr, vprin, vprogir, vprog, vprt, vsf, vsupir, vsup, coord</td>
</tr>
<tr>
<td>SV5</td>
<td>Right</td>
<td>auag, aueg, vgt, vgsf, vgs, vrg, vrc, vrg, vcl, vpro, vit, vii, vci, vpp, vsf, vppp, coord</td>
</tr>
<tr>
<td>SV3</td>
<td>Right</td>
<td>ppre, ppas, vpr, vprin, vppin, vppc, vcl, coord, savv</td>
</tr>
<tr>
<td>SV2</td>
<td>Right</td>
<td>vprog, vit, vii, vci, vcl, vppin, vpppc, coord</td>
</tr>
<tr>
<td>SQ</td>
<td>Left</td>
<td>qc, q, ind, qd, num, in, coord, neg</td>
</tr>
<tr>
<td>FAC</td>
<td>Left</td>
<td>pk, f, cong, ir, infl, neg, coord</td>
</tr>
<tr>
<td>FS</td>
<td>Left</td>
<td>cosu, cosul, congl, coord</td>
</tr>
<tr>
<td>FC</td>
<td>Left</td>
<td>cong, cong, f, f3, ccom, ccongl, fc, punto, punt</td>
</tr>
<tr>
<td>FINT</td>
<td>Right</td>
<td>int, escl, puntint, sn, ir_infl, else any other element</td>
</tr>
<tr>
<td>F2</td>
<td>Left</td>
<td>rel, relq, relob, sp, spd, sn, spda, sq</td>
</tr>
<tr>
<td>COORD</td>
<td>Left</td>
<td>cong, cong, congl, punt, neg, par, ccom, fc, punto</td>
</tr>
<tr>
<td>SP</td>
<td>Left</td>
<td>p, part, php, coord</td>
</tr>
<tr>
<td>SC</td>
<td>Left</td>
<td>ccom, ccong, partd, pd</td>
</tr>
</tbody>
</table>
3.4. Clause Level Head-Dependency Table (hence CLHDT)

The third step in our work has been the creation of the CLHDT which contains a column where word numbers indicate the dependency or head relation, with the root of each clause bearing a distinctive dash, to indicate its role, as shown in Table 2. Rules for head-dependent relations are formulated below.

<table>
<thead>
<tr>
<th>#ID=sent_00002</th>
</tr>
</thead>
<tbody>
<tr>
<td>0  il</td>
</tr>
<tr>
<td>1  raggiungimento</td>
</tr>
<tr>
<td>2  e</td>
</tr>
<tr>
<td>3  il</td>
</tr>
<tr>
<td>4  mantenimento</td>
</tr>
<tr>
<td>5  di</td>
</tr>
<tr>
<td>6  posizioni</td>
</tr>
<tr>
<td>7  competitive</td>
</tr>
<tr>
<td>8  sono</td>
</tr>
<tr>
<td>9  sempre</td>
</tr>
<tr>
<td>10 più</td>
</tr>
<tr>
<td>11  il</td>
</tr>
<tr>
<td>12 risultato</td>
</tr>
<tr>
<td>13 della</td>
</tr>
<tr>
<td>14 interazione</td>
</tr>
<tr>
<td>15 tra</td>
</tr>
<tr>
<td>16  le</td>
</tr>
</tbody>
</table>
3.5 Rules for Head-Dependent Relations

At first we formulated a set of general rules as follows:
Heads with no constituent – or dangling heads - are unallowed.
Constituents with no heads are unallowed.

Coordinate structures are assigned an abstract head: they have conjunctions, punctuation or nil as their heads. Conjunctions are a thorny question to deal with: in dependency grammars they are not treated as heads. However, we take this case to represent a simple case of functional head government, very much in vein with a complementizer heading its complement clause. Punctuation plays an important role in parsing and in general it constitutes a prosodically related non-linguistic item. This is very clear in transcribed spoken corpora where all pauses had to be turned into appropriate punctuation, as we had to do in our work on Italian Spontaneous Speech Corpora (see Delmonte et al., 2007). This is why we assign a similar treatment to all “meaningful” punctuation marks. Punctuation marks like dash, quotations, parenthesis, angled brackets, which may introduce Parentheticals, Direct Speech, Reported Direct Speech are treated as functional heads. Other punctuation marks like commas introduced just to mark a pause and play no additional structural role are left interspersed in the text, as happens with PTB.

To better grasp the role of each constituent and its head in the conversion task, we divided up constituents according to their function and semantic import, into three main categories. As can be noticed, we specialized our non generic X-bar scheme into a set of constituent labels which were required to set apart functional types as well as structural and semantic types. For these reasons sentential constituent typologies differentiate between:

- Simple Declarative (F)
- Complex Declarative (CP)
- Subordinate Clause (FS)
- Coordinate Clause (FC)
- Complement Clause (FAC)
- Relative Clause (F2)
- Nonfinite tense Clause (SV2-SV3-SV5)
- Interrogative (FINT, CP_INT)
- Direct (Reported) Speech (DIRSP)
- Parenthetical, Appositive and Vocative (FP)
- Stylistically (literary and bureaucratic) marked utterances (TOPF)
- Fragments or non propositionally relatable utterances – lists, elliptic linguistic material, etc. (F3)

Sentence internally we used both Structural, Functional and Substantial constituent labels as shown in the table below:
### Table 5. Structural Constituents

<table>
<thead>
<tr>
<th>Label</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>sentence</td>
</tr>
<tr>
<td>F3</td>
<td>sentence fragment</td>
</tr>
<tr>
<td>CP</td>
<td>Dislocated/preposed constituents, adjunct constituents</td>
</tr>
<tr>
<td>CP_INT</td>
<td>Dislocated/preposed constituents, adjunct constituents</td>
</tr>
<tr>
<td>TOPF</td>
<td>Aux-to-comp constituents</td>
</tr>
<tr>
<td>COMPT</td>
<td>Complements governed by Transitive Verbs</td>
</tr>
<tr>
<td>COMPIN</td>
<td>Complements governed by Intransitive Verbs</td>
</tr>
<tr>
<td>COMPC</td>
<td>Complements governed by Copulative Verbs</td>
</tr>
<tr>
<td>COMPPAS</td>
<td>Complements governed by Passive Verbs</td>
</tr>
<tr>
<td>FP</td>
<td>Parenthetical, Apposition with punctuation – adjunct constituents</td>
</tr>
<tr>
<td>DIRSP</td>
<td>Direct speech with punctuation – any constituent</td>
</tr>
</tbody>
</table>

### Table 6. Lexical/Functional Constituents

<table>
<thead>
<tr>
<th>Label</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FAC</td>
<td>Complement sentence with/without complementizer (S’)</td>
</tr>
<tr>
<td>FC</td>
<td>Coordinated sentences with conjunction (S’)</td>
</tr>
<tr>
<td>FS</td>
<td>Subordinated sentence with subordinator (S’)</td>
</tr>
<tr>
<td>FINT</td>
<td>Interrogative sentence with/without interrogative pronoun (S’)</td>
</tr>
<tr>
<td>F2</td>
<td>Relative Clause with relative pronoun (S’)</td>
</tr>
<tr>
<td>COORD</td>
<td>Coordinate structure for constituents - heads with conjunction or punctuation (COORD)</td>
</tr>
<tr>
<td>SC</td>
<td>Comparative/Quantified Phrase with conjunction (QP)</td>
</tr>
<tr>
<td>SP</td>
<td>Prepositional Phrase with preposition (PP)</td>
</tr>
<tr>
<td>SQ</td>
<td>Quantified Phrase with quantifier (QP)</td>
</tr>
<tr>
<td>SPD</td>
<td>Prepositional Phrase with preposition DI (of) (PP)</td>
</tr>
<tr>
<td>SPDA</td>
<td>Prepositional Phrase with preposition DA (by) (PP)</td>
</tr>
</tbody>
</table>

All of these constituent labels have another constituent as their heads. Typically, F has the verbal complex as its head. As discussed above, syntactic subcategorization has been used as a label for a merely structural constituent that has the task to include all complements and adjuncts following the verb apart from higher level ones, which should be related at complex sentential level.

The constituent labels listed in the table below are headed by a grammatical function word with no semantic independent meaning.

These constituents have as head a lexical element which is not semantic, thus turning the constituent into a functional one. In this list also COORD is included which in some cases has been introduced as abstract constituent label with no head. Notice the functional distinction between different types of Prepositional Phrases, where those headed by preposition OF usually play the role of internal NP modifier, while the remaining ones usually are attached to the verb. We also differentiated the PP headed by preposition BY because in the majority of cases it would mark the presence of an Agent Adjunct phrase of a passive sentence.
Finally here below is the list of constituents which have a substantial semantic lexical element as head.

**Table 7. Substantial Constituents**

<table>
<thead>
<tr>
<th>SN</th>
<th>Nominal Phrase – Empty with F2 headed by Indefinite Relative Pronouns (NP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA</td>
<td>Adjectival Phrase (AP)</td>
</tr>
<tr>
<td>SAVV</td>
<td>Adverbial Phrase (ADVP)</td>
</tr>
<tr>
<td>AUXTOC</td>
<td>Verbal Group with tensed auxiliary (VP)</td>
</tr>
<tr>
<td>IBAR</td>
<td>Verbal Group with tensed verb (VP)</td>
</tr>
<tr>
<td>IR_INFL</td>
<td>Verbal Group with unreal verb (VP)</td>
</tr>
<tr>
<td>SV2</td>
<td>Infinitival Clause (VP)</td>
</tr>
<tr>
<td>SV3</td>
<td>Participial Clause (VP)</td>
</tr>
<tr>
<td>SV5</td>
<td>Gerundive Clause (VP)</td>
</tr>
</tbody>
</table>

SN has been used as structural marker and not as substantial constituent in two cases:
to indicate Subject Infinitivals in order to allow for easy conversion;
to head an empty Relative Clause with Indefinite Relative Pronouns.

**Table 8. Local Heads/Constituents Relations**

<table>
<thead>
<tr>
<th>#ID=sent_00002</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>F Sentence</td>
<td>IBAR Verbal group with tensed verb</td>
</tr>
<tr>
<td>COORD Coordinate structure for constituents</td>
<td>CONG Conjunction</td>
</tr>
<tr>
<td>SN Nominal phrase</td>
<td>N Noun</td>
</tr>
<tr>
<td>SN Nominal phrase</td>
<td>N Noun</td>
</tr>
<tr>
<td>SPD Prepositional phrase with preposition DI</td>
<td>PD Preposition_di</td>
</tr>
<tr>
<td>SN Nominal phrase</td>
<td>N Noun</td>
</tr>
<tr>
<td>SA Adjectival phrase</td>
<td>AG Adjective</td>
</tr>
<tr>
<td>IBAR Verbal group with tensed verb</td>
<td>VC Verb_copulative</td>
</tr>
<tr>
<td>COMPC Complements governed by Copulative Verbs</td>
<td>SAVV Adverbial phrase</td>
</tr>
<tr>
<td>SAVV Adverbial phrase</td>
<td>AVV Adverb</td>
</tr>
<tr>
<td>SN Nominal phrase</td>
<td>N Noun</td>
</tr>
<tr>
<td>SPD Prepositional phrase with preposition DI</td>
<td>PARTD Preposition_di_plus_article</td>
</tr>
</tbody>
</table>

In more detail, we run a series of routines in order to link every token to the head of the constituent it belongs to following a top-down procedure. First, we looked for the sentence root. The algorithm is organized as follows:
1. Find sentence:
   i. If it is a single sentence, return this
   Else:
   ii. If it is a complex sentence, find main sentence and return this

2. In sentence, find root:
   iii. If there is a tensed verb, return verb.
   Else:
   iv. If there is no tensed verb, return untensed verb.
   Else:
   iii. If there is no verb, return head of first NP.

Then, we used the head table to identify the heads of each local constituent and marked it in a file:

**Table 9. Local Heads with Clause Indices**

<table>
<thead>
<tr>
<th>#ID=sent_01144</th>
<th>c1_ibar</th>
<th>ROOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0_vin</td>
<td>c1_ibar</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>1_ag</td>
<td>c1_sa</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>2_art</td>
<td>c1_sn</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>3_n</td>
<td>c1_sn</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>4_avv</td>
<td>c1_savv</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>5_ppas</td>
<td>c1_sv3</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>6_punt</td>
<td>c1_sn</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>7_p</td>
<td>c1_sp</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>8_art</td>
<td>c2_sn</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>9_rel</td>
<td>c2_sn</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>10_vin</td>
<td>c2_ibar</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>11_art</td>
<td>c3_sn</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>12_n</td>
<td>c3_sn</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>13_punto</td>
<td>c1_f</td>
<td>LOCAL_HEAD</td>
</tr>
</tbody>
</table>

Then, we tried to link upwards all terminals without “LOCAL_HEAD” label to the head of the constituent they belong to by writing the id of the head in a third column:

**Table 10. Local Heads with Word Indices**

<table>
<thead>
<tr>
<th>#ID=sent_01144</th>
<th>c1_ibar</th>
<th>HEAD_MAIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>0_vin</td>
<td>c1_ibar</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>1_ag</td>
<td>c1_sa</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>2_art</td>
<td>c1_sn</td>
<td>3</td>
</tr>
<tr>
<td>3_n</td>
<td>c1_sn</td>
<td>LOCAL_HEAD</td>
</tr>
<tr>
<td>4_avv</td>
<td>c1_savv</td>
<td>LOCAL_HEAD</td>
</tr>
</tbody>
</table>
Step by step, we move upwards in the syntactic tree and trace the chain of substantial, lexical and structural constituents every head belongs to (the “head path”). In this way, we gather information to link e.g. an SN to a COMPIN and then the COMPIN to the IBAR, thus marking a relation between the head of the SN and the head of the IBAR. Besides, we add another column for the tokens:

We had to organize two special routines for the ‘cosu’ and the ‘cong’ labels. This was due to the fact that we could not follow the path upwards, but we had to look for the head inside the subordinate sentence they introduce.
4. Rules for Grammatical Relation Labels

The final step in the overall treebank full-fledged conversion is constituted by the assignment of Grammatical Relation labels/roles. In a language like English which imposes strict position for SUBJect NP and OBJect NP, the labeling is quite straightforward. The same applies for German which in addition has case marking to supplement for constituent scrambling, i.e. the possibility to scramble OBJect and Indirect OBJect in a specific syntactic area.

Differently from these two languages and other similar languages which constitute the majority of Western language typology, Italian is an almost “free word-order” language. In Italian, non canonical positions would indicate the presence of marked construction - which might be intonationally marked - as containing linguistic information which is “new”, “emphasized” or otherwise non thematic. Italian also allows freely the omission of SUBject pronouns whenever it is a discourse topic; it also has lexically empty non-semantic expletive SUBjects for impersonal constructions, weather verbs etc. This makes the automatic labeling of complements or arguments vs. adjuncts a difficult task to achieve, if tried directly from constituent labels without help from any external additional (lexical) information.

We thus started to relabel non-canonical SUBject and OBJect NPs, but the idea was that of relabeling all non-canonical arguments. However, we realized that we could operate a distinction between SUBject and complements in general, where the former can be regarded EXTernal arguments, receiving no specific information at syntactic level from the governing predicate to which they are related. On the contrary, arguments which are complements are strictly INTernal and are directly governed by the predicate, be it Verb, ADJective or Noun. Preposition constitute a case “per sé” in that they govern PPs which are exocentric constituents and are easily relatable to the NP head they govern. However, PPs need to be related to their governing predicate which may subcategorize for them or not according to Preposition type.

We thus produced rules for specific labeling and rules for default labeling. Default labeling is a generic less specific complement label which will undergo modification, if needed in the second phase. On the contrary, specific labeling will remain the same.

Here below are some of the rules for GRs labels.

4.1. Specific GR Labels

4.1.1. SN/SQ Manually Relabeled as S_Top

s_top (topicalized subject)>> topicalized subject positioned to the right and contained in any COMP type constituent

- SN/SQ specific

  It depends directly from – is contained directly under - F it is translated as SUBJ;
  
  It is contained in COMPT it is translated as OBJ even if it is preceded by some other constituent like SAVV;
  
  It is contained in COMPIN it is translated as ADJ;
  
  It is contained in COMPC it is translated as NCOMP;
It is contained in F2 (F2-[sn--...]), then it is translated as BINDER. The same applies to simple relative pronouns tagged REL/RELQ/RELIN/RELOB which are translated also directly as BINDER  
- SN/SQ default  
  GENERAL REMARK: any SN head must never be NT. In that case it will be translated by default ADJT (temporal adjunct);  
  In all other cases, SN/SQ will not be translated (in f3, etc).  
- SN/SQ remaining cases  
  It is in canonical preverbal position, i.e. and the head is different from NT, but does not fall under previous rules:  
    If it is separated from the verb by some constituent or by punctuation, it is positioned under CP, and is not the OBJECT, than it is labeled S_DIS (dislocated subject);  
    If it is an OBJECT or an NCOMP and is contained within CP and it is reinforced by the presence of a clitic in IBAR/IR_INFL – or not, it is treated as LDC (left dislocated complement).  
  It is in postverbal position, i.e. and the head is different from NT:  
    S_FOC (focalized subject)>> focalized subject positioned to the right and contained in any COMP type constituent.  

4.1.2. SPs specific and default rules  
All SPs can either be treated as arguments, adjuncts or modifiers. The subdivision we introduced makes it relatively easy to translate some argument and modifier position. However, as is the case with the previous constituents, finer distinctions require manual intervention or else the use of a subcategorized lexicon.  
We decided to treat argument PP as marked by the verb rather than by the preposition: in this way the PP becomes OBL or ARG_MOD thus indicating the Agent role of passivized verbs. Non argument PPs are translated as ADJs and have a semantically important preposition which consequently is the governor of a POBJ.  

4.1.2.1. SPDA specific and default  
It is contained in COMPPAS it is translated as ARG_MOD.  
It is contained in other COMP (T, IN, C) or under sentential level constituents, it is translated as ADJ. As said above, the constituent contained in the PP is translated into POBJ.  
It is contained in F2 (F2-[spda--...]), it is translated according to the rules specified above, but the SN/SQ is translated as BINDER and not as POBJ.  
- SPDA specific and default  
  It is contained in any non sentential constituent it is translated as MOD.  
  In all other cases, SPDA will not be translated (in f3-, etc).  
  If the SN head is tagged NP it will be translated POBJ-LOC; if it is headed by NT it will be translated ADJT.  

4.1.2.2. SPD specific and default  
It is contained in any COMP (T, IN, C) it is translated as OBL.  
It is contained in F2 (F2-[spd--...]), it is translated as OBL, but the SN/SQ is translated as BINDER and not as POBJ.
- **SPD default**

  It is contained in any non sentential constituent it is translated as *MOD*.

  It is positioned under sentential level constituents, it is translated as *ADJ*.

  As said above, the constituent contained in the PP is translated into *POBJ*. If the SN head is tagged NP it will be translated *POBJ-LOC*; if it is headed by NT it will be translated *ADJT*.

  In all other cases, SPD will not be translated (in f3-, etc).

  In more detail we carried out the following steps. First, we manually listed all s_dis (preposed subject under CP), s_foc (focalized object/subject in inverted position, no clitic), s_top (topicalized subject/object to the right, with clitic) and ldc (left dislocated complement, usually SA/SQ/SN/SP/SPD/SPDA).

  Second, we compared all verbs to an external verb list with verb valence and assigned the OBL role to the prepositions heading an oblique constituent. Then, we assigned a semantic role to the head of every constituent listed in column 1 of Table 1 according to the following rules:

### Table 12. Dependency Relations And Roles

<table>
<thead>
<tr>
<th>#ID=</th>
<th>sent_00002</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>il</td>
</tr>
<tr>
<td>1</td>
<td>raggiungimento</td>
</tr>
<tr>
<td>2</td>
<td>e</td>
</tr>
<tr>
<td>3</td>
<td>il</td>
</tr>
<tr>
<td>4</td>
<td>mantenimento</td>
</tr>
<tr>
<td>5</td>
<td>di</td>
</tr>
<tr>
<td>6</td>
<td>posizioni</td>
</tr>
<tr>
<td>7</td>
<td>competitive</td>
</tr>
<tr>
<td>8</td>
<td>sono</td>
</tr>
<tr>
<td>9</td>
<td>sempre</td>
</tr>
<tr>
<td>10</td>
<td>più</td>
</tr>
<tr>
<td>11</td>
<td>il</td>
</tr>
<tr>
<td>12</td>
<td>risultato</td>
</tr>
<tr>
<td>13</td>
<td>della</td>
</tr>
<tr>
<td>14</td>
<td>interazione</td>
</tr>
<tr>
<td>15</td>
<td>tra</td>
</tr>
<tr>
<td>16</td>
<td>le</td>
</tr>
<tr>
<td>17</td>
<td>azioni</td>
</tr>
<tr>
<td>18</td>
<td>dei</td>
</tr>
<tr>
<td>19</td>
<td>singoli</td>
</tr>
<tr>
<td>20</td>
<td>soggetti</td>
</tr>
</tbody>
</table>
Table 13. Role assignment rule table

<table>
<thead>
<tr>
<th>Constituent</th>
<th>Dependency</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCONG/ CONGF/ CONJL</td>
<td>Always</td>
<td>CONG</td>
</tr>
<tr>
<td>CCOM/ CONG</td>
<td>Governed by F</td>
<td>SUBJ</td>
</tr>
<tr>
<td>SN/SQ</td>
<td>Root of a sentence without a verb</td>
<td>SUBJ</td>
</tr>
<tr>
<td></td>
<td>Governed by COMPT</td>
<td>OBJ</td>
</tr>
<tr>
<td></td>
<td>Governed by COMPIN</td>
<td>ADJ</td>
</tr>
<tr>
<td></td>
<td>Governed by COMPC</td>
<td>NCOMP</td>
</tr>
<tr>
<td></td>
<td>Governed by F2</td>
<td>BINDER</td>
</tr>
<tr>
<td></td>
<td>Headed by NT</td>
<td>ADJT</td>
</tr>
<tr>
<td></td>
<td>Governed by SP/SPD/SPDA</td>
<td>-headed by NP (noun proper geographic)</td>
</tr>
<tr>
<td></td>
<td>- else</td>
<td>POBJ</td>
</tr>
<tr>
<td>SA</td>
<td>Governed by COMPC/COMPT/COMPPAS/COMPIN/F3/FP</td>
<td>ACOMP</td>
</tr>
<tr>
<td></td>
<td>- if verb is copulative</td>
<td>ADJ</td>
</tr>
<tr>
<td></td>
<td>- else</td>
<td>ADJ</td>
</tr>
<tr>
<td></td>
<td>Governed by F2</td>
<td>BINDER</td>
</tr>
<tr>
<td></td>
<td>Governed by SN where there is a SV2/SV3</td>
<td>ADJ</td>
</tr>
<tr>
<td></td>
<td>Governed by all other SNs/SAVV/SQ</td>
<td>MOD</td>
</tr>
<tr>
<td></td>
<td>Governed by SC</td>
<td>ADJ-COMP</td>
</tr>
<tr>
<td>SAVV</td>
<td>Governed by COMP</td>
<td>ADJ</td>
</tr>
<tr>
<td></td>
<td>Set between IBAR/IR_INFL and COMP</td>
<td>ADJT</td>
</tr>
<tr>
<td></td>
<td>Any other case</td>
<td>ADJM</td>
</tr>
<tr>
<td></td>
<td>Simple AVV/IN placed inside IBAR/IR_INFL</td>
<td>ADJV</td>
</tr>
<tr>
<td></td>
<td>Governed by F2</td>
<td>BINDER</td>
</tr>
<tr>
<td>REL/RELQ/ RELIN/RELOB</td>
<td>Governed by COMP</td>
<td>ADJ</td>
</tr>
<tr>
<td></td>
<td>Governed by all other SNs/SAVV/SQ</td>
<td>MOD</td>
</tr>
<tr>
<td>SPDA</td>
<td>Governed by COMPPAS</td>
<td>ARG_MOD</td>
</tr>
<tr>
<td></td>
<td>Governed by COMPC/COMPIN/COMPT/ CP/F/FAC/FC/FS/FP/F3/F2/FINT/DIRSP</td>
<td>ADJ</td>
</tr>
<tr>
<td></td>
<td>Governed by SN/SAVV/SQ</td>
<td>MOD</td>
</tr>
<tr>
<td>SPD</td>
<td>Governed by F2/COMPIN/COMPT/COMPC</td>
<td>OBL</td>
</tr>
<tr>
<td></td>
<td>Governed by CP/F/FAC/FC/FS/FP/F3/F2/FINT/DIRSP</td>
<td>ADJ</td>
</tr>
<tr>
<td></td>
<td>Governed by SN/SAVV/SQ</td>
<td>MOD</td>
</tr>
<tr>
<td>SP</td>
<td>Governed by COMPC</td>
<td>PCOMP</td>
</tr>
<tr>
<td></td>
<td>Governed by COMPIN/COMPT (not introduced by preposition “a”)</td>
<td>OBL</td>
</tr>
<tr>
<td></td>
<td>Governed by COMPIN/COMPT (introduced by preposition “a”)</td>
<td>IOBJ</td>
</tr>
</tbody>
</table>
And here below is the sentence we use to show the conversion process:

Table 14. Full conversion from phrase structure to dependency structure

<table>
<thead>
<tr>
<th>#ID=sent_01144</th>
<th>VIN(verb_intrans_tensed)</th>
<th>IBAR</th>
<th>-</th>
<th>CL(main)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 restano</td>
<td>VIN(verb_intrans_tensed)</td>
<td>IBAR</td>
<td>-</td>
<td>CL(main)</td>
</tr>
<tr>
<td>1 valide</td>
<td>AG(adjective)</td>
<td>ACOMP</td>
<td>0</td>
<td>SA</td>
</tr>
<tr>
<td>2 le</td>
<td>ART(article)</td>
<td>SN</td>
<td>3</td>
<td>SN</td>
</tr>
<tr>
<td>3 multe</td>
<td>N(noun)</td>
<td>S_TOP</td>
<td>0</td>
<td>SN</td>
</tr>
<tr>
<td>4 già</td>
<td>AVV(adverb)</td>
<td>ADJM</td>
<td>3</td>
<td>SAVV</td>
</tr>
<tr>
<td>5 irrogiate</td>
<td>PPAS(past_participle_absolute)MOD</td>
<td>3</td>
<td>SV3</td>
<td></td>
</tr>
<tr>
<td>6 ','</td>
<td>PUNT(sentence_internal)</td>
<td>SN</td>
<td>3</td>
<td>SN</td>
</tr>
<tr>
<td>7 per</td>
<td>P(preposition)</td>
<td>ADJ</td>
<td>3</td>
<td>SP</td>
</tr>
<tr>
<td>8 le</td>
<td>ART(article)</td>
<td>SN</td>
<td>9</td>
<td>SN</td>
</tr>
<tr>
<td>9 quali</td>
<td>REL(relative)</td>
<td>BINDER</td>
<td>7</td>
<td>SN</td>
</tr>
<tr>
<td>10 pende</td>
<td>VIN(verb_intrans_tensed)</td>
<td>IBAR</td>
<td>3</td>
<td>IBAR</td>
</tr>
<tr>
<td>11 il</td>
<td>ART(article)</td>
<td>SN</td>
<td>12</td>
<td>SN</td>
</tr>
<tr>
<td>12 giudizio</td>
<td>N(noun)</td>
<td>S_TOP</td>
<td>10</td>
<td>SN</td>
</tr>
<tr>
<td>13 davanti_al</td>
<td>PHP(preposition_locution)</td>
<td>MOD</td>
<td>12</td>
<td>SP</td>
</tr>
<tr>
<td>14 Tar</td>
<td>NPRO(noun_proper_institution)</td>
<td>POBJ</td>
<td>13</td>
<td>SN</td>
</tr>
<tr>
<td>15 .</td>
<td>PUNTO(sentence_final)</td>
<td>F</td>
<td>0</td>
<td>F</td>
</tr>
</tbody>
</table>

The treebank has 10,607 constituents with subject role, 3,423 of which have been manually assigned because they are in non-canonical position. Among the 7,184 SUBJ labels which were automatically identified, 46 constituents should have been assigned another
function, with a precision of 0.99. On the other hand, 218 constituents should bear a SUBJ label instead of their actual label, with a recall of 0.97

5. A QUANTITATIVE STUDY OF VIT

In this second part of the chapter, we introduce and discuss the quantitative data concerning the written portion of VIT and the constituents present in the 10,200 utterances of its Treebank; the data are displayed in Table 15 below and will be compared to those in PT and to those of the spoken part of VIT. Number of tokens indicated is the original one and does not take into account the number of additional tokens added because of split amalgams and cliticized verbs which amount to some 20k additional tokens.

In particular, we will focus on some structures that are interesting from a parsing point of view and are called “stylistic” structures.

Table 15. Quantitative Data of VIT Constituents

<table>
<thead>
<tr>
<th>Constituent type</th>
<th>Constituent Label</th>
<th>No. of occurrences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Utterances</td>
<td></td>
<td>10,200</td>
</tr>
<tr>
<td>Total of Tokens</td>
<td></td>
<td>256,365</td>
</tr>
<tr>
<td>Nominal Phrase</td>
<td>SN</td>
<td>69,580</td>
</tr>
<tr>
<td>Prepositional Phrase</td>
<td></td>
<td>21,393</td>
</tr>
<tr>
<td>Prepositional Phrase with prepositions DI/DA</td>
<td>SPD/SPDA</td>
<td>20,592</td>
</tr>
<tr>
<td>Adjectival Phrase</td>
<td>SA</td>
<td>21,205</td>
</tr>
<tr>
<td>Adverbial Phrase</td>
<td>SAVV</td>
<td>4571</td>
</tr>
<tr>
<td>Quantified Phrase</td>
<td>SQ</td>
<td>2523</td>
</tr>
<tr>
<td>Comparative Phrase</td>
<td>SC</td>
<td>520</td>
</tr>
<tr>
<td>Verbal Group with Tensed Verb</td>
<td>IBAR</td>
<td>13,404</td>
</tr>
<tr>
<td>Verbal Group with Unreal Verb</td>
<td>IR_INFL</td>
<td>2526</td>
</tr>
<tr>
<td>Abstract Coordinate Structure –with conjunction or punctuation as head</td>
<td>COORD</td>
<td>5703</td>
</tr>
<tr>
<td>Sentence</td>
<td>F</td>
<td>15,851</td>
</tr>
<tr>
<td>Subordinated Sentence with Subordinator</td>
<td>FS</td>
<td>1063</td>
</tr>
<tr>
<td>Coordinate Sentence with conjunction</td>
<td>FC</td>
<td>3718</td>
</tr>
<tr>
<td>Parenthetical, Apposition with Punctuation – Adjuncts Constituent</td>
<td>FP</td>
<td>4381</td>
</tr>
<tr>
<td>Interrogative Sentence with/without Interrogative Pronoun</td>
<td>FINT</td>
<td>585</td>
</tr>
<tr>
<td>Dislocated/preposed constituents, adjunct const.</td>
<td>CP</td>
<td>4906</td>
</tr>
<tr>
<td>Dislocated/preposed constituents, adjunct const.</td>
<td>CP_INT</td>
<td>203</td>
</tr>
<tr>
<td>Complement Sentence with/without Complementizer</td>
<td>FAC</td>
<td>956</td>
</tr>
<tr>
<td>Infinitival Clause/Participial Clause/Gerundive Clause</td>
<td>SV2/SV3/SV5</td>
<td>7568</td>
</tr>
<tr>
<td>Relative Clause with Relative Pronoun</td>
<td>F2</td>
<td>3425</td>
</tr>
<tr>
<td>Direct Speech with Punctuation – Any Constituent</td>
<td>DIRSP</td>
<td>1101</td>
</tr>
<tr>
<td>Sentence Fragment</td>
<td>F3</td>
<td>3552</td>
</tr>
</tbody>
</table>
We also listed subtypes of sentence F – i.e. Sentence with Null Subject –, in order to highlight its internal distribution. In a recent paper, Corazza et al. (2004) use a portion of VIT – 90,000 tokens produced in the SI-TAL project – to verify the possibility to train a statistic-probabilistic parser on the basis of procedures already experimented in English with PT by Collins and Bikel. Since the results they obtained are quite scarce (inferior to 70% accuracy), the authors wonder whether the poor performance might be due to intrinsic difficulties in the structure of the Italian language, to the different linguistic theory that has been adopted (cf. the lack of a VP node) or to the different tagset adopted, more detailed if compared to the one used in the PT.

According to what stated by Bikel regarding Collins’ work, still a landmark for the creation of probabilistic parsers, the work done for the creation of a language model is to be anticipated by an important phase of preprocessing. This means that in order to produce the language model one does not work on the raw data of a treebank, but on a version modified on purpose. Collin’s aim was to capture the biggest amount of regularities with the smallest number of parameters.

Probabilities are associated to lexicalized structural relations, i.e. structures where the head of the constituent to encode is present, that aim at helping to make decisions concerning the choice of arguments vs. adjuncts, of levels of attachment of a modifier and other similarly important matters otherwise difficult to capture when using only tags. For this purpose, it was necessary to intervene on the treebank by marking complements, sentences with null or inverse subject, and so on.

The preprocessing task accomplished by Corazza et al. is summarized here below and is actually restricted to the use of lemmas in place of word forms as head of lexicalized constituents:

“As a starting point, we considered Model 2 of Collins’ parser, as implemented by Dan Bikel, as its results on the WSJ are at the state-of-the-art. This model applies to lexicalized grammars approaches traditionally considered for probabilistic context-free grammars (PCFGs). Each parse tree is represented as the sequence of decisions corresponding to the head-centered, top-down derivation of the tree. Probabilities for each decision are conditioned on the lexical head. Adaptation of Collins’ parser to Italian included the identification of rules for finding lexical heads in ISST data, the selection of a lower threshold for unknown words (as the amount of available data is much lower), and the use of lemmas instead of word forms (useful because Italian has a richer morphology than English; their use provides a non negligible improvement). At least at the beginning, we did not aim to introduce language-dependent
adaptations. For this reason no tree transformation (analogous to the ones introduced by Collins for WSJ) has been applied to ISST.”(p.4)

From the verifications carried out using two different parsers, researchers have come to the conclusion that,

“These preliminary results... confirm that performance on Italian is substantially lower than on English. This result seems to suggest that the differences in performance between the English and Italian treebanks are independent of the adopted parser... our hypothesis is that the gap in performance between the two languages can be due to two different causes: intrinsic differences between the two languages or differences between the annotation policies adopted in the two treebanks.”(p.5-6)

From the experiment computed on the basis of the information theory it turns out that the difference in performance cannot be imputed to the amount of rules and therefore to the type of annotation introduced, but to the scarce predictability of their structural relations, as stated by the authors,

“First of all, it is interesting to note how the same coverage on rules results in the Italian corpus in a sensibly lower coverage on sentences (26.62% vs. 36.28%). This discrepancy suggests that missing rules are less concentrated in the same sentences, and that, in general, they tend to be less correlated the one with the other. This would not be contradicted by a lower entropy, as the entropy does not make any hypothesis on the correlation between rules, but only on the likelihood of the correct derivation. This could be a first aspect making the ISST task more difficult than the WSJ one. In fact, the choice of the rules to introduce at each step is easier if they are highly correlated with the ones already introduced.”(p. 9)

5.1. Regularity and Discontinuity in the Language and its Linguistic Representation

A number of conclusions can be safely drawn from what the researchers stated and from the results of their test. Intuitively one could assert that the better the structural regularity of a language or its representation is, the wider its reproducibility on a statistical basis; on the contrary, in a language containing many cases recurring only once, in general hapax, bis-, tris- legomena, a good statistical result of the model is less probable – this is called sparcity/sparseness. In linguistic terms the issue can be due to the division of grammar into core and periphery and this partition should be characterized in a quantitative manner. A statistical parser needs a great number of canonical structures belonging to the core grammar and it is not a case that in his procedure of creation of the model Collins deliberately introduces some corrections in the original treebank; that is, one has to accurately account for the structures which compose the core grammar, while the ones that constitute the periphery are amended ad hoc. Therefore, the malfunctioning of a statistical parser trained on a treebank must be related to the reference linguistic framework chosen by the annotators and hence to the reference language.

From the global quantitative data reported in Table 16. below, one can see that much more than half of the Italian sentences (9.800 in 19.099) do NOT have a subject lexically expressed in canonical position: this makes it very aleatory to locate the SN Subject. If we
compare this with PT we get a completely different picture. For instance, in PT there are 4647 sentences which have been classified with the node of topicalized structure (S-TPC) which includes argument preposing, sentences in direct reported speech, and so on. Moreover there are sentences with an inverse structure, classified as SINV, only 827 of which are also TPC: SINV sentences are 2587 and they all typically have the subject in post-verbal position.

While as for the work on PT it is sensible to correct the problem in the pre-processing phases as made by Collins and commented by Bikel, in our case this issue is less sensible and certainly more complicated. In fact, the SN subject can be realized in four different ways: it can be lexically omitted, it can be found with an inverted position in the COMP constituents where complements are placed, it can be found in dislocated position on the left or on the right of the sentence to which it is related, at CP level. In a preliminary annotation of such cases we counted a total of more than 3000 cases of lexically expressed subject in non-canonical position. Then there are about 6000 cases of omitted subject to be taken into account. All these sentences must be dealt with in different ways during the creation of the model.

Table 16. Comparison of non-canonical Structures in VIT and in PTB where we differentiate TU (total utterances) and TS (total simple sentences)

<table>
<thead>
<tr>
<th>Treebanks Vs. Non-canonical Structures (TU)</th>
<th>Non-canonical Structures with Non-Canonical Subject (TS)</th>
<th>Total (TU) Utterances</th>
<th>Total (TS) Simple Sentences</th>
<th>Total (TU) Complex Sentences</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIT</td>
<td>3719</td>
<td>9800</td>
<td>10,200</td>
<td>19,099</td>
</tr>
<tr>
<td>Percentage</td>
<td>27.43%</td>
<td>51.31%</td>
<td>63.75%</td>
<td>66.5%</td>
</tr>
<tr>
<td>PT</td>
<td>7234</td>
<td>2587</td>
<td>55,600</td>
<td>93,532</td>
</tr>
<tr>
<td>Percentage</td>
<td>13.01%</td>
<td>0.27%</td>
<td>59.44%</td>
<td>69.4%</td>
</tr>
</tbody>
</table>

If one considers that in PT there are 93532 sentence structures – identifiable with the reg_ex “(S (“ – 38600 of which are complex sentences, that is the 41% of all the “(S (“ – adding up all the cases of non-canonical SUBJECT sums up to a very low percentage, around 1%. On the contrary, in VIT the same phenomenon has a much higher percentage, over 27% in the case of non-canonical structures, and over 50% as to the omitted or unexpressed subject. We have also taken into consideration the annotation of complements in non-canonical position, and they have been listed in a table below.

Here below in Table 17. we show absolute values for all non-canonical structure we relabeled in VIT. Considering that the total number of canonical lexically expressed SUBJECTs is 7172, we can compute the number of non-canonical subjects as constituting 1/3 of all expressed SUBJECTs – total number of lexically expressed subjects corresponding to 10,100. We labeled as S_TOP subject NPs positioned to the right of the governing verb; as S_DIS those subject NP which are positioned to the left of the governing verb but are separated from it by a parenthetical or a heavy complement; S_FOC are typically subject in inverted postverbal position of presentational structures; finally LDC are all types of Left Dislocated Complements with or without a doubling clitic.
Table 17. Non-canonical Structures in VIT

<table>
<thead>
<tr>
<th>Constituent/ Distance</th>
<th>LDC (left dislocated complements)</th>
<th>S_DIS (dislocated subject)</th>
<th>S_TOP (topicalized subject)</th>
<th>S_FOC (Focalized Subject)</th>
<th>Total Non-Canonical</th>
</tr>
</thead>
<tbody>
<tr>
<td>251</td>
<td>1037</td>
<td>2165</td>
<td>266</td>
<td>3719</td>
<td></td>
</tr>
</tbody>
</table>

5.2. Discontinuous Modification

We also looked at all those structures that in Italian can have a modification or argument role in nominal structures, some of which can be found either before or after the head and some others can be dislocated in a distant position – separated by other intermediate constituents – in particular we here are referring to adjectival structures that can freely occur in post-nominal position even at a remarkable distance from the head – this structure is not possible in English. The relative data are reported in Table 18, where we counted the frequency in terms of the distance represented as the number of square brackets from the closest head, always a nominal head. In the case of a complement (argument) the constituent would be adjacent to the head otherwise it would be separated by a certain number of brackets that varies from 1 to 4.

A first reading gives us quite intuitive data as regard to the role that these constituents have in sentence structures: in particular the Ratio AM/AC tells us how many constituents there are of a certain type that have the function of argument/adjunct at sentence level compared to those that have the function of modifier in nominal structures. As one can see, PP-OF and RC are the two structures that more than others can be found in nominal structures, on the contrary the PP-BY rarely takes this role. Moreover, PP-OFs and APs differentiate themselves roughly from all the other constituents as to the argument position we named “Head Adjacent”. It is important to note how, in the case of APs, 8169 constituents (35%) are actually in a pre-head position although calculated as “Head Adjacent”. PP-OFs and RCs in post-head nominal position are respectively about 90% and 73% of all this type of constituent. As for RCs, 845 of them (25%) are of the non-restrictive type, i.e. they are separated by a comma: 98 of these are separated from the nominal head by a modifier, while the remaining ones present a PP type structure or a deeper embedded structure between the head and the RC whose dependency is thus difficult to identify. In the case of PPs and PP-BYs they are respectively 48% and 38% of the total amount of occurrences: in fact, in most cases these constituents have the function of complement/argument and of adjunct located in the COMP structure or at CP level. Lastly, 65% of the VPs are distributed as modifiers, while in the remaining cases they can occur as SUBJECT of copulative sentences or as argument in the COMP structure.

Table 18. Comparative Data of the position, in relation to the head, of the constituents that can be modifiers in nominal structures.

<table>
<thead>
<tr>
<th>Constituent/ Distance</th>
<th>SP</th>
<th>SPD</th>
<th>SPDA</th>
<th>SV</th>
<th>F2</th>
<th>SA</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>251</td>
<td>1037</td>
<td>2165</td>
<td>266</td>
<td>3719</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
It is easy to guess that the constituents with a higher structural ambiguity in Italian are those whose position in respect to the head is less predictable: respectively AP>VP>PP>RC>PP-BY>PP-OF. Besides, we must consider other elements that can lead to discontinuity or non-canonicity problems. In particular, the number of F3 or sentence fragments is quite high compared to the number of total utterances, 3552 (35%); the number of complex utterances is quite high – 6782 if compared to the total number (10.200) of utterances, therefore much higher than the 41% of PT.

6. AMBIGUITY AND DISCONTINUITY IN VIT

We will briefly present and discuss some of the most interesting structures contained in VIT as regards the two important question of ambiguity and discontinuity in Italian. The most ambiguous structures are constituted by Adjectival related structures. As already commented above, adjectives in Italian may be positioned in front or after the noun they modify almost freely for most lexical classes. Only few classes require to be in predicative position and a very small number of adjectives must be placed in front of the noun they modify, in attributive position. A count of the functional conversion of adjectival structures is presented here below:

<table>
<thead>
<tr>
<th>AM=all modifiers; AC=all constituents; HA=head adjacent.</th>
</tr>
</thead>
<tbody>
<tr>
<td>HA</td>
</tr>
<tr>
<td>Distance=1</td>
</tr>
<tr>
<td>Distance=2</td>
</tr>
<tr>
<td>Distance=3</td>
</tr>
<tr>
<td>Distance=4</td>
</tr>
<tr>
<td>Total All Mods (AM)</td>
</tr>
<tr>
<td>Ratio AM/AC</td>
</tr>
<tr>
<td>Totals Non HA</td>
</tr>
<tr>
<td>Ratio Non HA/AM</td>
</tr>
<tr>
<td>All Constituents</td>
</tr>
</tbody>
</table>

We will briefly present and discuss some of the most interesting structures contained in VIT as regards the two important question of ambiguity and discontinuity in Italian. The most ambiguous structures are constituted by Adjectival related structures. As already commented above, adjectives in Italian may be positioned in front or after the noun they modify almost freely for most lexical classes. Only few classes require to be in predicative position and a very small number of adjectives must be placed in front of the noun they modify, in attributive position. A count of the functional conversion of adjectival structures is presented here below:

1296 Complement APs (ACOMP), 18748 Modifiers (MOD), 324 Adjects (ADJ), 2001 COORDinate APs

6.1. Ambiguous Predicative SA

Postnominal adjectives constitute the most challenging type since they may be considered as either post or premodifiers of a following nominal head. Even though postnominal non-adjacent SA recur in a small number – only 5.34%, they need to be identified by the parser. In the examples below we try to show how this process requires knowledge of adjectival lexical
class besides feature matching. For every example taken from VIT we report the relevant portion of structure and a literal translation preceded by a slash.

(1) sn-[art-i, n-posti, 
spd-[partd-della, sn-[n-dotazione, sa-[ag-organica_aggiuntiva]]], 
sa-[ag-disponibili, sp-[p-a, 
/the posts of the pool organic additive available to

Syntactic ambiguity arises and agreement checking is not enough even though in some cases it may solve the attachment preferences for the predicative vs. the attributive position.

(2) sn-[sa-[ag-significativi], n-ritardi]], 
sn-[sa-[ag-profonde], n-trasformazioni], 
ibar-[vt-investono], 
/significative delays profound transformations affect

Adjectival structures may come in a row and modify different heads as in,

(3) sn-[art-il, n-totale, spd-[partd-dei, sn-[n-posti, 
spd-[partd-della, sn-[n-dotazione, sa-[ag-organica]]], ag-vacanti], 
sa-[ag-disponibili 
/the total of the posts of the pool organic additive vacant available

where “vacant” modifies the local head “posti”, as well as “disponibili” which however governs some complement. On the contrary, in the example below, “maggiori” is not attached to the a possible previous head “orientamenti”, but to a following one as the structure indicates,

(4) ibar-[vin-darebbe], 
compin-[sp-[in-anche, part-agli, sn-[n-orientamenti, 
spd-[pd-di, sn-[n-democrazia, sa-[ag-laica]]]],]sn-[sa-[ag-maggiori 
/would give also to the viewpoints of democracy laic main

6.2. Sentence Complement

Another interesting phenomenon SA is their ability to head Sentential Complements: however in case of copulative constructions they are nominalized SA, as in the following example.

(5) f-[sn-[art-il, 
   sa-[ag-bello]], 
   ibar-[vc-è], 
   compc-[fac-[pk-che] 
   /the beatiful is that

or this one
6.3. Tough Problems: Quantification

As can be noted, in both cases complement sentences have an implicit impersonal subject pronoun. Structures which constitute tough problems to represent are Quantified structure. They can be SQ or SC, i.e. Quantifier Phrase or Comparative Phrase. Let’s consider some example for both cases:

(7) sq-[in-molto, q-più, coord-[sa-[ag-efficace, punt-,, ag-controllabile, cong-e, ag-democratico]],
    sc-[ccom-di,
    f2-[sq-[relq-quanto],
    cp-[savv-[avv-oggi]],
    f-[ibar-[neg-non, vsup-poteva, vci-essere]
    /much more effective, controllable and democratic of how much today not be

where we see a case of coordinate SA governed by the quantifier operator PIU’. Here is another case,

(8) cp-[sq-[in-Più, sa-[ag-buono],
    sc-[ccom-di, savv-[avv-cosi]],
    f-[ibar-[neg-non, vsup-poteva, vci-essere]
    /more good than so not could be

(9) cp-[sc-[ccom-tanto, sq-[q-più],
    f-[ibar-[ve-sono], compc-[sa-[ag-lunghi]],
    sc-[ccom-tanto, sq-[q-maggiore],
    f-[ibar-[vc-e]],
    compc-[sn-[art-la, n-soddisfazione, sa-[ag-fina]]
    /much more are long much higher is the satisfaction final

where we see cases of comparative structures at sentence level. On the contrary the following example is a case of quantification in the form of a relative construction,

(10) cp-[
  cp-[sa-[ag-generali],

(6) f-[sn-[art-1,,
  sa-[ag-importante]],
  savv-[avv-ora],
  ibar-[vc-e],
  compc-[sv2-[vcl-aprirlo,
  compt-[clitac-lo],
  savv-[pd-di, avv-più]]]]
  /the important now is to open it of more
6.4. Fronted SPs in Participials

Another interesting construction present in Italian is the possibility to have fronted PP complements in Participials. This structure may cause ambiguity and problems of attachment, as shown in the examples below,

(11) sp-[p-in, sn-[n-base, sp-[part-al, sn-[n-puneggio, sv3-[sp-[p-ad, sn-[pron-essi]], ppas-attribuito, compin-[sp-[p-con, /on the basis of the scoring to them attributed with

where we see that “ad essi” could be regarded as a modifier of the previous noun “puneggio”, whereas it is a complement of “attribuito” which however follows rather precede it.

Another more complex case is constituted by,

(12) sp-[p-a, coord-[sn-[sa-[ag-singoli], n-plessi], cong-o, sn-[n-distretti], sv3-[sp-[p-in, sn-[pron-essi]], ppas-compresi, punto-.]]]]]]]]
/to single groups or districts in them comprised

As the examples show, this could also be computed as a case of proclitic, seen that only personal pronouns are allowed to be fronted and not nouns,

(13) spd-[partd-degli, sn-[n-importi, sv3-[sp-[p-ad, sn-[pron-essi]], ppre-spettanti]], cong-e, /of the amounts to them owed and

The structure is not only found in bureaucratic language but also in literary genre, as in,

(14) spd-[partd-della, sn-[n-cortesia, sv3-[sp-[p-in, sq-[q-più, pd-di, sn-[art-un_, n-occasione]]], vppt-dimostrata, compin-[coord-[sp-[p-a, sn-[pron-me]], /of the courtesy in more than one occasion demonstrated to me

6.5. Subject Inversion and Focus Fronted APs

Other non canonical structures are constituted by Subject Inversion, Focus Inverted APs, Left Clitic Dislocation with Resumptive pronoun.
A very frequent construction is constituted by the possibility to invert the Subject NP in postverbal position. This is usually linked to the presence of an Unaccusative verb governing the sentence.

(15) f-[ibar-[vc-diventa],
    compc-[savv-[avv-cosi],
    sa-[in-più, ag-acuta],
    sn-[art-la, n-contraddizione], sp-[p-tra
    /becomes so more acute the contradiction between
the same may happen with copulative verbs, where we see however that the subject is postponed after the open SA complement,

(16) f-[ibar-[vc-è],
    compc-[sa-[ag-peculiare,
    sp-[part-all, sn-[np-Italia///]],
    sn-[art-l, n-esistenza, spd-[pd-di
    /is peculiar to Italy the existence of

Here are some cases of Fronted APs,

(17) cp-[s_foc-[ag-Buono],
    f3-[sn-[cong-anche, art-l, n-andamento,
    spd-[partd-delle, sn-[n-vendite
    /good also the behaviour of the sales

(18) cp-[s_foc-[ag-Calmo],
    f3-[sn-[art-il, n-listino,
    spd-[partd-del, sn-[n-granoturco]
    /quite the price list of mais

(19) cp-[s_foc-[ag-buono], cong-f-invece,
    savv-[p-nel, avvl-complesso],
    f3-[sn-[art-il, n-resto
    /good instead on the whole the rest

All these structures are quite peculiar to the Italian language and also belong stylistically to a certain domain – financial news – and type of newspaper,

6.6. Hanging Topic and Left Clitic Dislocation

Italian allows to move locally in front of the utterance a portion of information which is somewhat resumed in the following sentence or may be left implicit and constitutes an elliptical material. Resumption usually takes place with a clitic pronoun. When the material fronted is not separated by a comma – a pause – it becomes a case of Left Clitic Dislocation.
The one below is a case of Hanging Topic

(21) cp-[sn-[sa-[ag-brutta], n-faccenda], punt-,,
    f-[sn-[art-i, n-sudditi],
    ibar-[clit-si, vt-ribellano, punto-]]
/bad story, the populace self rebel

(22) cp-[ldc-[art-una, n-decisione, sa-[ag-importante]],
    f-[sn-[nh-Ghitti],
    ibar-[clitac-l, ausa-ha, vppt-riservata],
/a decision important Ghitti it has reserved

(23) cp_int-[ldc-[art-il, n-concorso],
    f-[ibar-[clitac-l, ausa-ha, vppt-vinto],
    compt-[coord-[sn-[nh-Francesco],
    cong-o,
    sn-[nh-Giovanni]]],
    puntint-?
/the competition it has won Francesco or Giovanni ?

6.7. Aux-to-Comp Structures

Finally we will present and discuss some Aux-to-comp structures attested again both in bureaucratic and literary genres.

(24) cp-[f-[sn-[art-La, n-perdita],
    sp-[p-per, sn-[art-il, npro-Rolo]],
    ibar-[vcir-sarebbe],
    compt-[congf-però,
    spd-[pd-di, sn-[in-circa, num-'30', num-miliardi]]],
    topf-[auxtoc-[auag-avendo],
    f-[sn-[art-la, npro-Holding],
    sv3-[vppt-incassato,
    compt-[sn-[n-indennizzi,
    sp-[p-per, sn-[num-'28',
    num-miliardi]]]]], punto-]
/the loss for the Rolo would be then of about 30 billion having the Holding cashed payments for 28 billions
Here the gerundive auxiliary precedes the subject NP which in turn precedes the lexical verbal head in participial form. Below is a typical only Italian aux-to-comp structure,

(25) fc-[congf-e, punt-';',
    topf-[auxtoc-[clit-si, aueir-fosse],
    f-[sn-[pron-egli],
    sv3-[vppin-trasferito, cong-pure,
    compt-[sp-[part-nel,
    sn-[sa-[in-più, ag-remoto], n-continente]]]]]

/and, self would be he moved also in the more remote continent

This case and the following only belong to literary genre,

(26) cp-[sn-[topf-[auxtoc-[art-l, ausai-avere],
    f-[sn-[art-il, n-figlio],
    sv3-[vpp-t-abbandonato,
    compt-[sn-[art-il, n-mare],
    sp-[p-per, sn-[art-la, n-città]]]]]]],
    f-[ibar-[clitdat-le, ause-era, avv-sempre, vppt-seembrato]

/the have the son abandoned the sea for the city her was always seemed

Peculiarities in common with classical aux-to-comp is the presence of an auxiliary as structural indicator of the beginning of the construction. We introduced a new special constituent TOPF to include the auxiliary and the sentence where the lexical verbal head has to be searched in order to produce an adequate semantic interpretation.

6.8. (In)Direct Reported Speech

Now we will present some cases of sentential structures which are/should be marked by special punctuation to indicate reported Direct or Indirect speech. In all these sentences we have treated the governing sentence which usually is marked off by commas or by dashes, as a parenthetical. We will briefly comment 4 types of constructions as follows,
parenthetical inserted between SUBJ and IBAR
parenthetical inserted between material in CP and the F
free reported direct speech and then quoted direct speech
Direct speech is ascribed to an anonymous "someone" quoted anyhow

(27) dirsp-[par-",
    cp-[sp-[p-a, sn-[sa-[dim-questo], n-punto]],
    f-[sn-[art-la, n-data],
    par-".
    fp-[punt-", f-[ibar-[ausa-ha, vppt-detto],
    compt-[sn-[npro-d_, npro-Alema],
As can be noted, quotes individuate the portions of the utterance which is reported Direct Speech. The question is that the Subject NP “la data”/the date is separated from the Main Verb by the presence of the parenthetical governing clause. Below is a similar case,

(28) dirsp-[par-”,
    cp-[sp-[p-in, sn-[sa-[dim-questo], n-libro]],
    f-[sn-[nh-madre, npro-Teresa],
    fp-[par-, f-[ibar-[vt-spiegano],
    compt-[sp-[part-alla, sn-[npro-Mondadori]]], par-],
    ir_infl-[vcir-darà],
    /“in this book Mother Theresa -- explain at the Mondadori - will give

Punctuation does not help much in this example since the parenthetical is just introduced without indicating the end of the Reported Direct Speech.

6.9. Residual Problems: Relatives and Complement Clauses as Main Sentences

Italian allows freely to use Relative and Complement (with complementizer) Clauses as main clauses. This is due partly to the fact that Latin allowed it freely. But certainly it can be regarded a stylish way of organizing a text.

(29) cp-[f2-[rel-Che,
    cp-[fp-[punt-., f-[ibar-[vt-sostengono],
    compt-[sp-[part-alla, sn-[npro-Farnesina]]], punt-],
    f-[ibar-[neg-non, ausa-ha,
    sp-[p-per, avvl-niente],
    vppt-gradito],
    compt-[sn-[art-l, n-operazione, n-by_pass]],
    punto-.]]])

/That, maintain at the Farnesina, not has in no case liked the operation by_pass.

This example has the additional problem of the presence of a parenthetical sentence which should indicate the presence of an Indirect Reported Speech structure. Certainly hard to spot.

(30) cp-[f2-[rel-che,
    f-[ibar-[neg-non, vc-è],
    compc-[sn-[n-cifra, spda-[pda-da, sn-[in-poco]]]],
    fp-[punt-., fc-[ccong-cosi, conj]-come,
That not is figure by nothing, so as not is thing by every day to have one heir to the throne as guide turistic.

(31) cp-[fac-[pk-che, savv-[avv-poi]],
    f-[sn-[art-la, n-legge],
    ibar-[neg-non, virin-riesca],
    compin-[sv2-[pt-a, viin-funzionare]]],
    punt-,,
    f-[ibar-[vc-è],
    compc-[sn-[art-un, n-discorso, f2-[rel-che

That then the law not manages to work, is a matter that

7. PRELIMINARY EVALUATION

Here below we present preliminary data made available by Alberto Lavelli from IRST/ITC who implemented Bikel’s model and parser on VIT with the usual machine learning procedure of 10-Fold Cross Validation. The first table refers to the homogeneous subset of VIT composed of sentences from the financial newspaper called “Il Sole-24 Ore”.

Table 19a. Statistical parsing on VIT select homogenous subcorpus

| Number of sentences | = 3109 |
| Number of Error sentences | = 0 |
| Number of Skip sentences | = 0 |
| Number of Valid sentences | = 3109 |
| Bracketing Recall | = 67.47 |
| Bracketing Precision | = 66.48 |
| Complete match | = 6.66 |
| Average crossing | = 4.17 |
| No crossing | = 30.33 |
| 2 or less crossing | = 53.43 |
| Tagging accuracy | = 97.26 |

A slight improvement is obtained in case sentence length is limited to 40 tokens,

Table 19b. Statistical parsing on VIT select homogenous subcorpus with sentence length fixed to 40 tokens
Here below we present data related to the whole of VIT. As can be noticed, there is no remarkable difference between the this and the previous results.

Table 20a. Statistical parsing on complete VIT

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of sentences</td>
<td>10189</td>
</tr>
<tr>
<td>Number of Error sentences</td>
<td>12</td>
</tr>
<tr>
<td>Number of Skip sentences</td>
<td>0</td>
</tr>
<tr>
<td>Number of Valid sentences</td>
<td>10177</td>
</tr>
<tr>
<td>Bracketing Recall</td>
<td>68.61</td>
</tr>
<tr>
<td>Bracketing Precision</td>
<td>68.29</td>
</tr>
<tr>
<td>Complete match</td>
<td>8.70</td>
</tr>
<tr>
<td>Average crossing</td>
<td>3.25</td>
</tr>
<tr>
<td>No crossing</td>
<td>38.37</td>
</tr>
<tr>
<td>2 or less crossing</td>
<td>61.73</td>
</tr>
<tr>
<td>Tagging accuracy</td>
<td>96.65</td>
</tr>
</tbody>
</table>

Again a slight improvement is obtained when sentence length is reduced,

Table 20b. Statistical parsing on complete VIT with sentence length limitation

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentence length&lt;=40 --</td>
<td></td>
</tr>
<tr>
<td>Number of sentences</td>
<td>8519</td>
</tr>
<tr>
<td>Number of Error sentences</td>
<td>12</td>
</tr>
<tr>
<td>Number of Skip sentences</td>
<td>0</td>
</tr>
<tr>
<td>Number of Valid sentences</td>
<td>8507</td>
</tr>
<tr>
<td>Bracketing Recall</td>
<td>71.87</td>
</tr>
<tr>
<td>Bracketing Precision</td>
<td>71.58</td>
</tr>
<tr>
<td>Complete match</td>
<td>10.40</td>
</tr>
<tr>
<td>Average crossing</td>
<td>1.94</td>
</tr>
<tr>
<td>No crossing</td>
<td>45.47</td>
</tr>
<tr>
<td>2 or less crossing</td>
<td>71.72</td>
</tr>
<tr>
<td>Tagging accuracy</td>
<td>96.55</td>
</tr>
</tbody>
</table>
VIT differs greatly from PT not only for the amount of sentences and data, but also for the choice to include linguistic material of different nature: in VIT there are five different genres – news, bureaucratic genre, political genre, scientific genre, literary genre -, while in PT only one is represented. Hence the wider homogeneity we expect from PT and consequently the scarcer homogeneity in VIT.

The sparsity of VIT makes it difficult, if not impossible, to use it as a Language Model in the construction of probabilistic grammars for Italian. Therefore it is necessary to introduce corrective elements in order to enable the learning phase to distinguish sentences with different typologies (subject in canonical preverbal position, subject in non-canonical postverbal position, lexically unexpressed subject, left dislocated “hanging Topic” subject – separated from the verb by other complements (or composed of a “heavy” SN followed by punctuation) - right dislocated Hanging Topic subject – separated from the verb by other complements), etc. To this end, we implemented Bikel’s language model directly on VIT and from preliminary results we can safely say that the same poor performance is reconfirmed – around 70% accuracy. More experiments will be carried out to confirm the hypothesis in Corazza et al., even though from the data in our possession such a confirmation is very likely.

APPENDIX 1:
POS AND OTHER CATEGORY LABELS

Here below is the list of the extended tagset with 106 categories and augmented descriptions associated to them.

Tag Inventory for tags used in the examples and in our Italian tagger

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>abbr</td>
<td>abbreviation</td>
</tr>
<tr>
<td>ag</td>
<td>adjective</td>
</tr>
<tr>
<td>agn</td>
<td>nominal adjective</td>
</tr>
<tr>
<td>art</td>
<td>article def/indef</td>
</tr>
<tr>
<td>auag</td>
<td>auxiliary &quot;avere&quot; gerundive</td>
</tr>
<tr>
<td>auair</td>
<td>auxiliary &quot;avere&quot; mood irrealis</td>
</tr>
<tr>
<td>aueg</td>
<td>auxiliary &quot;essere&quot; gerundive</td>
</tr>
<tr>
<td>aueir</td>
<td>auxiliary &quot;essere&quot; mood irrealis</td>
</tr>
<tr>
<td>ausa</td>
<td>auxiliary &quot;avere&quot; tensed</td>
</tr>
<tr>
<td>ausai</td>
<td>auxiliary &quot;avere&quot; infinitival</td>
</tr>
<tr>
<td>ause</td>
<td>auxiliary &quot;essere&quot; tensed</td>
</tr>
<tr>
<td>ausei</td>
<td>auxiliary &quot;essere&quot; infinitival</td>
</tr>
<tr>
<td>ausep</td>
<td>auxiliary &quot;essere&quot; past participle</td>
</tr>
<tr>
<td>avv</td>
<td>adverb</td>
</tr>
<tr>
<td>avvl</td>
<td>adverbial locution</td>
</tr>
<tr>
<td>ccom</td>
<td>conjunction «like» NP comparative</td>
</tr>
<tr>
<td>clit</td>
<td>clitic pronoun</td>
</tr>
<tr>
<td>clitac</td>
<td>clitic pronoun accusative</td>
</tr>
<tr>
<td>clitabl</td>
<td>clitic pronoun ablative(locative)</td>
</tr>
<tr>
<td>clitdat</td>
<td>clitic pronoun dative</td>
</tr>
<tr>
<td>cong</td>
<td>conjunction</td>
</tr>
<tr>
<td>ccong</td>
<td>conjunction comparative</td>
</tr>
<tr>
<td>congf</td>
<td>conjunction sentential</td>
</tr>
<tr>
<td>conjl</td>
<td>conjunction locution</td>
</tr>
<tr>
<td>cosu</td>
<td>conjunction subordinate</td>
</tr>
<tr>
<td>cosul</td>
<td>conjunction subordinate locution</td>
</tr>
<tr>
<td>da _riempire</td>
<td>dots…</td>
</tr>
<tr>
<td>dim</td>
<td>demonstrative adjective</td>
</tr>
<tr>
<td>dirs</td>
<td>cue direct/indirect discourse ( : )</td>
</tr>
<tr>
<td>doll</td>
<td>dollar sign</td>
</tr>
<tr>
<td>equal</td>
<td>equal sign</td>
</tr>
<tr>
<td>exc</td>
<td>exclamation pronoun</td>
</tr>
<tr>
<td>fw</td>
<td>foreign word</td>
</tr>
<tr>
<td>in</td>
<td>intensifier</td>
</tr>
<tr>
<td>ind</td>
<td>adjective indef</td>
</tr>
<tr>
<td>int</td>
<td>interrogative pronoun</td>
</tr>
<tr>
<td>intj</td>
<td>interjection</td>
</tr>
<tr>
<td>n</td>
<td>noun</td>
</tr>
<tr>
<td>nc</td>
<td>noun colour</td>
</tr>
<tr>
<td>neg</td>
<td>negation</td>
</tr>
<tr>
<td>nf</td>
<td>noun factive</td>
</tr>
<tr>
<td>nh</td>
<td>noun human</td>
</tr>
<tr>
<td>np</td>
<td>noun proper geographic/institution</td>
</tr>
<tr>
<td>nt</td>
<td>noun temporal</td>
</tr>
<tr>
<td>npro</td>
<td>proper noun for upper case oow</td>
</tr>
<tr>
<td>num</td>
<td>numeral</td>
</tr>
</tbody>
</table>
APPENDIX 2:
HEAD/CONSTITUENT TRANSLATION RULES

The list below includes some of the 31 constituent labels, their heads, and their dependents. Heads are listed right below the constituent label, whereas dependents are listed separately.

**Head Rules For IBAR/IR_INFL**

IBAR/IR_INFL is headed by an inflected Verb: it may contain auxiliaries, modal, support verbs, progressive paraphrase verbs, passive paraphrase verbs, clitics, adverbials, negation, prepositional phrases, conjunctions. But also COORD, Pk. It may not be started or ended by adverbs or conjunctions: these should be moved outside.

**Head Rules for SA**

SA can be headed by Abbr, AG, DIM, Fw, POSS, NUM, AGN, PPAS, PPRE, COORD, Star, and may contain minor categories like IN, NEG. It may contain SAVV constituents.

Adjectives used as Noun Phrases, or pronominalized structures preceded by an article have the category AGN. Possessive pronouns behaving as Noun Phrases are computed as SN. SA can be followed by F2 with headless relative pronouns like “come”/like.
Head Rules for AUXTOC (OMP)

AUXTOC or Aux-to-Comp constituents can only contain non-head verbal categories like auxiliaries and other support verbs like modals, which have been dislocated to allow the positioning of the SUBJECT NP or other similar constituent in front of the main verb, which then has an untensed form. Possible heads are ause, auag, aueir, ausai, vsup. It may contain conjunctions.

Head Rules for TOPF

TOPF constituent is used to contain F clauses which are headed by AUXTOC.

Head Rules for SV5

SV5 constituent contains gerundives whose head verb is in simple gerundive form or complex gerundive form, this in turn is preceded by auxiliaries or other support verbs. The verb may be in enclitic form. It may be headed by: auag, aueg, vgt, vgsf, vgs, vgin, vcg, vcl, vprog, vit, viin, vci. Other possible tags contained in SV5 are COORD congf, cong, neg.

Head Rules for SV3

SV3 constituent contains participials whose head is in simple participle. It may contain preposed adverbial structures, and the verb may be expressed in enclitic form. It may be headed by: ppre, ppas, vpr, vprin, vppt, vppin, vppc, vcl. Other possible tags contained in SV3 are: ausai, ausei, cong, cong, punt, par, punto, neg, in.

Head Rules for SV2

SV2 constituent contains infinitivals whose head is in simple infinitive or complex infinitive, the verb being preceded by auxiliaries and other support verbs. It may be headed by: vprog, vit, viin, vci, vppt, vppin, vppc, vcl. Other possible tags contained in SV2 are: ausai, ausei, ausep, cong, in, neg, php, pt, vsf, vsup, vppc, vppt, vppin, par, punt.

Head Rules for SPDA

SPDA constituents contain prepositional phrases headed by DA/by-from preposition and its amalgamated versions. Its meaning depends on the complement label. It may be headed by: pda, partda. Other possible tags contained in SPDA are: p, par, dirs, neg, in.

Head Rules for SPD

SPD constituent contains prepositional phrases headed by DI/of preposition and its amalgamated. It may be headed by: pd, partd. Other possible tags contained in SPD are: par, p, dirs, neg, in.

Head Rules for SP
SP constituent contains prepositional phrases headed by any preposition DI-DA excluded. It may be headed by: p, part, php. Other possible tags contained in SP are: neg, p, in.

**Head Rules for SN**

SN constituent contains all possible noun categories. Noun Phrases can also be headless in cases listed at the end in which an infinitival is used as clausal SUBject, and when a relative pronoun is used in indefinite mode. It may be headed by: sect, fw, n, nt, nh, nf, np, nc, npro, per_cento, rel, relq, relin, relob, pron, int, abbr, num, deit, date, doll, poss, agn. Other possible tags contained in SN are: art, int, rel, num, par, punt.

**Head Rules for SQ**

SQ constituent contains quantifier phrases and may contain comparatives or other quantifier phrases as complements. It may be headed by: q, qd, qc, ind, neg, num. Other possible tags contained in SQ are: cong, in, num, q, par, partd, pd.

**Head Rules for SC**

SC constituent contains comparative phrases and quantifier phrases which are complement of SQ. We decided to differentiate SQ complements using this label because they are headed by the preposition DI and its amalgams which would then be ambiguous with SN modifiers adjuncts or arguments. It may be headed by: ccom, ccong. Other possible tags contained in SC are: neg.

**Head Rules for F2**

F2 nodes introduce relative clauses: in turn relative clauses should contain relative pronouns in their F2 node, or else they may in some limited cases be empty. The relative pronoun may be bare or it may be included in a constituent structure whose head may be a noun, or the relative pronoun itself. It may be headed by: rel, relq, relin.

**Head Rules for F**

F is headed by a tensed inflected verb, which can only be child of IBAR or IR_INFL, or a COORDinated version of these two.

**Head Rules For FAC**

FAC is introduced by PK or is empty in case the complementizer is left implicit. In some other cases it may be introduced by CONG. In presence of negation this appears as minor category.

**Head Rules for F3**

F3 nodes are used as structural markers to indicate the lack of a tensed full sentential or clause level structure. As such, they are not allowed to contain minor nor major categories: except for
intj, and neg. This is due to the fact that F3 also mark elliptical structures. All substantial constituents may be present together with COORD.

**Head Rules for Coord**

Coord nodes may be started by a constituent label, a substantial head or by a coordinating conjunction. Coord started by minor categories are banned. There are special coordination structure headed by negation and doubled. Also functional heads can be coordinated.

The remaining structural constituents may contain almost any other constituent, so they are just listed here below.

*CP, CP_INT, DIRSP, FP, COMPT, COMPIN, COMPC, COMPPAS*

**APPENDIX 3**

The main treebanks and related tools available nowadays are listed here below. They have been subdivided into 6 categories:

1. Feature Structure or Dependency Representation
2. Phrase Structure Representation
3. Spoken Transcribed and Discourse Treebanks
4. Tools
5. Other resources based on treebanks
6. Generic website for corpora

1. Feature Structure or Dependency Representation
   - Parc 700 Dependency Bank
     700 sentences from section 23 of the Upenn Wall Street Journal Treebank
     http://www2.parc.com/isl/groups/nltt/fsbank/
   - Prague Arabic Dependency Treebank
     100,000 words approximately
     http://ufal.mff.cuni.cz/padt
   - Prague Dependency Treebank: 1.5 million words
     3 layers of annotation: morphological, syntactic, tectogrammatical
     http://ufal.mff.cuni.cz/pdt2.0/
   - Danish Dependency Treebank
     5,500 trees approximately
     http://www.id.cbs.dk/~mtk/treebank/
   - Bosque, Floresta sintàctica
     10,000 trees approximately
     http://acdc.linguateca.pt/treebank/info_floresta_English.html
   - French Functional Treebank
     abeille@linguist.jussieu.fr
     http://www.llf.cnrs.fr/Gens/Abeille/French-Treebank-fr.php
   - LinGO Redwoods: 20,000 utterances (as for Fifth Growth)
     http://lingo.stanford.edu/redwoods/
     http://wiki.delp-h.in.net/moin/RedwoodsTop
2. Phrase Structure Representation
- Penn Treebank: 1 million words
dependency rules available for conversion
  http://www.cis.upenn.edu/~treebank/home.html
- ICE – International Corpus of English
  2 million words tagged and parsed
  http://www.ucl.ac.uk/english-usage/ice/
- BuTreckBank: 14,000 sentences
dependency version available
  http://www.butreckbank.org/
- Penn Chinese Treebank: 40,000 sentences
  http://www.cis.upenn.edu/~chinese/ctb.html
- Sinica Treebank: 61,000 sentences
  http://godel.iis.sinica.edu.tw/CKIP/engversion/treebank.htm
- Alpino Treebank for Dutch: 150,000 words
  http://www.let.rug.nl/vannoord/trees
- TIGER/NEGRA: 50,000/20,000 sentences
  Dependency version available
  http://www.ims.uni-struttgarter.de/.../negra-corpus/
- TueBa-D/Z: 22,000 sentences
  Dependency version available
  http://www.sfs.uni-tuebingen.de/en_tuebadz.shtml
- TueBa-J/S: 18,000 sentences
  Dependency version available
  http://www.sfs.uni-tuebingen.de/en_tuebajs.shtml
- Cast3LB: 18,000 sentences
  Dependency version available
  http://www.dlsi.ua.es/projectes/3lb/index_en.html
- SUSANNE
  Subset of Brown Corpus made up of 130,000 words
  http://www.grsampson.net/Resources.html

3. Spoken Transcribed and Discourse Treebanks
- Maptask: 128 dialogues turned into 2597 files
  there are similar efforts for other languages: Portuguese, Swedish, Dutch, Japanese
  http://www.hcrc.ed.ac.uk/maptask/
- PDTB – Penn Discourse TreeBank
  Penn Treebank turned into Discourse Relation Treebank
  http://www.seas.upenn.edu/~pdtb/
- DGB – Discourse GraphBank
  3110 sentences containing 8910 relations and clause pairs - 73Kwords
  http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogld=LDC2005T08
- RSTDT – Rhetorical Structure Theory Discourse Treebank
  Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski
  http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogld=LDC2002T07
- Talbanken05: 300,000 words
  http://w3.msi.vxu.se/~nivre/research/Talbanken05.html
  Dependency version available
- API-AVIP-IPAR - treebank
  60,000 words - 5000 dialogue turns
  http://www.cirass.unina.it/
- CLIPS corpus
100 hours of spoken dialogues - phonetically annotated
http://www.clips.unina.it/

- LIP corpus
  500.000 tokens, 57 hours of spoken dialogues, fully tagged and lemmatized
  http://languageserver.uni-graz.at/badip/badip/20_corpusLip.php
- CHRISTINE: 80500 words
  http://www.grsampson.net/RChristine.html

4. Tools
@annotate: http://www.coli.uni-saarland.de/projects/sfb378/negra-corpus/annotate.html
Ananas: http://www.atilf.fr/ananas/
BulTreebank Project: http://www.bultreebank.org
CLaRK System: http://www.bultreebank.org/clark/
DTAG Treebank Tool: http://www.isv.cbs.dk/~mbk/dtag/
KPML development environment:
  http://www.fb10.uni-bremen.de/anglistik/langpro/kpml/README.html
LTChunk Systemic Coder: http://www.ltg.ed.ac.uk/~mikheev/tagger_demo.html
Poliqarp: http://poliqarp.sourceforge.net/
RST Tool for annotating with RST relations by Marcu
  http://www.isi.edu/~marcu/software.html
SALSA: http://www.coli.uni-saarland.de/projects/salsa/
UAM Corpus Tool: http://www.wagsoft.com/CorpusTool/
SysFan tool: http://minerva.ling.mq.edu.au/
TnT tagger: http://www.coli.uni-saarland.de/~thorsten/tnt/
Wordfreak: http://wordfreak.sourceforge.net/
FreeLing: http://garraf.epsevg.upc.es/freeling/

5. Other resources based on treebanks
ACE project: PropBank/VerbNet/FrameNet
http://verbs.colorado.edu/~mpalmer/projects/ace.html
FrameNet
http://framenet.icsi.berkeley.edu/
NomBank
http://nlp.cs.nyu.edu/meyers/NomBank.html
NomLex
http://nlp.cs.nyu.edu/nomlex/index.html
ComLex
http://nlp.cs.nyu.edu/comlex/index.html

6. Generic website for corpora and other linguistic resources
http://billposer.org/Linguistics/Computation/Resources.html
http://nlp.stanford.edu/links/linguistics.html
http://www.bmanuel.org/
http://www.bmanuel.org/clr/clr2_tt.html
http://www.glue.umd.edu/~drg/clir/arabic.html
http://www.ims.uni-stuttgart.de/info/FTPServer.html
http://www.lai.com/mtct.html
http://www.aclweb.org/index.php?option=com_content&task=view&id=31&Itemid=31
Chapter 3

PARSING 1:  
THE PARTIAL PARSER FOR ARGUMENTS AND ADJUNCTS

The following three chapters are devoted to parsing and the two basic parsers that the system GETARUNS possesses: the Partial and the Deep Parser. We have organized the chapters in such a way as to present the most important topics usually related to the problem of parsing, while at the same time explaining the way in which our parsers have been devised. So the first chapter on parsing, this one, is devoted to the problem of how strict the parser is with regards to the need of respecting principles of grammaticality and semantic transparency. This topic has been tackled on the one side by making reference to Chomsky’s supposed interest in performance issues and the notion of Phase as the best structural interface between the parser and the theoretical needs. On the other side, we present a detailed criticism of what dependency parsers – regarded the best nonrecursive semantically transparent parsers - can or cannot do and why. We discuss difference between word-based and constituency based parser on the basis of real examples and mistakes usually made by dependency parsers.

The second chapter on parsing presents parsing strategies and how they can cope with ambiguous structures in a principled way. At the same time it presents the debate about the use of statistical approaches to parsing, which as we saw in the previous chapter, are heavily dependent on the treebank and the language chosen, and in the case of a language like Italian are unable to reach a sensible level of performance.

Eventually the third chapter, Chapter 5, is devoted to the Deep Parser and the way in which it copes with grammatically interesting structures. We will present the architecture, the code in Prolog and the most interesting structural f-structure representations. Each representation will be introduced by a brief theoretical introduction of LFG theoretical background and the way this has been implemented in the parser.

1. INTRODUCTION

In the last ten years or so Chomsky has been referring to the human parser as a gauge of the way in which syntactic processes are carried out in the mind of the language user. The parser has
also been referred to as a metaphor of grammatical processes underlying sentence comprehension as is being purported within the current Minimalist Theory (hence MT). At the same time, Hauser, Chomsky & Fitch 2002 (hence HCF02), have defined the human processor a specialist in the use of a computational device, that of recursivity, when packaging together lumps of reality in sentences or when expressing thoughts in spoken messages. It would seem that this is what makes humans different from any other living species using language as a form of communication.

This chapter is a computationally based analysis of linguistic facts which also relies on empirical data coming from real texts (Greval’s benchmark for parsing), and not from made-up sentences, usually argument-only and no adjuncts, appearing both in theoretically oriented work and in psycholinguistic work. Lots of constructions – like for instance coordination or parentheticals - which constitute hard to parse examples for all state-of-the-art parsers, don’t appear to deserve serious treatment in the theoretical and the psycholinguistic literature, where all analyses are carried out effortlessly. To this end, a substantial part of this chapter will be devoted to discussion of hard to parse sentences by showing parses produced by online Head-Dependent word-level parsers. The choice of sentences reflects partly what is usually deemed relevant in theoretical and psycholinguistic literature.

This interest for performance related notions can be regarded as an attempt on Chomsky’s side to support/endow MT with a psychological and computational basis, thus making MT a unified theory for language. However, a parser based on any linguistic theory that aims to realize such a goal, should account also for the determining factor in sentence comprehension, that is ambiguity. This in turn is the cause of Garden Path on the one side, and on the other it motivates the existence of parsing preferences in the human processor. So, in the last resort, a theory that aims at the explanation of performance facts should satisfy three different types of requirements: psycholinguistic plausibility, computational efficiency in implementation, coverage of grammatical principles and constraints. This is also what a parser should satisfy, but see below.

It is plausible to say that for the first time performance facts can be brought to bear on theoretical assumptions based on competence. In fact this is also what HCF02 seem to be aiming at, when they write:

“Recent work on FLN suggests the possibility that at least the narrow-syntactic component satisfies conditions of highly efficient computation to an extent previously unsuspected…. [T]he generative processes of the language system may provide a near-optimal solution that satisfies the interface conditions to FLB. Many of the details of language that are the traditional focus of linguistic study … may represent by-products of this solution, generated automatically by neural/computational constraints and the structure of FLB – components that lie outside of FLN.”

(Note that FLN stands for Faculty of Language Narrow, and FNB stands for Faculty of Language Broad). In particular, the notion we will be referring to, that of Phases, will be used in this chapter to test its validity in coping with the effects of ambiguity and Garden-Path related examples through the working of state-of-the-art Dependency Based parsers, in comparison with our linguistically based symbolic parser. We will try to show that the way in which Phases have been formulated is both too strong and too weak, even though their final status is under debate. Rather than Phases, we will be referring to a related and/or derivable notion, that of Argument and Adjunct as the semantically complete object of any step the parser should pursue in its process of analysis. Sentence level parsing requires in turn
a first clause-level preliminary structure (something close to pseudosyntax, as T&B call it),
which is then submitted to proper interpretation – and possible LF computation, before
interacting with higher than clause level for complex sentences, which can eventually license
the parser output for PF.

Constituency-based parsing models are lately starting to be supplanted by word-level
parsing models in the vein of Dependency-Based parsing. These parsers are organized in such
a way as to limit the scope of syntactic operations to adjacent head-dependent word pairs.
Recursivity is thus eliminated from the grammar and computational efficiency is usually
guaranteed. The same applies to bottom-up cascaded ATN-like parsers, which decompose the
task of syntactic structure building into a sequence of intermediate steps, with the goal of
avoiding recursivity as much as possible.

Parsers today are required to produce a semantically interpretable output for any text: in
order to achieve such a goal, Grammatical Relations need to be assigned to words in some
kind of hierarchical (constituent-based) representation, before some Logical Form can be
built. As will be argued, word-based head-dependent parsers are not good candidates for the
generation of such an output.

However, as will be shown below, both coverage, precision and recall don’t speak in
favour of such parsers which work strictly bottom-up and adopt parsing policies which are
not strictly left-to-right. In this respect, we believe that a parser should embody a
psycholinguistically viable model, i.e. it should work strictly left-to-right and be subject to
Garden Path effects. We also believe that by eliminating constituents from the parsing
process, and introducing the notion of Head-Dependent relations, grammaticality principles
may become harder to obey, as will be discussed in details below.

To reduce computational complexity Chomsky (1998/2000) introduced the idea of Phases
– units of syntactic computation - within the Minimalist Theory. The general idea is that of
limiting the burden of syntactic operations in order to ease workload or what must be retained
in active memory. This seems a counterargument to the fact that human language (actually
FLN) exhibits an intrinsic defining property that makes computation hard, that of recursivity
(HCF02). So it would seem that Chomsky’s concern in proposing Phases is doublefold: on
the one side it is motivated by performance related issues, on the other hand it is coupled to
theory internal motivations. In fact, we will only tackle performance questions and not
questions affecting the Minimalist Program, where the introduction of phases and their
theoretical status is still under debate (see Svenonius, 2001,2004; Legate, 2002,2003; Epstein,
2004). We would like to prove Phases to be a theory-independent principle governing the
functioning of the human parser which we will investigate from a psycholinguistic and a
computational point of view. The human parser is so efficient that it must obey some
principle-based criterion in coping with recursivity: Phases are Chomsky’s solution. Quoting
from Chomsky,

“A phase is a unit of syntactic computation that can be sent to the Spell-Out. Syntactic
computation proceeds in stages: a chunk of structure (a vP or a CP) is created and then
everything but its edge can be sent off to the interfaces.”

Phases are semantically complete constituents or “complete propositions”, which could
be independently given a Logical Form and a Phonetic Form. Carnie & Barss propose to
relativize the definition of phase to that of Argument which we subscribe fully here below: in their words,

“Each phase consists of an argument, the predicative element that introduces the argument (V or vP) and a functional category that represents a temporal operator which locates the predicate in time or space (Asp, T, etc.). Phases consist of:

a) a predicative element (v or V)
b) a single argument
c) a temporal operator that locates the predicate and argument in time and space (Asp or T)

To this definition we will add the need to regard arguments as semantically complete constituents with their adjuncts and modifiers, something which is asserted by Epstein (2004) and introduced in Chomsky 2004b, when they assume that the specification of a phase has “full argument structure”. In addition this could be derived where they assume that a partial LF could be produced. It goes without saying that in order to produce a partial or complete LF from syntactic chunks they need to be semantically interpretable: this includes semantic role assignment, being exempt from quantificational related problems like the presence of unbound variables. In this theory and in our parser, the LF is a flat structure with unscoped quantifiers.

In line with S.Pinker & R.Jackendoff's paper (hence P&J) produced as an answer to HCF02, we assume that lexical information is the most important static knowledge source in the processing of natural language. However, we also assume that all semantic information should be made to bear on the processing and this is only partially coincident with lexical information as stored in lexical forms. In particular, subcategorization, semantic roles and all other semantic compatibility evaluative mechanisms should be active while parsing each word of the input string. In addition, the Discourse Model and External Knowledge of the World should be tapped when needed to disambiguate ambiguous antecedents for pronominal binding: this in turn would be triggered by grammatical information like Grammatical Relations and Semantic Roles, and cannot be used independently from it.

In that perspective, we believe that a sound parsing strategy should opt for a parser that strives for an even higher than constituent semantically closer level: i.e. arguments and adjuncts, where mixed/hybrid strategies (bottom-up and top-down) are activated by the use of a strongly language-dependent lookahead mechanism.

We would like to speak in favour of such an approach, in which locality is sacrificed for a mixed or hybrid model, partially bottom-up, which uses both grammatical function driven information and lexical information from subcategorization frames to direct the choices of the argument vs adjunct building parsing process.

In a word-level parser, uncertainty will have to be overcome at two levels: constituency boundary and at clause boundary, in order to choose between argument and adjunct. A parser that looks for constituents has to cope only with the clause boundary uncertainty level and then decide which constituent can be interpreted as argument or else be left for adjunct interpretation. English has a number of peculiar structures - that we discuss in detail below - which don't allow a parser writer to be too self-confident: in particular, it allows complex clauses to be processed freely, both as complements and as adjuncts. This option, which is rather common in real texts, may cause the parser to treat a NP as the OBJect of a higher
matrix clause, rather than as SUBJect of a lower complement clause thus causing garden-path phenomena at psycholinguistic level and freezing the parser at a computational level.

At first, word-based dependency parsers have been proposed for free-word order languages, like Latin, Walpiri (see Covington, 1990). As will be shown below, word-level parsers suffer from the inability to parse non-canonical sentence structure, from long-distance dependency problems, from structural discontinuities related problems, as well as other language-specific aspects. We will present our mildly bottom-up parser for arguments and adjuncts by discussing a number of language-specific issues, as the one raised by compless complement clauses. This will be done by comparing our parser with two online dependency-based parsers: Connexor (see Tapanainen & Jarvinen) and Link Grammar (see Sleator, Temperley), which can be freely used at their respective websites,

http://www.connexor.com/demos/syntax_en.htm
http://www.link.cs.cmu.edu/link

So on the one side we endorse a position purported by linguistic theories like MT which require LF licencing of constituency at some level - and clause level Phases are here assumed as the only possible counterpart to LF; on the other side, we speak against MT as in a sense - at least some MT linguist would accept it - Dependency Parsing implements it because it assumes that parsing cannot just be bottom-up word level parsing, but some top-down guidance is needed. Besides, Dependency Parsing would not subscribe to introducing some Phase level form of licencing, and as we will see below, this is one of its major flaws. Furthermore, neither MT nor Dependency Parsing would accommodate a strictly semantic and lexicalist notion like "Argument-Adjunct" parsing together with a performance related notion like ambiguity and the accompanying effect of Garden Path, which is familiar in psycholinguistic literature. In addition, language dependent rules would suit best an MT-like approach with parameters driven options or any other linguistic theory which allows rule of Core Grammar to be set apart from rules of the Periphery. All this can be nicely obtained in our parser, which however, needs a lot of manual tuning.

The chapter is organized as follows: section 2. below presents the Partial parser and argues for its being compliant with Phases as proposed by Chomsky; in section 3. we present the most common mistakes produced by dependency-based parsers using sentences from the Greval Corpus and commenting the reasons for their failures; in section 4. we discuss why a constituency-based A-A parser is superior to word-level parsers and present the results from the evaluation of our parser. Finally we attempt some conclusions.

2. GETARUNS: A PARTIAL A-AS HYBRID PARSER

As commented above, to be Phase-compliant a parser needs to build up each constituent as a fully interpreted chunk with all its internal arguments and adjuncts if any. In this process, we know that there are two boundaries which need to be taken into account: the CP level and the Ibar level, where the finite verb is parsed. From a computational perspective we might paraphrase the concomitant contribution of the two Phases as follows:

v. parse all that comes before the finite verb and then reset your internal indices.
Our Partial parser has a manually-built grammar described in some detail in a section below, and is written in Prolog a programming language that provides freely for backtracking and has a variable passing mechanism useful to cope with a number of well-known grammatical problems like agreement (local and non-local) as well as Long-Distance Dependency. The parser is not a dependency parser in that it imposes constituent-based global restrictions on the way in which words can be parsed: only legal constituents are licenced by the parser.

We defined our parser “mildly bottom-up” because the structure building process cycles on a call that collects constituents until it decides that what it has parsed might be analysed as Argument or Adjunct. To do that it uses Grammatical Function calls that tell the parser where it is positioned within the current parse. This proceeds until finite verb is reached and the parse is continued with the additional help of Verb Guidance whichh provides subcategorization information.

The recursive procedure has access to calls collecting constituents that identify preverbal Arguments and Adjuncts including the Subject if any: when the finite verb is found the parser is hampered from accessing the same preverbal portion of the algorithm and switches to the second half of it where Object NPs, Clauses and other complements and adjuncts may be parsed. Punctuation marks are also collected during the process and are used to organize the list of arguments and adjuncts into tentative clauses.

The clause builder looks for two elements in the input list: the presence of the verb-complex and punctuation marks, starting from the idea that clauses must contain a finite verb complex: dangling constituents will be adjoined to their left adjacent clause, by the clause interpreter after failure while trying to interpret each clause separately.

The clause-level interpretation procedure interprets clauses on the basis of lexical properties of the governing verb: verbless clauses are not dealt with by the bottom-up parser, they are passed down – after failure - to the top-down parser which can licence such structures.

The final processor takes as input fully interpreted clauses which may be coordinate, subordinate, main clauses. These are adjoined together according to their respective position. Care is taken to account for Reported Speech complex sentences which require the Parenthetical Clause to become Main governing clause.

The parser is organized into nine layers as shown below in Figure 1.

Both topdown and bottomup parsers use the same rule modules. We opted to deal with Questions and Imperatives with the top-down parser rather than with the bottom-up one. Also sentences with Reported Direct speech are treated in that way due to the presence of inverted commas that must be interpreted accordingly. Non-clausal Subject sentences and extraposed That-clause fronted sentences are also computed top-down. The advantage of using fully top-down processing is that the clause-building stage is completely done away with: the parser posits the the clause type as a starting point, so that constituents are searched for and collected at the same level in which the parsing has started. However, this is only conceivable in such non-canonical structures as the ones discusssed below.

If the parser does not detect any of the previous structures, control is passed to the bottom-up/top-down parser, where the recursive call simulates the subdivision of structural levels in a grammar: all sentential fronted constituents are taken at the CP level and the IP (now TP) level is where the SUBject NP must be computed or else the SUBject NP may be in postverbal position with Locative Inversion structures, or again it might be a subjectless
coordinate clause. Then again a number of ADJuncts may be present between SUBJect and verb, such as adverbials and parentheticals. When this level is left, the parser is expecting a verb in the input string. This can be a finite verb complex with a number of internal constituents, but the first item must be definitely a verb. After the (complex) verb has been successfully built, the parser looks for complements: the search is restricted by lexical information. If a copulative verb has been taken, the constituent built will be labelled accordingly as XCOMP where X may be one of the lexical heads, P,N,A,Adv.

Figure 1. Pipeline of parsing modules for Partial hybrid (bottom-up-topdown) version of GETARUNS.

The clause-level parser simulates the sentence typology where we may have verbal clauses as SUBJect, Inverted postverbal NPs, fronted that-clauses, and also fully inverted OBJect NPs in preverbal position.

2.1. Parsing and Robust Techniques
As far as parsing is concerned, we purport the view that the implementation of sound parsing algorithm must go hand in hand with sound grammar construction. Extragrammaticalities can be better coped with within a solid linguistic framework rather than without it. Our parser is a rule-based deterministic parser in the sense that it uses a lookahead and a Well-Formed Substring Table to reduce backtracking. It also implements Finite State Automata in the task of tag disambiguation, and produces multiwords whenever lexical information allows it. In our parser we use a number of parsing strategies and graceful recovery procedures which follow a strictly parameterized approach to their definition and implementation. Recovery procedures are also used to cope with elliptical structures and uncommon orthographic and punctuation patterns. The parser is written in Prolog Horn clauses and uses Extraposition variables to compute Long-Distance Dependencies.

The grammar is equipped with a lexicon containing a list of fully specified inflected word forms where each entry is followed by its lemma and a list of morphological features, organized in the form of attribute-value pairs. However, morphological analysis for English has also been implemented and used for OOV words. The system uses a core fully specified lexicon, which contains approximately 10,000 most frequent entries of English. Subcategorization is derived from FrameNet, VerbNet, PropBank and NomBank and from a core of manually subcategorized lexicon for Nouns, Verbs and Adjectives of 6600, and the dictionary for morphological decomposition with 76,000 root entries. These are all consulted at runtime. Eventually the semantics from the WordNet and other sources derived from the web make up the encyclopaedia. In addition to that, there are all lexical forms provided by a fully revised version of COMLEX. In order to take into account phrasal and adverbial verbal compound forms, we also use lexical entries made available by UPenn and TAG encoding. Their grammatical verbal syntactic codes have then been adapted to our formalism and is used to generate an approximate subcategorization scheme with an approximate aspectual and semantic class associated to it. Semantic inherent features for Out of Vocabulary words, be they nouns, verbs, adjectives or adverbs, are provided by a fully revised version of WordNet – 270,000 lexical entries - in which we used 75 semantic classes similar to those provided by CoreLex.

All parser rules from lexicon to c-structure and to f-structure amount to 11,000 rules, including rules for tagging, chunking, constituency structure building both in the partial and the deep parser.

Our training corpus for the complete system is made up 200,000 words and is organized by a number of texts taken from different genres, portions of the UPenn WSJ corpus, test-suite for grammatical relations, narrative texts, and sentences taken from COMLEX manual.

Both shallow and deep parser will be presented and discussed in detail in the following chapters. Here we will only highlight some of its features while presenting Dependency Based Parsers and commenting their deficiencies.

2.2. Lookahead and FSA

One of the important differences distinguishing our parser from Dependency parsers we would like to highlight is the use of topdown lookahead based parsing strategies. As has been reported in the literature (see Tapanainen and Voutilainen 1994; Brants and Samuelsson 1995), English is a language with a high level of homography: readings per word are around 2
(i.e. each word can be assigned in average two different tags depending on the tagset). Lookahead in our system copes with most cases of ambiguity; however, we also had to introduce a disambiguating tool before the input string could be safely passed to the parser. Disambiguation is applied to the lookahead stack and is operated by means of Finite State Automata: we tested our tagging algorithm on PARC-700 corpus - a portion of UPenn treebank see King et al. - and got 97% accuracy. The reason why we use FSA is simply due to the fact that for some important categories, English has unambiguous tags which can be used as anchoring in the input string, to reduce ambiguity. I am now referring to the class of determiners which is used to tell apart words belonging to the ambiguity class [verb,noun], the most frequent in occurrence in English. Besides, all FSA may be augmented by tests related to linguistic properties needed for disambiguation. Some such tests are,

- check subcategorization frame for current word: this is used for [n,v], [a,n,v] ambiguity classes followed by a preposition, or followed by “that”
- check for gerundive verb form: this is used to check for –ing endings of words
- check for auxiliary and modals: this is used to disambiguate [n,v], [a,n,v] ambiguity classes when preceded by an auxiliary or a modal
- check for noun belonging to factive class – which may govern “that”-clauses: this is used to disambiguate “that” [a,c,r] ambiguity class when preceded by a governing noun
- check for verbs of saying: this is used to detect complex sentential complements. This is also used to disambiguate verbs preceded or followed by punctuation marks.

Here below is the list of tags used for English texts presented in our books.

### Tag Inventory for tags used in all examples and in our English tagger

<table>
<thead>
<tr>
<th>Tag</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>abbr</td>
<td>abbreviation</td>
</tr>
<tr>
<td>ag</td>
<td>adjective</td>
</tr>
<tr>
<td>arti</td>
<td>article indefinite</td>
</tr>
<tr>
<td>artd</td>
<td>article definite</td>
</tr>
<tr>
<td>auag</td>
<td>auxiliary have gerundive</td>
</tr>
<tr>
<td>aueg</td>
<td>auxiliary be gerundive</td>
</tr>
<tr>
<td>ausa</td>
<td>auxiliary have tensed</td>
</tr>
<tr>
<td>ausa3</td>
<td>auxiliary have 3rd pers.</td>
</tr>
<tr>
<td>ausa3n</td>
<td>auxiliary have 3rd pers. + negation</td>
</tr>
<tr>
<td>ausae3</td>
<td>auxiliary have/be 3rd pers.contraction</td>
</tr>
<tr>
<td>ausai</td>
<td>auxiliary have infinitival</td>
</tr>
<tr>
<td>ausan</td>
<td>auxiliary have + negation</td>
</tr>
<tr>
<td>ausd</td>
<td>auxiliary do tensed</td>
</tr>
<tr>
<td>ausd3</td>
<td>auxiliary do 3rd pers. tensed</td>
</tr>
<tr>
<td>ausd3n</td>
<td>auxiliary do 3rd pers. + negation</td>
</tr>
<tr>
<td>ausdn</td>
<td>auxiliary do tensed +negation</td>
</tr>
<tr>
<td>ause</td>
<td>auxiliary be tensed</td>
</tr>
<tr>
<td>ause3</td>
<td>auxiliary be 3rd pers.</td>
</tr>
<tr>
<td>ause3n</td>
<td>auxiliary be 3rd pers. + negation</td>
</tr>
<tr>
<td>ausei</td>
<td>auxiliary &quot;esser&quot; infinitival</td>
</tr>
<tr>
<td>ausen</td>
<td>auxiliary be + negation</td>
</tr>
<tr>
<td>ausep</td>
<td>auxiliary be past participle</td>
</tr>
<tr>
<td>avv</td>
<td>adverb</td>
</tr>
<tr>
<td>avvl</td>
<td>adverbial location</td>
</tr>
<tr>
<td>ccom</td>
<td>conjunction «like» NP comparative</td>
</tr>
<tr>
<td>cong</td>
<td>conjunction</td>
</tr>
<tr>
<td>ccong</td>
<td>conjunction subordinate comparative</td>
</tr>
<tr>
<td>conf</td>
<td>conjunction sentential</td>
</tr>
<tr>
<td>cosu</td>
<td>conjunction subordinate</td>
</tr>
<tr>
<td>cosul</td>
<td>conjunction subordinate locution</td>
</tr>
<tr>
<td>doll</td>
<td>dollar sign</td>
</tr>
<tr>
<td>dots</td>
<td>dots…</td>
</tr>
<tr>
<td>date</td>
<td>date number</td>
</tr>
<tr>
<td>deit</td>
<td>deictic pronoun</td>
</tr>
<tr>
<td>dim</td>
<td>demonstrative adjective</td>
</tr>
<tr>
<td>dirs</td>
<td>cue direct/indirect discourse</td>
</tr>
<tr>
<td>equal</td>
<td>equal sign</td>
</tr>
<tr>
<td>exc</td>
<td>exclamation pronoun</td>
</tr>
<tr>
<td>ext</td>
<td>there - expletive</td>
</tr>
<tr>
<td>fw</td>
<td>foreign word = noun</td>
</tr>
<tr>
<td>in</td>
<td>intensifier</td>
</tr>
<tr>
<td>int</td>
<td>interrogative pronoun</td>
</tr>
<tr>
<td>inta</td>
<td>interjection affirmative</td>
</tr>
<tr>
<td>intj</td>
<td>interjection</td>
</tr>
<tr>
<td>intjl</td>
<td>interjection locution</td>
</tr>
</tbody>
</table>
Dependency parsers dispense with the notion of constituency (even though they may recover it), on the basis of a local head-dependent relation and a few principles regarding crossing links. Some of these requirements are expressed by (Sleator & Temperly, 1988; 1993):

1. the local requirements of each word are satisfied
2. the links do not cross
3. the words form a connected graph

Covington (1990) defines these requirements as follow:

- Every word is the dependent of exactly one head (except the head of the sentence, which is not a dependent at all).
- Any number of words can be dependents of the same head.
- Thus, the dependency links form a tree whose root is the head of the sentence (normally taken to be the main verb).
Any type of dependency can be taken care of by such parser: head-dependent, but also governor-governed; and argument-subcategorizand.

Other a-theoretical positions are purported by Melchuk (see Melch’uk 1988: 12)

“All I intend to do is to suggest an artificial FORMAL LANGUAGE, or a formalism, for describing natural sentences at the syntactic level. ... By its logical nature, dependency formalism cannot be "proved" or "falsified." Leaving aside simple errors and inconsistencies, it can be evaluated solely in terms of expediency or naturalness, not in terms of truth or falsity. Dependency formalism is a tool proposed for representing linguistic reality, and, like any tool, it may not prove sufficiently useful, flexible or appropriate for the task for which it has been devised; but it cannot be true or false.”

Melchuk seems to disregard the relevance of having a linguistic theory to support the parsing process. Predictivity for this theory is a real problem as Covington admits even though some lookahead mechanism could be implemented. Discontinuity in natural language texts is what usually makes parsing hard to succeed. As will be shown, dependency parsing does not heal the problem.

The drawbacks of this approach will be discussed here below with real sentences which exemplify what in our opinion are inherent deficiencies of the theory – if any – behind the parser. The computational point of view will thus be used to highlight the fact that these parsers are insufficient in themselves to guarantee a linguistically valid performance in lack of a strong linguistic basis.

What papers in the field usually underline is the difficulty in treating with coordinate structures and long distance dependencies. However, from our survey, these parsers suffer also from what are regarded the usual classical problems, like attachment (PP but also (extraposed) Relative Clauses, Sentential Modifiers like Participials). The impression one gets from our survey is that the parsers don’t seem to obey any strong grammaticality constraint, that would require a SUBject or a COMPlement (when obligatory) to be always present in tensed clauses, a principle common to most linguistic theories. In addition, they don’t seem to be able to compute Complement of Predicates other than Verbs: so that Sentential Complements of deverbal nouns and Adjectives are usually wrongly computed. Other interesting cases are constituted by Locative Inversion sentences where no SUBject is computed.

As will be noticed, these parsers (and theory behind) don’t produce empty categories so that both syntactic and lexical control structures will be only partially instantiated. However, lexical control is encoded in both parsers we have analyzed.

3.1. Ambiguity and Phases

One of the important differences between our parser and Dependency Based Parsers we would like to highlight is the use of topdown lookahead-based parsing strategies. The use of strategies calls for psycholinguistically related disambiguation processes which are strictly bound to linguistic parameters. For instance, English is a language that freely allows compless (complementizer-less) sentential complement and relative clauses. Being the
sentence the highest recursive structural level, it is plausible that English speakers will adopt some strategy in order to avoid falling into a garden path - thus freezing the parser. Another peculiar feature of English regards the inherent ambiguity of Past Tense/Past Participle verb forms, exception made for irregular verbs which however only constitute a small subset in the verb lexicon of English, amounting in our case to some 20,000 entries. Seen that Reduced Relative Clauses are headed by past participle verb form, and that Participial Adjuncts may be attached to any NP head nouns quite consistently; and seen also that is very hard to apply strict subcategorization tests for participial SUBJect - or deep OBJect in case of passives - with good enough confidence we assume that such tests will only be performed in case the parser is at the complement level of the SUBject NP. The reason for this being that we need to prevent as much as possible failures at the Ibar level. Thus, the grammatical function information needs to be passed down into the NP complement level in order to be used for that purpose. More data are discussed in a following chapter.

Consider now the case of compless clauses to be taken as sentential complements of a transitive/intransitive governing verb. Most important is the treatment of compless (complementizer-less sentential complement and relative) clauses, where we combine bottom-up information – presence vs absence of tensed verb in the remaining input string (using lookahead) with top-down information – subcategorization requirements. These strategies are tuned on with language specific ambiguity classes, the well-known DO (Direct Object) vs Subject of SC (Sentence Complement) reading of an NP which cause a compless SC to be left without its Subject NP in case of parser’s error. As for compless relatives, we take the stance that this can only happen in non-subject position of the governing NP head. Thus, by using the Grammatical Function information that we pass down into the complement part of the NP rule we achieve such a discrimination.

As said above, all examples we will use are taken from the SUSAN corpus in its Greval Grammatical Relations representation. Only in a few cases we used sentences taken from online English newspapers (The Guardian), to better illustrate our point. Notice that numbers attached to examples are the ones taken from the suite.

We show at first two cases in which no strategy needs to be fired. In sentence (1) the NP following the governing verb SAID is in fact expressed as a personal pronoun in the nominative case, thus eliminating the problem of ambiguity,

(1) A Yale historian, writing a few years ago in The Yale Review, said We in New England have long since segregated our children.

And now consider the case in which a sentential complement is used in subject position, as in

(2) That any sort of duty was owed by his nation to other nations would have astonished a nineteenth century statesman.

No rule accessing sentential complement would be used to look for subject complement clauses which are only accessed at sentence level as a special case of sentence structure.

In order to take care of compless complement clauses the parser checks subcategorization frames, then in case the complementizer is missing, it activates a check for the semantic typology that allows the complementizer to be omitted, which coincides with bridge verbs or non-factive verbs. Now consider the case in which the complement clause follows the object (direct,indirect) NP, as in

(3) He told the committee the measure would merely provide means of enforcing the escheat law which has been on the books since Texas was a republic.
This strategy is organized in a similar way to checking for the attachment of a PP complement following a NP. The complementation information is turned into a “that” word in case of sentential complement, and into the whole set of prepositions subcategorized by the verb with PP complements. These words are used as Cues-set to prevent the NP from entering relative clause rules, or any PP headed with one of the prepositions listed in the Cues-set, after the head has been taken and the parser is in the complement block of rules. The Cues-set is passed as a list from the Ibar level down to the VP level and into the object NP if any. Now consider cases in which the parser has to choose between an object NP/Sentence complement SUBJECT in case the verb is compless as shown in:

(4) Mitchell decried the high rate of unemployment in the state and said the Meyner administration and the Republican controlled State Senate Must share the blame for this.

In sentence (4) the clause containing the governing predicate SAID is a coordinate subjectless structure, where – in our parser - an empty pronominal coinindex with the lexically expressed subject of the previous clause will be provided by the Clause-Builder before the interpretation procedure. Or this sentence where the complement is started by a comma, and a vocative,

(5) A man must be able to say, Father, I have sinned, or there is no hope for him.

As said above, we ascertain that the verb belongs to the semantic class of non-factive verbs and then look for a finite verb ahead before allowing the Sentence complement rule to be fired. Other similarly difficult cases that can be adequately treated in our parser are shown below,

(6) I told him what Liston had said and he said Liston was a double crosser and said anything he (Liston) got was through a keyhole.

where all the complement clauses headed by SAID and the following relative clause headed by ANYTHING are compless. In order to cope with such parentheticals as the one introduced in this sentence took us one month/man work to account for rules and lexical entries missing in the parser. Other such coverage problems were caused by sentences like 7 and 8 below,

(7) Wagner replied, Can't you just see the headline City Hooked for $172,000 ?
(8) Yet, I responded, could not similar things be said about the art of the past ?

Or an imperative/exhortative followed by a question as in,

(9) Take Augustine's doctrine of grace given and grace withheld: have you pondered the dramatic qualities in this theology ?

or a subordinate clause followed by a question as in,

(10) If he attaches little importance to personal liberty, why not make this known to the world ?

Hard sentences to parse were the following ones,

(11) Battalion Chief Stanton M. Gladden, 42, the central figure in a representation dispute between the fire fighters association and the teamsters union, suffered multiple fractures of both ankles.
(12) He bent down, a black cranelike figure, and put his mouth to the ground. where in 11 there are two long parentheticals fairly hard to process before the main verb comes; in 12 the appositive comes after the verb and not after head noun, the pronoun “he”.


In order to cope with new sentence structure and new cases of ambiguity, new lexical entries had to be added to account for special multiwords basically in the area of grammatical function words. We also added some new lexical multiwords which caused the FSA disambiguation problems.

3.2. Dependency Word-Level Parsers: A Close Look at their Output

We will now discuss cases of parsing failures by Dependency Parsers related to highly ambiguous structure as those presented here above, and comment them with reference to Phases.

The first sentence we discuss is taken from "The Guardian" online and is presented to show how complements complement clauses and relative clauses attached to SUBJECT NPs can cause parser to fail. This is what happens both to Connexor and to LGrammar in the output reported here below and taken from the web. As can be seen, LG mistakes the structure of the that-headed relative clause and also the main verb is wrongly structured. On the contrary Connexor, gets most of the structure right, and only mistakes attachment of the that-RC which is attached to CLERICS - a plural NP - but the RC has a singular IS as main verb, thus showing that there is no checking of subject agreement in relative clauses with the head governing Noun.

(13) Officials said only a meeting of an estimated 1,000 of Afghanistan's clerics that is expected to be held in the capital, Kabul, on Wednesday or Thursday, could declare jihad.

No complete linkages found.
+++Time 2.96 seconds (21.35 total)
Found 1250 linkages (59 with no P.P. violations) at null count 2
Linkage 1, cost vector = (UNUSED=2 DIS=6 AND=0 LEN=75)

(S (S (NP Officials))
  (VP said
    (ADVP only)
    (SBAR (S (NP (NP a meeting)
      (PP of
        (NP (NP an estimated 1,000)
          (PP of
            (NP (NP Afghanistan's)
              clerics)
              (SBAR (S (NP that)
                (VP is)))))))))
  (VP expected
    (SBAR (S (NP to
      (VP be
        (VP held
          (PP in
            (NP (NP the capital)))))))
)}
(NP Kabul)),
(PP on
(NP (NP Wednesday)
or
(NP Thursday))
)

(VP could))))))))

declare jihad.)

The representation produced by Connexor and reported here below is accompanied by POS tags and Functional labels which we don't comment here because it is fairly intuitive and in any case the information can be downloaded from their website - reported above.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform</th>
<th>Syntactic relation</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Officials</td>
<td>official</td>
<td>subj:&gt;2</td>
<td>@SUBJ %NH N NOM PL</td>
</tr>
<tr>
<td>2</td>
<td>said</td>
<td>say</td>
<td>main:&gt;0</td>
<td>@+FMAINV %VA V PAST</td>
</tr>
<tr>
<td>3</td>
<td>only</td>
<td>only</td>
<td>meta:&gt;5</td>
<td>@ADVL %EH ADV</td>
</tr>
<tr>
<td>4</td>
<td>a</td>
<td>a</td>
<td>det:&gt;5</td>
<td>@DN&gt; %&gt;N DET SG</td>
</tr>
<tr>
<td>5</td>
<td>meeting</td>
<td>meeting</td>
<td>subj:&gt;30</td>
<td>@SUBJ %NH N NOM SG</td>
</tr>
<tr>
<td>6</td>
<td>of</td>
<td>of</td>
<td>mod:&gt;5</td>
<td>@&lt;NOM-OF %N&lt; PREP</td>
</tr>
<tr>
<td>7</td>
<td>an</td>
<td>an</td>
<td>det:&gt;9</td>
<td>@DN&gt; %&gt;N DET SG</td>
</tr>
<tr>
<td>8</td>
<td>estimated</td>
<td>estimated</td>
<td>attr:&gt;9</td>
<td>@A&gt; %&gt;N A ABS</td>
</tr>
<tr>
<td>9</td>
<td>1,000</td>
<td>1,000</td>
<td>pcomp:&gt;6</td>
<td>@&lt;P %NH NUM CARD</td>
</tr>
<tr>
<td>10</td>
<td>of</td>
<td>of</td>
<td>mod:&gt;9</td>
<td>@&lt;NOM-OF %N&lt; PREP</td>
</tr>
<tr>
<td>11</td>
<td>Afghanist</td>
<td>afghanistan</td>
<td>attr:&gt;12</td>
<td>@A&gt; %&gt;N N GEN SG</td>
</tr>
<tr>
<td>12</td>
<td>clerics</td>
<td>cleric</td>
<td>pcomp:&gt;10</td>
<td>@&lt;P %NH N NOM PL</td>
</tr>
<tr>
<td>13</td>
<td>that</td>
<td>that</td>
<td>subj:&gt;14</td>
<td>@SUBJ %NH &lt;Rel&gt; PRON</td>
</tr>
<tr>
<td>14</td>
<td>is</td>
<td>be</td>
<td>v-ch:&gt;15</td>
<td>@+FAUXV %AUX V PRES SG</td>
</tr>
<tr>
<td>15</td>
<td>expected</td>
<td>expect</td>
<td>mod:&gt;12</td>
<td>@-FMAINV %VP EN</td>
</tr>
<tr>
<td>16</td>
<td>to</td>
<td>to</td>
<td>pm:&gt;17</td>
<td>@INFMARK&gt; %AUX INFMARK</td>
</tr>
<tr>
<td>17</td>
<td>be</td>
<td>be</td>
<td>v-ch:&gt;18</td>
<td>@-FAUXV %AUX V INF</td>
</tr>
<tr>
<td>18</td>
<td>held</td>
<td>hold</td>
<td>obj:&gt;15</td>
<td>@-FMAINV %VP EN</td>
</tr>
<tr>
<td>19</td>
<td>in</td>
<td>in</td>
<td>loc:&gt;18</td>
<td>@ADVL %EH PREP</td>
</tr>
<tr>
<td>20</td>
<td>the</td>
<td>the</td>
<td>det:&gt;21</td>
<td>@DN&gt; %&gt;N DET</td>
</tr>
<tr>
<td>21</td>
<td>capital</td>
<td>capital</td>
<td>pcomp:&gt;19</td>
<td>@&lt;P %NH N NOM SG</td>
</tr>
<tr>
<td>22</td>
<td>,</td>
<td>,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>Kabul</td>
<td>kabul</td>
<td>mod:&gt;21</td>
<td>@APP %NH N NOM SG</td>
</tr>
<tr>
<td>24</td>
<td>,</td>
<td>,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>on</td>
<td>on</td>
<td></td>
<td>@ADVL %EH PREP</td>
</tr>
<tr>
<td>26</td>
<td>Wednesday</td>
<td>wednesday</td>
<td>pcomp:&gt;25</td>
<td>@&lt;P %NH N NOM SG</td>
</tr>
<tr>
<td>27</td>
<td>or</td>
<td>or</td>
<td></td>
<td>@CC %CC CC</td>
</tr>
<tr>
<td>28</td>
<td>Thursday</td>
<td>thursday</td>
<td>comp:&gt;18</td>
<td>@PCOMPL-S %NH N NOM SG</td>
</tr>
<tr>
<td>29</td>
<td>,</td>
<td>,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>could</td>
<td>could</td>
<td>v-ch:&gt;31</td>
<td>@+FAUXV %AUX V AUXMOD</td>
</tr>
<tr>
<td>31</td>
<td>declare</td>
<td>declare</td>
<td>obj:&gt;2</td>
<td>@-FMAINV %VA V INF</td>
</tr>
<tr>
<td>32</td>
<td>jihad</td>
<td>jihad</td>
<td>obj:&gt;31</td>
<td>@OBJ %NH N NOM SG</td>
</tr>
</tbody>
</table>
This type of sentences constitutes a psycholinguistically hard to parser case for English speakers, given the fact that they have preferences for low attachment and in cases such as these will encur in the need to recompute the parsed structure. Another important issue that comes from the analysis of these sentences is the need to take into account Agreement between Nominal Head governing the Relative Clause and the internal Subject position which will be bound to the variable governed by the Head Noun. Phases in the MT theory must provide all semantically useful features to LF, and we know that Subject Agreement features have that property: gender, number and person are all relevant for subsequent interpretation requirements. Thus, the CP containing the Relative Clause will have to be checked for Agreement with the Head Noun after the structure has been fully built and licensed. This Agreement Check will need Verb features to be made visible at the Phase edge. In case of failure, as in this sentence, the lower NP "an estimated 1,000 Afghanistan's clerics" will be closed, and the Relative Clause will tentatively be attached to the higher NP "a meeting", this time with success, and the complete Subject Argument will now be available for semantic processing. Sentence analysis will successfully continue with the verb complex "could declare" which in turn will cause the parser to assume a NP as possible continuation (Verb Guidance) through lexical information coming from subcategorization frames.

Structures such as these are hard also because the Relative Clause is attached to the NP Subject, thus preventing the construction of an LF at clause level, before the attachment site is properly attained. Other similar problems arise from Relative Clauses containing reflexive pronouns, as reported in the psycholinguistic literature and discussed at length in Delmonte (2002a) and in a following chapter on pronominal binding, relatively to the couple of made-up examples below:

(14a) The doctor called in the son of the nice nurse who hurt himself.
(14b) The doctor called in the son of the nice nurse who hurt herself.

Because of low attachment preferences the (human) parser will at first attach the RC containing the reflexive pronoun to the local head, "the nurse", but then failure will ensue due to mismatch of features between controlled SUBJECT and reflexive bound pronounal, "himself". However this is detected only when LF is computed for the whole argument and Binding is attempted. Failure at LF will oblige the parser to backtrack and disrupt the previously assembled constituents, to try attachment higher up, which will succeed.

3.2.1. Coordinate Structures

As noted at the beginning, all parsers and in particular DepParsers are at pains in handling coordinate structures, being in a sense headless structures. Coordination is allowed between any two or more constituent of the same type, but ambiguity may cause the number of possible choices to become computationally very heavy to parse and a common source of failure.

It is hard to tell what Phases might be able to suggest as a principled solution to the problem of coordination seen that almost anything can undergo coordination. We assume semantic features to be the most important linguistic element to be taken into consideration when deciding attachment point of a coordinated constituent - apart from syntactic type of constituent. As will be shown below, however, this may not be always sufficient. In the
following example, the coordinate sentence headed by "was launched" is wrongly analyzed as a coordination of the participial headed by "supplied".

(17) The Dreadnought was built on designs supplied by the United States in 1959 and was launched last year.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The</td>
<td>the</td>
<td>det:&gt;2</td>
<td>@DN&gt; %&gt;N DET</td>
</tr>
<tr>
<td>2</td>
<td>Dreadnought</td>
<td>dreadnought</td>
<td>subj:&gt;3</td>
<td>@SUBJ %NH N NOM SG</td>
</tr>
<tr>
<td>3</td>
<td>was</td>
<td>be</td>
<td>v-ch:&gt;4</td>
<td>@+FAUXV %AUX V PAST</td>
</tr>
<tr>
<td>4</td>
<td>built</td>
<td>build</td>
<td>main:&gt;0</td>
<td>@-FMAINV %VP EN</td>
</tr>
<tr>
<td>5</td>
<td>on</td>
<td>on</td>
<td>phr:&gt;4</td>
<td>@ADVL %EH PREP</td>
</tr>
<tr>
<td>6</td>
<td>designs</td>
<td>design</td>
<td>pcomp:&gt;5</td>
<td>@&lt;P %NH N NOM PL</td>
</tr>
<tr>
<td>7</td>
<td>supplied</td>
<td>supply</td>
<td>mod:&gt;6</td>
<td>@-FMAINV %VP EN</td>
</tr>
<tr>
<td>8</td>
<td>by</td>
<td>by</td>
<td>ha:&gt;7</td>
<td>@ADVL %EH PREP</td>
</tr>
<tr>
<td>9</td>
<td>the</td>
<td>the</td>
<td>det:&gt;10</td>
<td>@DN&gt; %&gt;N DET</td>
</tr>
<tr>
<td>10</td>
<td>United States</td>
<td>united states</td>
<td>pcomp:&gt;8</td>
<td>@&lt;P %NH N NOM SG</td>
</tr>
<tr>
<td>11</td>
<td>in</td>
<td>in</td>
<td>tmp:&gt;7</td>
<td>@ADVL %EH PREP</td>
</tr>
<tr>
<td>12</td>
<td>1959</td>
<td>1959</td>
<td>pcomp:&gt;11</td>
<td>@&lt;P %NH NUM CARD</td>
</tr>
<tr>
<td><strong>13</strong></td>
<td>and</td>
<td>and</td>
<td>cc:&gt;7</td>
<td>@CC %CC CC</td>
</tr>
<tr>
<td>14</td>
<td>was</td>
<td>be</td>
<td>v-ch:&gt;15</td>
<td>@+FAUXV %AUX V PAST</td>
</tr>
<tr>
<td>15</td>
<td>launched</td>
<td>launch</td>
<td>cc:&gt;7</td>
<td>@-FMAINV %VP EN</td>
</tr>
<tr>
<td>16</td>
<td>last</td>
<td>last</td>
<td>det:&gt;17</td>
<td>@DN&gt; %&gt;N DET</td>
</tr>
<tr>
<td>17</td>
<td>year</td>
<td>year</td>
<td>tmp:&gt;15</td>
<td>@ADVL %EH N NOM SG</td>
</tr>
<tr>
<td>18</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>19</td>
<td>&lt;p&gt;</td>
<td>&lt;p&gt;</td>
<td>.</td>
<td>.</td>
</tr>
</tbody>
</table>
3.2.2. Wrong Tag Disambiguation

It is usually the case that tag disambiguation may affect badly the parsing process: tagging is in most cases performed before the parser starts and is independent of syntactic and lexical information. Mistakes can be disruptive as shown below. As discussed above, linguistic theories disregard problems related to performance like tagging disambiguation which even though coped by most parsers successfully never reaches 100% accuracy with any text.

‘That’ is wrongly tagged as conjunction rather than as deictic pronoun: as a result the verb VIEW takes a complement clause headed by THAT and HOW is analyzed as an adverbial governing VIEW with no relation to the real governor, the adjective CERTAIN. However this is a problem related to lack of Nominal and Adjectival lexical information.

(15) It is difficult to be certain how the administration views that $28 million, since the views of one leader may not be the same as the views of another one.

In sentence (16), SAYS is wrongly tagged as Noun and then turned into the SUBJECT of the title sentence with the verb SEES as main tensed verb. So no complement clause is computed. This sentence is taken from The Guardian online.

(16) Afghan Taliban Says Sees Holy War Against U.S.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Afghan</td>
<td>afghan</td>
<td>attr:&gt;2</td>
<td>@A&gt; %N A ABS</td>
</tr>
<tr>
<td>2</td>
<td>Taliban</td>
<td>taliban</td>
<td>attr:&gt;3</td>
<td>@A&gt; %N N NOM SG</td>
</tr>
<tr>
<td>**3</td>
<td>Says</td>
<td>say</td>
<td>subj:&gt;4</td>
<td>@SUBJ %NH N NOM PL</td>
</tr>
<tr>
<td>4</td>
<td>Sees</td>
<td>see</td>
<td>main:&gt;0</td>
<td>@+FMAINV %VA V PRES SG</td>
</tr>
<tr>
<td>5</td>
<td>Holy</td>
<td>holy</td>
<td>attr:&gt;6</td>
<td>@A&gt; %N A ABS</td>
</tr>
<tr>
<td>6</td>
<td>War</td>
<td>war</td>
<td>obj:&gt;4</td>
<td>@OBJ %NH N NOM SG</td>
</tr>
<tr>
<td>7</td>
<td>Against</td>
<td>against</td>
<td>ha:&gt;4</td>
<td>@ADVL %EH PREP</td>
</tr>
<tr>
<td>8</td>
<td>U.S</td>
<td>u.s</td>
<td>pcomp:&gt;7</td>
<td>@&lt;P %NH N NOM SG</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>&lt;s&gt;</td>
<td>&lt;s&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In sentence (36), ANTIQUE is wrongly tagged as Noun and then turned into the SUBJECT of the sentence with WORRY as main tensed verb

(36) The frequently postulated antique worry that the daylight hours might dwindle to complete darkness apparently gave rise to a ritual and celebration which we still recognize.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The</td>
<td>the</td>
<td>det:&gt;4</td>
<td>@DN&gt; %N DET</td>
</tr>
<tr>
<td>2</td>
<td>frequently</td>
<td>frequently</td>
<td>ad:&gt;3</td>
<td>@AD-A&gt; %E&gt; ADV</td>
</tr>
<tr>
<td>3</td>
<td>postulated</td>
<td>postulated</td>
<td>attr:&gt;4</td>
<td>@A&gt; %N A ABS</td>
</tr>
<tr>
<td>**4</td>
<td>antique</td>
<td>antique</td>
<td>subj:&gt;5</td>
<td>@SUBJ %NH N NOM SG</td>
</tr>
<tr>
<td>**5</td>
<td>worry</td>
<td>worry</td>
<td></td>
<td>@+FMAINV %VA V PRES</td>
</tr>
</tbody>
</table>
### 3.2.3. Wrong Attachment or Unconnected Constituents

This section shows cases of wrong attachment, which are quite common in constituency based parsing when no lexical information is made available. This is also the case with dependency parsers.

Dependency parsers are inherently lacking sufficient expressivity power for missing empty categories. In the following sentence, the participial headed by “HELD” is wrongly taken as a finite verb heading an OBJect Sentential Complement to SAY, then the Adverbial NP “Tuesday night” is wrongly computed as SUBJ of “BRING”

> (18) Robert Snodgrass, state GOP chairman, said a meeting held Tuesday night in Blue Ridge brought enthusiastic responses from the audience.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform relation</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TRENTON</td>
<td>trenton @A&gt; %&gt;%N N NOM SG</td>
<td>@NH %NH N NOM SG</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>@SUBJ %NH N NOM SG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Fifteen</td>
<td>fifteen</td>
<td>qn:&gt;4</td>
<td>@QN&gt; %&gt;%N NUM CARD</td>
</tr>
<tr>
<td>4</td>
<td>members</td>
<td>member</td>
<td></td>
<td>@NH %NH N NOM PL</td>
</tr>
</tbody>
</table>
The following sentence is a case of SUBJect NP with attached relative clause, one that usually creates conditions for failure in topdown parsers. However, the parser seems to lack grammaticality constraints and no heuristics to be able to decide on the subjecthood of one of the NPs available: so the result is a totally unconnected sentence.

(21) TRENTON - Fifteen members of the Republican State Committee who are retiring voluntarily this year were honored yesterday by their colleagues.

Sentence (26) is a case of wrong tag disambiguation which turns out as a concomitant case of wrong attachment of the governing verb CHANGED which is turned into a participial clause head. The new head verb of the sentence is SURFACE. However the real problem as in other cases listed in this chapter, is the presence of a SUBJect NP which has a restrictive relative clause.
(26) This region which had a higher temperature than the rest of the anode surface changed size and location continuously.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform relation</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This</td>
<td>this</td>
<td>det:&gt;2</td>
<td>@DN&gt; %&gt;N DET DEM SG</td>
</tr>
<tr>
<td>2</td>
<td>region</td>
<td>region</td>
<td>subj:&gt;14</td>
<td>@SUBJ %NH N NOM SG</td>
</tr>
<tr>
<td>3</td>
<td>which</td>
<td>which</td>
<td>subj:&gt;4</td>
<td>@SUBJ %NH &lt;Rel&gt; PRON</td>
</tr>
<tr>
<td>4</td>
<td>had</td>
<td>have</td>
<td>mod:&gt;2</td>
<td>@+FMAINV %VA V PAST</td>
</tr>
<tr>
<td>5</td>
<td>a</td>
<td>a</td>
<td>det:&gt;7</td>
<td>@DN&gt; %&gt;N DET SG</td>
</tr>
<tr>
<td>6</td>
<td>higher</td>
<td>high</td>
<td>attr:&gt;7</td>
<td>@A&gt; %&gt;N A CMP</td>
</tr>
<tr>
<td>7</td>
<td>tempera</td>
<td>temperature</td>
<td>obj:&gt;4</td>
<td>@OBJ %NH N NOM SG</td>
</tr>
<tr>
<td>8</td>
<td>than</td>
<td>than</td>
<td>mod:&gt;6</td>
<td>@ADV %EH PREP</td>
</tr>
<tr>
<td>9</td>
<td>the</td>
<td>the</td>
<td>det:&gt;10</td>
<td>@DN&gt; %&gt;N DET</td>
</tr>
<tr>
<td>10</td>
<td>rest</td>
<td>rest</td>
<td>pcomp:&gt;8</td>
<td>@&lt;P %NH N NOM SG</td>
</tr>
<tr>
<td>11</td>
<td>of</td>
<td>of</td>
<td>mod:&gt;10</td>
<td>@&lt;N-OF %N&lt; PREP</td>
</tr>
<tr>
<td>12</td>
<td>the</td>
<td>the</td>
<td>det:&gt;13</td>
<td>@DN&gt; %&gt;N DET</td>
</tr>
<tr>
<td>13</td>
<td>anode</td>
<td>anode</td>
<td>pcomp:&gt;11</td>
<td>@&lt;P %NH N NOM SG</td>
</tr>
<tr>
<td>14</td>
<td>surface</td>
<td>surface</td>
<td>main:&gt;0</td>
<td>@+FMAINV %VA V PRES</td>
</tr>
<tr>
<td>15</td>
<td>changed</td>
<td>changed</td>
<td>attr:&gt;16</td>
<td>@A&gt; %&gt;N A ABS</td>
</tr>
<tr>
<td>16</td>
<td>size</td>
<td>size</td>
<td>obj:&gt;14</td>
<td>@OBJ %NH N NOM SG</td>
</tr>
<tr>
<td>17</td>
<td>and</td>
<td>and</td>
<td></td>
<td>@CC %CC CC</td>
</tr>
<tr>
<td>18</td>
<td>location</td>
<td>location</td>
<td></td>
<td>@OBJ %NH N NOM SG</td>
</tr>
<tr>
<td>19</td>
<td>continuously</td>
<td>continuously</td>
<td>man:&gt;14</td>
<td>@ADV %EH ADV</td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Sentence (27) is another unconnected sentence which is a case of wrong tag disambiguation that affects badly the overall analysis: ASKED is interpreted as a Noun and becomes the SUBJECT of the Infinitival headed by ELABORATE. The main verb seems though SAY which should be interpreted as Sentential Complement to REPLY. It is interesting to note that the “White House” is not taken as a multiword but is decomposed into two separate entries.

(27) Asked to elaborate, Pierre Salinger, White House press secretary, replied, I would say it's got to go thru several more drafts.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform relation</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Asked</td>
<td>asked</td>
<td></td>
<td>@SUBJ %NH A ABS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>@NH %NH A</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The following sentence is a case of wrong attachment of BY gerundive to the wrong governing verb, the main verb SAY, rather than to the local verb HELP.

(28) Economically, Martin said, the United States could best help foreign countries by helping them help themselves.

Sentence (29) is a case of wrong attachment of a TO Indirect Object complement which has been displaced by the coordinate NP from its deverbal governing Head Noun REACTION. The Indirect Object is wrongly attached to Liberal Party.

(29) At the same time reaction among anti-organization Democratic leaders and in the Liberal party to the Mayor's reported plan was generally favorable.

The same argument applies to the following sentence where the wrong attachment of a TO Indirect Object complement to the adjacent Head Noun is again due to lack of subcategorization requirements imposed by the governing verb OWE.

(30) That any sort of duty was owed by his nation to other nations would have astonished a nineteenth century statesman.
3.3. Grammaticality Principles Constraints Violation

In the examples of this section we review cases of wrong clausal complement association, or no complement at all detected for copulative verb BE. In sentence (1), repeated here below, the complement clause of the verb SAY is left unconnected; the same applies to sentence (20) where the governing verb SAY is left without complement the clause governed by SPECULATE is left unconnected.

3.3.1. Complex Clauses

(1) A Yale historian, writing a few years ago in The Yale Review, said We in New England have long since segregated our children.

(20) She said, when she learned Jackie was heading home I'm just speculating, but I have to think Jack feels he's hurting Boston's chances.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>She</td>
<td>she</td>
<td>subj:&gt;2</td>
<td>@SUBJ %NH PRON PERS NOM SG3</td>
</tr>
<tr>
<td>2</td>
<td>said</td>
<td>say</td>
<td>main:&gt;0</td>
<td>@+FMAINV %VA V PAST</td>
</tr>
<tr>
<td>3</td>
<td>,</td>
<td>,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>when</td>
<td>when</td>
<td>tmp:&gt;6</td>
<td>@ADVL %EH ADV WH</td>
</tr>
<tr>
<td>5</td>
<td>she</td>
<td>she</td>
<td>subj:&gt;6</td>
<td>@SUBJ %NH PRON PERS NOM SG3</td>
</tr>
<tr>
<td>6</td>
<td>learned</td>
<td>learn</td>
<td>obj:&gt;2</td>
<td>@+FMAINV %VA V PAST</td>
</tr>
<tr>
<td>7</td>
<td>Jackie</td>
<td>jackie</td>
<td>@OBJ %NH N NOM SG</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>was</td>
<td>be</td>
<td>v-ch:&gt;9</td>
<td>@+FAUXV %AUX V PAST</td>
</tr>
<tr>
<td>9</td>
<td>heading</td>
<td>header</td>
<td></td>
<td>@-FMAINV %VA ING</td>
</tr>
<tr>
<td>10</td>
<td>home</td>
<td>home</td>
<td>obj:&gt;9</td>
<td>@OBJ %NH N NOM SG</td>
</tr>
<tr>
<td>11</td>
<td>I</td>
<td>i</td>
<td>subj:&gt;12</td>
<td>@SUBJ %NH PRON PERS NOM SG1</td>
</tr>
<tr>
<td>12</td>
<td>'m</td>
<td>be</td>
<td>v-ch:&gt;14</td>
<td>@+FAUXV %AUX V PRES SG1</td>
</tr>
<tr>
<td>13</td>
<td>just</td>
<td>just</td>
<td>meta:&gt;14</td>
<td>@ADVL %EH ADV</td>
</tr>
<tr>
<td>14</td>
<td>speculatin</td>
<td>speculate</td>
<td></td>
<td>@-FMAINV %VA ING</td>
</tr>
<tr>
<td>15</td>
<td>,</td>
<td>,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>but</td>
<td>but</td>
<td>cc:&gt;14</td>
<td>@CC %CC CC</td>
</tr>
<tr>
<td>17</td>
<td>I</td>
<td>i</td>
<td>subj:&gt;18</td>
<td>@SUBJ %NH PRON PERS NOM SG1</td>
</tr>
<tr>
<td>18</td>
<td>have</td>
<td>have</td>
<td>cc:&gt;14</td>
<td>@+FMAINV %VA V PRES</td>
</tr>
</tbody>
</table>
| 19 | to   | to       | pm:>20    | @INFMARK> %AUX INFMARK>
| 20 | think | think    | obj:>18   | @-FMAINV %VA V INF      |
| 21 | Jack | jack     | subj:>22  | @SUBJ %NH N NOM SG      |
| 22 | feels | feel    | obj:>20   | @+FMAINV %VA V PRES SG3 |
| 23 | he   | he       | subj:>24  | @SUBJ %NH PRON PERS NOM SG3 |
| 24 | 's   | be       | v-ch:>25  | @+FAUXV %AUX V PRES SG3 |
| 25 | hurting | hurt | obj:>22  | @-FMAINV %VA ING       |
The following sentence is a case of Subject NP with attached relative clause, one that usually creates conditions for failure in topdown parsers. However, the parser seems to lack grammaticality constraints and no heuristics to be able to decide on the subjecthood of one of the NPs available: so the result is a totally unconnected sentence.

(32) He told the committee the measure would merely provide means of enforcing the escheat law which has been on the books since Texas was a republic.

The same applies to the following sentence: no connections between the two main clauses and the disjunctive “OR” headed clause wrongly attached locally to “SINNED”.

(33) A man must be able to say, Father, I have sinned, or there is no hope for him.

The following sentence has the two main verbs unconnected,

(34) Wagner replied, Can't you just see the headline City Hooked for $172,000 ?

The same seems to happen in the following sentence where SAID is left unconnected,

(35) Yet, I responded, could not similar things be said about the art of the past ?
The following sentence is an exortative sentence which is totally mistaken by Connexor: TAKE is not tagged properly as verb but as a noun, WITHHELD is wrongly analysed as a Main Verb and left unconnected: the parser then wrongly coordinates WITHHELD with PONDERED.

(37) Take Augustine's doctrine of grace given and grace withheld : have you pondered
the dramatic qualities in this theology ?
3.3.2. Floating Quantifiers and Emphatic Pronouns

Both floating quantifiers and emphatic pronouns constitute non-arguments which need an adequate treatment both at a theoretical level and in the parser implementation. Both cases are nicely adjucted in GETARUNS. Only Connexor parses floating quantifiers as a case of pronominal binding: Link Grammar leaves the word 'all' unlinked, in the example below,

(14) The students went all to their classroom.

In the case of sentences containing emphatic pronominals it is again Connexor that can parse appropriately the output. Link Grammar wrongly tags the pronoun as adverbial in example 14a. and leaves the pronoun unlinked in example 14b.

(14a) The students like themselves to organize their parties.
(14b) The students are themselves organizers of their parties.

In sentence 14b. the pronoun is left unlinked also in Connexor however, as happens in the sentence below, which is taken from the Greval Corpus.

(15) In addition, some of Mr. Mason’s critics have implied that his type of ethnic humor is itself a form of racism.
Connexor takes “itself” as the predicative complement of BE and “a form of racism” is left unlinked. On the contrary, LG leaves “itself” unlinked and takes the following NP as predicative complement.

LINK GRAMMAR Output
No complete linkages found.
++Time 0.37 seconds (31.14 total)
Found 2 linkages (2 with no P.P. violations) at null count 1
Linkage 1, cost vector = (UNUSED=1 DIS=1 AND=0 LEN=48)

LEFT-WALL in addition_n, some of Mr.x Mason ’s_p critics_n have_v implied_v that_c his type_n of ethnic_a humor_n is_v [itself] a form_n of racism_n.

Constituent tree:
(S (PP In
  (NP addition))
, (S (NP (NP some)
  (PP of
   (NP (NP Mr. Mason ’s
    critics)))
  (VP have
   (VP implied
    (SBAR that
     (S (NP (NP his type)
      (PP of
       (NP ethnic humor)))
     (VP is itself
      (NP (NP a form)
       (PP of
        (NP racism)))))))))))
)

3.3.3. Clausal Subject

In the following section we analyze two cases of Clausal Subjects which however get wrongly interpreted by CONNEXOR and no connected output is presented. One could surmise that this is due to the presence of Indirect Discourse Parenthetical which messes up the structure; however, the following sentence – presented here below as (32) - is again unconnected even though it is just a simple and linear case of Clausal Subject.
GETARUNS deals with Clausal Subject at the interpretation level after the clause has been structurally licenced. It is again the presence of a specific lexical subcategorizing predicate that triggers the parse. Phases would have to licence both SUBJECT clause and the remaining clause where the governing predicate has to be appropriately analyzed. A sort of two-stage licensing.

(32) Complementing the political principle of nationalism is the legal principle of sovereignty.

(33) Insuring that the countries have a freedom of choice, he said, was the biggest detriment to the Soviet Union.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Insuring</td>
<td>insure</td>
<td>cla:&gt;12</td>
<td>@-FMAINV%VA ING</td>
</tr>
<tr>
<td>2</td>
<td>that</td>
<td>that</td>
<td>pm:&gt;5</td>
<td>@CS%CS CS</td>
</tr>
<tr>
<td>3</td>
<td>the</td>
<td>the</td>
<td>det:&gt;4</td>
<td>@DN&gt;%N DET</td>
</tr>
<tr>
<td>4</td>
<td>countries</td>
<td>country</td>
<td>subj:&gt;5</td>
<td>@SUBJ%NH N NOM PL</td>
</tr>
<tr>
<td>5</td>
<td>have</td>
<td>have</td>
<td>obj:&gt;1</td>
<td>@+FMAINV%VA V PRES</td>
</tr>
<tr>
<td>6</td>
<td>a</td>
<td>a</td>
<td>det:&gt;7</td>
<td>@DN&gt;%N DET SG</td>
</tr>
<tr>
<td>7</td>
<td>freedom</td>
<td>freedom</td>
<td>obj:&gt;5</td>
<td>@OBJ%NH N NOM SG</td>
</tr>
<tr>
<td>8</td>
<td>of</td>
<td>of</td>
<td>mod:&gt;7</td>
<td>@&lt;NOM-OF%N&lt; PREP</td>
</tr>
<tr>
<td>9</td>
<td>choice</td>
<td>choice</td>
<td>pcomp:&gt;8</td>
<td>@&lt;P%NH N NOM SG</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>he</td>
<td>he</td>
<td>subj:&gt;12</td>
<td>@SUBJ%NH PRON PERS NOM SG3</td>
</tr>
<tr>
<td>12</td>
<td>said</td>
<td>say</td>
<td>main:&gt;0</td>
<td>@+FMAINV%VA V PAST</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>was</td>
<td>be</td>
<td></td>
<td>@+FMAINV%VA V PAST</td>
</tr>
<tr>
<td>15</td>
<td>the</td>
<td>the</td>
<td>det:&gt;17</td>
<td>@DN&gt;%N DET</td>
</tr>
<tr>
<td>16</td>
<td>biggest</td>
<td>big</td>
<td>attr:&gt;17</td>
<td>@A&gt;%A SUP</td>
</tr>
<tr>
<td>17</td>
<td>detriment</td>
<td>detriment</td>
<td>comp:&gt;14</td>
<td>@PCOMPL-S%NH N NOM SG</td>
</tr>
<tr>
<td>18</td>
<td>to</td>
<td>to</td>
<td>goa:&gt;14</td>
<td>@ADV L%EH PREP</td>
</tr>
<tr>
<td>19</td>
<td>the</td>
<td>the</td>
<td>det:&gt;20</td>
<td>@DN&gt;%N DET</td>
</tr>
<tr>
<td>20</td>
<td>Soviet</td>
<td>soviet union</td>
<td>pcomp:&gt;18</td>
<td>@&lt;P%NH N NOM SG</td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>&lt;p&gt;</td>
<td>&lt;p&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 3.3.4. Nominal & Adjectival Complementation

Nominal Sentential Complements are usually correctly produced. However in some cases there are failures and either a Relative Clause is produced or no interpretation at all is reached. So in (42) below, the complement clause is produced correctly and connected by the verb TAKE to the governing head EVIDENCE. The same applies to (43) where the complement clause is produced correctly and connected by the verb TERMINATE to the governing head REQUEST. However (44) is an unconnected sentence where the governing NP “Small wonder” is analyzed in various manner and the attempt is made to have FEAR as main verb, failing.

(42) The Fulton County Grand Jury said Friday an investigation of Atlanta's recent primary election produced no evidence that any irregularities took place.
(43) This dissatisfaction led to Howsam's request that the video not be terminated before the end of the game.

(44) Small wonder, then, that we fear.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform relation</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Small</td>
<td>small</td>
<td>attr:&gt;2</td>
<td>@A&gt; %&gt;N A ABS</td>
</tr>
<tr>
<td></td>
<td>wonder</td>
<td>wonder</td>
<td></td>
<td>@NH %NH N NOM SG</td>
</tr>
<tr>
<td></td>
<td></td>
<td>@OBJ %NH N NOM SG</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>@SUBJ %NH N NOM SG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>,</td>
<td>,</td>
<td>meta:&gt;8</td>
<td>@ADVL %EH ADV</td>
</tr>
<tr>
<td>4</td>
<td>then</td>
<td>then</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>,</td>
<td>,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>that</td>
<td>that</td>
<td>pm:&gt;8</td>
<td>@CS %CS CS</td>
</tr>
<tr>
<td>7</td>
<td>we</td>
<td>we</td>
<td>subj:&gt;8</td>
<td>@SUBJ %NH PRON PERS NOM PL1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>fear</td>
<td>fear</td>
<td>main:&gt;0</td>
<td>@+FMAINV %VA V</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PRES</td>
</tr>
<tr>
<td>9</td>
<td>&lt;p&gt;</td>
<td>&lt;p&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>&lt;p&gt;</td>
<td>&lt;p&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Similar problems are detectable in (45), where a complement That-clause is generated and bound to the head noun IT by a Mod link.

(45) Is it not ironical that Roger Williams's state, Rhode Island, should have been the very last of the forty eight to establish a state university?

Other cases of MODifier sentences are sentence (46-47) where the governing verb of the that-clause APPLY is bound to the head noun PHILIPOFF rather than to the nominal governor ASSUMPTION; a relative clause bound to the head noun POSITION.

(46) This has an interesting analogy with the assumption stated by Philippoff that the deformational mechanics of elastic solids can be applied to flowing solutions.

(47) I am taking the position that the contract was clearly violated, Berger said.

Also considers the following sentence (23) where REMAIN has an indirect question as subject which is totally mistaken by both parsers.

(23) Just how many sub secrets were being handed over when the ring, watched for six months, was broken remained untold.

All these cases are correctly parser by GETARUNS.

### 3.4. Subject/Compless Sentences: Grammatical Principles Violated

Another case of sentence with a verb “BE” left with a subject NP and no complement: “AKIN TO” is computed as an adverbial modifying the SUBJect.

(24) Is the future of psychology akin to the rich future of physics at the time of Newton?
In sentence (25), the presence of the untensed “IF” clause prevents the parser from connecting the NP “STRIKES” analyzed as possible SUBJECT and the predicate “PREVENT”.

(25) Strikes threatening the security of the proprietorship, if internally motivated, prevent a healthy relationship.

It seems then the Dependency Parsers have no internal grammatical principle to prevent sentence parses from being accepted unless their obligatory Arguments have been fully parsed and interpreted. This is something nicely accounted for in GETARUNS.

3.4.1. Locative Inversion

Locative Inversion structures are not very frequent in Greval Corpus nor in UPenn WSJ corpus. However, this type of structure poses serious problems to word-level parser, seen that they systematically mistake the parse. As to its relevance for Phase theory, it must be said that this is a structure in which two constituents are displaced in a surface position which needs reinterpretation. Preverbal locative PP should be interpreted as the Oblique argument of the main verb rather than a simple Adjunct; while postverbal NP Object needs to be interpreted as Subject. This reinterpretation cannot be done on the basis of local information only: vP, i.e. finite verb information plus postverbal constituents need to wait for the existence of a locative PP in preverbal position in order to be reinterpreted. Otherwise, in case a NP had been parsed, the current structure would simply serve as predication of the Subject NP. Proper reinterpretation can thus only take place when clause level is reached, i.e. CP level, which can licence the assignment of semantic roles.

In the two following sentences (37-38), the governing verb BE and SIT are left without a SUBJECT NP, basically due to the fact that Locative Inversion has caused the NP to be displaced in postverbal position. In addition to that, sentence (38), is totally unconnected due
to the presence of other intermediate linguistic material which makes the parser collapse. So it is still important to notice that sentences receive a parse, which however is not an interpretation and in some cases just a fragmented interpretation.

(37) Here, in the old days - when they had come to see the moon or displays of fireworks - sat the king and his court while priests, soldiers, and other members of the party lounged in the smaller alcoves between.

(38) In his stead is a milquetoast version known as the corporation.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform relation</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>In</td>
<td>in</td>
<td>loc:4</td>
<td>@ADV %EH PREP</td>
</tr>
<tr>
<td>2</td>
<td>his</td>
<td>he</td>
<td>attr:3</td>
<td>@A %N PRON PERS GEN SG3</td>
</tr>
<tr>
<td>3</td>
<td>stead</td>
<td>stead</td>
<td>pcomp:1</td>
<td>@P %NH N NOM SG</td>
</tr>
<tr>
<td>4</td>
<td>is</td>
<td>be</td>
<td>main:0</td>
<td>@+FMAIN %VA V PRES SG3</td>
</tr>
<tr>
<td>5</td>
<td>a</td>
<td>a</td>
<td>det:7</td>
<td>@DN %N DET SG</td>
</tr>
<tr>
<td>6</td>
<td>milquetoast</td>
<td>milquetoast</td>
<td>attr:7</td>
<td>@A %N NOM SG</td>
</tr>
<tr>
<td>7</td>
<td>version</td>
<td>version</td>
<td>comp:4</td>
<td>@PCOMPL %NH N NOM SG</td>
</tr>
<tr>
<td>8</td>
<td>known</td>
<td>know</td>
<td>mod:7</td>
<td>@-FMAIN %VP EN</td>
</tr>
<tr>
<td>9</td>
<td>as</td>
<td>as</td>
<td>copred:8</td>
<td>@ADV %EH PREP</td>
</tr>
<tr>
<td>10</td>
<td>the</td>
<td>the</td>
<td>det:11</td>
<td>@DN %N DET</td>
</tr>
<tr>
<td>11</td>
<td>corporation</td>
<td>corporation</td>
<td>pcomp:9</td>
<td>@P %NH N NOM</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>&lt;p&gt;</td>
<td>&lt;p&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Cases like these are computed nicely in our framework given the fact that clause level interpretation takes place when the verb and its surrounding constituents have been fully parsed and locally interpreted. Again, access to the governing predicated ensures proper interpretation of the preposed PP.

3.5. Displaced Constituents

Another important factor influencing both Phase Theory and parsing is the presence of Displaced Constituents which need to be connected to their original argumental position. We saw Locative Inversion above and noted the objective difficulty of accounting for such facts on an argument level rather than on a clause CP level. The same type of reasoning shall have to be applied to cases listed here below.

Sentence (29) is a case of wrong attachment of a TO Indirect Object complement which has been displaced by the coordinate NP from its deverbal governing Head Noun REACTION. The Indirect Object is wrongly attached to Liberal Party.

(29) At the same time reaction among anti-organization Democratic leaders and in the Liberal party to the Mayor's reported plan was generally favorable.
The same argument applies to the following sentence where the wrong attachment of a TO Indirect Object complement to the adjacent preceding Head Noun is again due to lack of subcategorization requirements imposed by the governing verb OWE.

(30) That any sort of duty was owed by his nation to other nations would have astonished a nineteenth century statesman.

In the following sentence the wrong attachment depends on the wrong computation of the periphrastic use of HAD TO, meaning MUST/past, so that the displaced relative clause is computed as depending on the infinitival head verb USE. On the contrary, the verb of the Relative Clause should have been connected to the Noun Head governing the Relative Clause.

(31) A system had to be used which did not depend upon the feeding of the fluid into the manometer if measurements of the normal pressure were to be made in a reasonable time.

In both cases we may note that the theoretical approach would not be sufficient to recover optional arguments like TO Indirect Object, and that parsing strategies would be needed to explain full parse.

3.5.1. Cleft Sentences

In general, Cleft Sentences are not recognized by the two parsers, rather relative clauses are produced which are then attached to the local head Noun, or else a That-clause which is left unconnected. The result is not only a mistake in interpretation due to wrong attachment of the That-clause material, but also the impairment of LF construction as a whole. In the two following examples (39-41) the result is in (39) an unconnected That-Clause; whereas in (41) a Relative Clause is produced.

(39) But it has been during the last two centuries, during the scientific revolution, that our independence from the physical environment has made the most rapid strides.

(40) It was when he attempted to end the relationship that the murder took place.

(41) Not so, he answers, it is not the architect but the temple that is immortal.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform relation</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Not</td>
<td>not</td>
<td>neg:=2</td>
<td>@ADVL %EH NEG-PART</td>
</tr>
<tr>
<td>2</td>
<td>so</td>
<td>so</td>
<td>meta:=5</td>
<td>@ADVL %EH ADV</td>
</tr>
<tr>
<td>3</td>
<td>,</td>
<td>,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>he</td>
<td>he</td>
<td>subj:=5</td>
<td>@SUBJ %NH PRON NOM SG3</td>
</tr>
<tr>
<td>5</td>
<td>answers</td>
<td>answer</td>
<td>main:=0</td>
<td>@+FMAINV %VA V PRES SG3</td>
</tr>
<tr>
<td>6</td>
<td>,</td>
<td>,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>it</td>
<td>it</td>
<td>subj:=8</td>
<td>@SUBJ %NH PRON NOM SG3</td>
</tr>
<tr>
<td>8</td>
<td>is</td>
<td>be</td>
<td>obj:=5</td>
<td>@+FMAINV %VA V PRES SG3</td>
</tr>
<tr>
<td>9</td>
<td>not</td>
<td>not</td>
<td>neg:=8</td>
<td>@ADVL %EH NEG-PART</td>
</tr>
<tr>
<td>10</td>
<td>the</td>
<td>the</td>
<td>det:=11</td>
<td>@DN&gt; %&gt;N DET</td>
</tr>
<tr>
<td>11</td>
<td>architect</td>
<td>architect</td>
<td>comp:=8</td>
<td>@PCOMPL-S %NH N NOM SG</td>
</tr>
<tr>
<td>12</td>
<td>but</td>
<td>but</td>
<td>cc:=11</td>
<td>@CC %CC CC</td>
</tr>
<tr>
<td>13</td>
<td>the</td>
<td>the</td>
<td>det:=14</td>
<td>@DN&gt; %&gt;N DET</td>
</tr>
<tr>
<td>14</td>
<td>temple</td>
<td>temple</td>
<td>cc:=11</td>
<td>@PCOMPL-S %NH N NOM SG</td>
</tr>
<tr>
<td>15</td>
<td>that</td>
<td>that</td>
<td>subj:=16</td>
<td>@SUBJ %NH &lt;Rel&gt; PRON</td>
</tr>
<tr>
<td>16</td>
<td>is</td>
<td>be</td>
<td>mod:=14</td>
<td>@+FMAINV %VA V PRES SG3</td>
</tr>
<tr>
<td>17</td>
<td>immortal</td>
<td>immortal</td>
<td>comp:=16</td>
<td>@PCOMPL-S %NH A ABS</td>
</tr>
</tbody>
</table>
In (49) the parser produces a relative clause bound to the noun CLASS, rather than to the nominal governor then we have a coordinate VP, but no OBJECT NP is assigned and the NP headed by FRICTION is left unconnected.

(49) And it is precisely in this poorer economic class that one finds, and has always found, the most racial friction.

<table>
<thead>
<tr>
<th>#</th>
<th>Text</th>
<th>Baseform</th>
<th>Syntactic</th>
<th>Syntax and morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>And</td>
<td>and</td>
<td>cc:&gt;0</td>
<td>@CC %CC CC</td>
</tr>
<tr>
<td>2</td>
<td>it</td>
<td>it</td>
<td>subj:&gt;3</td>
<td>@SUBJ %NH PRON NOM SG3</td>
</tr>
<tr>
<td>3</td>
<td>is</td>
<td>be</td>
<td>main:&gt;0</td>
<td>@+FMAINV %VA V PRES SG3</td>
</tr>
<tr>
<td>4</td>
<td>precisely</td>
<td>precisely</td>
<td>man:&gt;3</td>
<td>@ADVL %EH ADV</td>
</tr>
<tr>
<td>5</td>
<td>in</td>
<td>in</td>
<td>loc:&gt;3</td>
<td>@ADVL %EH PREP</td>
</tr>
<tr>
<td>6</td>
<td>this</td>
<td>this</td>
<td>det:&gt;9</td>
<td>@DN&gt; %&gt;N DET DEM SG</td>
</tr>
<tr>
<td>7</td>
<td>poorer</td>
<td>poor</td>
<td>attr:&gt;8</td>
<td>@A&gt; %&gt;N A CMP</td>
</tr>
<tr>
<td>8</td>
<td>economic</td>
<td>economic</td>
<td>attr:&gt;9</td>
<td>@A&gt; %&gt;N A ABS</td>
</tr>
<tr>
<td>9</td>
<td>class</td>
<td>class</td>
<td>pcomp:&gt;5</td>
<td>@&lt;P %NH N NOM</td>
</tr>
<tr>
<td>10</td>
<td>that</td>
<td>that</td>
<td>obj:&gt;12</td>
<td>@OBJ %NH &lt;Rel&gt; PRON</td>
</tr>
<tr>
<td>11</td>
<td>one</td>
<td>one</td>
<td>subj:&gt;12</td>
<td>@SUBJ %NH PRON NOM SG</td>
</tr>
<tr>
<td>12</td>
<td>finds</td>
<td>find</td>
<td>mod:&gt;9</td>
<td>@+FMAINV %VA V PRES SG3</td>
</tr>
<tr>
<td>13</td>
<td>,</td>
<td>,</td>
<td></td>
<td>@CC %CC CC</td>
</tr>
<tr>
<td>14</td>
<td>and</td>
<td>and</td>
<td>cc:&gt;12</td>
<td>@+FAUXV %AUX V PRES SG3</td>
</tr>
<tr>
<td>15</td>
<td>always</td>
<td>always</td>
<td>frq:&gt;17</td>
<td>@ADVL %EH ADV</td>
</tr>
<tr>
<td>16</td>
<td>found</td>
<td>find</td>
<td>cc:&gt;12</td>
<td>@-FMAINV %VA EN</td>
</tr>
<tr>
<td>17</td>
<td>,</td>
<td>,</td>
<td>det:&gt;22</td>
<td>@DN&gt; %&gt;N DET</td>
</tr>
<tr>
<td>18</td>
<td>most</td>
<td>much</td>
<td>ad:&gt;21</td>
<td>@AD-A&gt; %E&gt; ADV SUP</td>
</tr>
<tr>
<td>19</td>
<td>racial</td>
<td>racial</td>
<td>attr:&gt;22</td>
<td>@A&gt; %&gt;N A ABS</td>
</tr>
<tr>
<td>20</td>
<td>friction</td>
<td>friction</td>
<td></td>
<td>@OBJ %NH N NOM SG</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>@&lt;P %NH N NOM SG</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>@PCOMPL-O</td>
<td>%NH N NOM SG</td>
</tr>
</tbody>
</table>

4. PARSER EVALUATION

4.1. Charniak’s Bad Parses

On his website, Charniak put a list of badly performing sentences in terms of precision. We looked through the structural representations and the overall impression we got is that the
parser is consistently attaching PPs locally within the previous NP rather than at clause level which results in a certain number of recurrent wrong decisions. It also has problems with coordinate structure as most parsers would, but again the same strategy seems to be at work. In some cases, compless relative clauses and complement clauses are missed: not so when a determiner is present in the former case, or a pronominal subject is being used rather than a nominal expression. We had the impression that the performance was also very much conditioned by the fact that test sentences belonged to the same genre/domain of the training treebank – the Penn Treebank. We ran Charniak’s parser with a section of Susan Corpus (a portion of Brown corpus – section A, 1550 sentences 33,000 words), and the problems increased quite a lot basically at the level of adjunct and argument attachment levels: one sentence every two had problems with ambiguous attachment or coordinate structure. The parser however performed very well with compless-clauses, only 12 mistakes out of 120 occurrences; nominally governed that-clauses, 3 mistakes over 30 occurrences. On the contrary cases of non-canonical sentences like clausal subjects, cleft sentences, locative inversion sentences, displaced constituents sentences, yes/no and wh+ questions, imperatives/exhortatives and other peculiar short sentences as also included in the list of bad parses get bad performance.

As to LinkGrammar and Connexor we ran a short evaluation test based on GREVAL corpus by choosing 100 sentences out of the 500 and evaluated the results: the recall for both parsers doesn’t get over 75%. We did the same with Charniak’s parser and got 100 sentences with bad attachment mistakes and other problems, again a 20% error rate when computed at sentence level. However, since constituents are much easier to match than GRs with gold standards we are unable to provide definite data: the overall impression is that 90% recall measured on constituency will not guarantee the same error rate when measured on GRs.

GETARUNS parser parses 89% of all text top down: then it parses 9.3% of the remaining linguistic material bottom up and adds it up to the parsed portion of the current sentence. That may produce wrong results in case a list has been partially parsed by the top down parser. But it produces right results whenever any additional complete subordinate or coordinate sentence structure has been left over – which constitutes the majority of cases. Overall almost the whole text – 98.3% - is turned into semantically consistent structures which have already undergone Pronominal Binding at sentence level in their DAG structural representation.

Table 21. Grammatical Relations produced by GETARUNS with Precision and Recall

<table>
<thead>
<tr>
<th>ALL-RELS</th>
<th>GOLD</th>
<th>VENICE</th>
<th>CORRECT</th>
<th>%PRECIS</th>
<th>RECALL/GOLD</th>
<th>RECALL/GUESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>xcomp</td>
<td>361</td>
<td>344</td>
<td>274</td>
<td>95.29</td>
<td>75.90</td>
<td>79.65</td>
</tr>
<tr>
<td>nsubj</td>
<td>1038</td>
<td>1038</td>
<td>883</td>
<td>100.00</td>
<td>85.06</td>
<td>85.06</td>
</tr>
<tr>
<td>iobj</td>
<td>158</td>
<td>126</td>
<td>99</td>
<td>79.74</td>
<td>62.65</td>
<td>78.57</td>
</tr>
<tr>
<td>dobj</td>
<td>409</td>
<td>394</td>
<td>331</td>
<td>96.33</td>
<td>80.92</td>
<td>84.01</td>
</tr>
<tr>
<td>arg_mod</td>
<td>38</td>
<td>30</td>
<td>26</td>
<td>78.94</td>
<td>68.42</td>
<td>86.66</td>
</tr>
<tr>
<td>ccomp</td>
<td>81</td>
<td>80</td>
<td>62</td>
<td>98.76</td>
<td>76.54</td>
<td>77.5</td>
</tr>
<tr>
<td>TOTAL</td>
<td>2085</td>
<td>2012</td>
<td>1675</td>
<td>96.49</td>
<td>80.33</td>
<td>83.25</td>
</tr>
</tbody>
</table>
We find it very important to remark the fact that the performance of our parser is mainly to be appreciated for the high coverage. None of the statistically and stochastically based parser reported in (Preis, 2003) reached such a high score. For the sake of comparison we report in Tab.21 the main data taken from the table presented in (Preis, 2003) to allow the reader to appreciate the results of our parser – where #occ are Gold data from annotation, BC=Briscoe & Carrol, BU=Buchholz, CH=Charniak, C1=Collins 1, C2=Collins 2. In particular, the overall number of main Grammatical Relations covered in the previous test is half the number (50% less) in comparison with ours. If we look at data referred to by precision or coverage, the easiest GR to parse, i.e. the NCSUBJ GR, we see that the number of cases found by the best parser in the previous test is by far lower that our result.

For the sake of comparison we also report the main data taken from the table presented under (Preis, 2003) to allow the reader to appreciate the results of our parser. In particular, the number of main Grammatical Relations treated in the previous test is half the number in comparison with ours. If we look at the easiest GR to parse, i.e. the NCSUBJ GR, we see that the number of cases found by the best parser in the previous test is by far lower that our result. The highest case of precision is for DOBJ in the BU parser which reaches 88.42% which is 8 points lower that our result. In absolute terms, limiting the comparison to the two most frequent GRs, the best parser – BU - has found 361 DOBJs against our 394; 891 NCSUBJs against our 1038.

The recall is also accordingly lower in absolute terms: there are 277 cases of correct DOBJs in BU against our 331, and 702 cases of correct NCSUBJs – in this case it is the BC parser that gets best recall – against our 883. And 277 and 702 are slightly better than chance - 67.72 and 67.63 respectively.

The impression one gets from the performance of statistically and stochastically based parsers is that they are inherently unable to cope with deep linguistic information. They are certainly impossible to undergo substantial improvements. On the contrary, rule based parsers would benefit from additional subcategorization frames as in our case: and for all those constructions which require setting up of new additional peripheral rules in the grammar, they would typically increase their coverage, as did our parser.

### Table 22. GR Precisions and Recalls as derived from Preis (2003)

<table>
<thead>
<tr>
<th>GF</th>
<th>prec</th>
<th>recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>dobj</td>
<td>409</td>
<td>85.83</td>
</tr>
<tr>
<td></td>
<td>87.48</td>
<td>88.42</td>
</tr>
<tr>
<td></td>
<td>76.53</td>
<td>84.43</td>
</tr>
<tr>
<td></td>
<td>75.55</td>
<td>86.16</td>
</tr>
<tr>
<td></td>
<td>74.57</td>
<td></td>
</tr>
<tr>
<td></td>
<td>84.85</td>
<td>75.31</td>
</tr>
<tr>
<td>iobj</td>
<td>158</td>
<td>31.93</td>
</tr>
<tr>
<td></td>
<td>67.09</td>
<td>57.75</td>
</tr>
<tr>
<td></td>
<td>51.9</td>
<td>27.53</td>
</tr>
<tr>
<td></td>
<td>67.09</td>
<td>27.09</td>
</tr>
<tr>
<td></td>
<td>69.62</td>
<td>27.01</td>
</tr>
<tr>
<td></td>
<td>70.25</td>
<td></td>
</tr>
<tr>
<td>ncssubj</td>
<td>1038</td>
<td>81.99</td>
</tr>
<tr>
<td></td>
<td>82.47</td>
<td>85.83</td>
</tr>
<tr>
<td></td>
<td>72.93</td>
<td>71.8</td>
</tr>
<tr>
<td></td>
<td>70.13</td>
<td>79.19</td>
</tr>
<tr>
<td></td>
<td>65.99</td>
<td>81.29</td>
</tr>
<tr>
<td></td>
<td>69.46</td>
<td></td>
</tr>
<tr>
<td>obj2</td>
<td>19</td>
<td>27.45</td>
</tr>
<tr>
<td></td>
<td>73.68</td>
<td>46.15</td>
</tr>
<tr>
<td></td>
<td>31.58</td>
<td>42.11</td>
</tr>
<tr>
<td></td>
<td>81.82</td>
<td>42.11</td>
</tr>
<tr>
<td></td>
<td>50.57</td>
<td></td>
</tr>
<tr>
<td>arg_mod</td>
<td>41</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>75.68</td>
</tr>
<tr>
<td></td>
<td>68.29</td>
<td>78.12</td>
</tr>
<tr>
<td></td>
<td>60.98</td>
<td>82.86</td>
</tr>
<tr>
<td></td>
<td>70.73</td>
<td>82.86</td>
</tr>
<tr>
<td></td>
<td>70.73</td>
<td></td>
</tr>
<tr>
<td>clausal</td>
<td>403</td>
<td>43.27</td>
</tr>
<tr>
<td></td>
<td>52.61</td>
<td>75.79</td>
</tr>
<tr>
<td></td>
<td>71.46</td>
<td>62.19</td>
</tr>
<tr>
<td></td>
<td>43.67</td>
<td>50.57</td>
</tr>
<tr>
<td></td>
<td>32.75</td>
<td>49.11</td>
</tr>
<tr>
<td></td>
<td>27.30</td>
<td></td>
</tr>
</tbody>
</table>

As a last comment, we started evaluating subsets of GREVAL corpus with the online version of "Connexor" dependency parser, on the assumption that that version would be
identical or even better than the one commercially available. We did that because this parser is regarded the best dependency parser on the market. We tried out a subset of 50 sentences, and on a first perusal of the output we discovered that only 40 sentences contained correct, and fully connected representations. The remaining 10 sentences either presented unconnected heads, or misconnected ones due to wrong attachments. Some remarks on the possible reasons for that:

- bottom up local parsing techniques are good at coping with typically hard to parse structures for a top down parser like coordinate structures but they are bad at computing long distance dependencies;
- they are good at computing attachment whenever it is local, but they make mistakes when there are extraposed elements;
- dependency parsing does not seem to obey to generally accepted grammaticality principles like the obligatoriness of SUBJect constituents, nor the need to provide some landing site for extracted wh- elements in relative and interrogative clauses;
- control structures like small clauses for predicative complements and adjuncts are all attached locally, which is not always the case.

So, even though word-level parsing may be more effective as to the number of connections (constituents) safely produced, without leaving off any fragment or skimmed fragment, it is nonetheless faced with the hard task of recomposing clause level control mechanisms which in a top down constituency-based parser are given for granted.

The F-measure derived from our P and R according to the usual formula:

$$F_1(r,p) = \frac{2rp}{r+p}$$

is 89.38%, which is by far higher than the 75% reported in (Crouch et al., 2002) as being the best result obtained by linguistic parsers today. On the contrary, on the 50 sentences used with Connexor the Recall was 74%, again measured on all main GRs.
Chapter 4

PARSING 2:
DEEP LINGUISTICALLY-BASED PARSING

1. INTRODUCTION

In this chapter we will present the Deep parser, a fully symbolic rule-based parser that is the core of the system GETARUNS. We will focus on the problem of ambiguities and the way in which such a parser can cope with them in a principled way. The parser we present now is strictly topdown and is used only in restricted domains and with limited length sentences (below 35 words). The main field of application for such a parser would be as a means to detect ungrammaticality, which should coincide with unparsability. This is out of the scope both of shallow, partial and statistically based parsing. So eventually in this chapter we will also tackle issues related to the use of shallow or partial techniques for unrestricted text NLP.

In the sections below we will criticize current statistical approaches for being inherently ill-founded. People working in the empirical framework have tried to credit the point of view that what happened to the speech research paradigm was also applicable to the NLP paradigm as a whole. People like Ken Church have paved the way to the NLP scientific community by presenting the task of NLP as being parallel with what the speech community did, in the Introduction to the Special Issue on Using Large Corpora published by Computational Linguistics in A. Warwick (1993). In this seminal but partially misleading article, the author claimed that what happened to the speech community in the last 20 years or so, that is a slow transformation of their approach to natural language understanding from a knowledge-based approach to an empirical statistically-based approach, could also be applied to NLP. Seen that both communities were apparently dealing with the same basic materials, linguistic units of some kind, the trsition to the empirical approach was simply to be treated as a truis by people of the NLP community. The only true part of the argument was on the contrary constituted by the need felt by most computational linguists to move away from the analysis of hand-made list of sentences and start tackling 'real texts'.

That is, the need to move away from what theoretical linguists would still regard as a sufficiently representative sample of the language under analysis - usually a list of 3-4000 simplex sentences; in order to start using large corpora. However, it is just the fact that the linguistic units being addressed as their main object were totally different, that the
comparison does not hold and is badly misleading. In the section below, we will discuss in
detail the reason why in our opinion the empirical statistical approach to NLP it its current
experimental and empirical design should not be pursued, and in fact this would be the same
reason why the speech community has come to the same conclusions already some time ago
with respect to the need to address higher level linguistic units in the speech waveform
usually referred to as prosodic units (see Delmonte, 2000d). In particular, both the speech
synthesis community and the speech recognition community have implicitly admitted to the
fact that the choice of the linguistic units to be addressed, i.e. a segmental unit constitutes
nowadays the bottleneck for further improvements in the field (see R. Sproat and J. van Santen,
1998).

On a strictly intuitive basis, the segmental unit approach is wrong for the simple reason
that one would need to model information coming from all linguistic levels into the one single
segment - being it a phone, a phoneme, a diphone unit or a triphone unit from the n-gram
approach advocated by the speech community and transferred by the new empiricists onto the
part-of-speech tagging task. This position is both untenable and implausible. It is untenable
for reasons related to tagset size and linguistic coverage, i.e. how big a training corpus should
be in order to cope with the well-known problem of data sparseness or sparsity. For instance
in the LinGO framework, the tagset being used amounts to over 8,000 different single tags,
which makes it very hard even with a database of 10,000 utterance to make up a
representative and statistical useful training corpus, as their authors comment in the entry
webpage of the project at http://lingo.stanford.edu, under the title "Why Another (Type of)
Treebank?" which we report here below:

“For the past decade or more, symbolic, linguistically oriented methods like those pursued
within the HPSG framework and statistical or machine learning approaches to NLP have
typically been perceived as incompatible or even competing paradigms; the former, more
traditional approaches are often referred to as 'deep' NLP, in contrast to the comparatively
recent branch of language technology focusing on 'shallow' (text) processing methods.
Shallow processing techniques have produced useful results in many classes of applications,
but they have not met the full range of needs for NLP, particularly where precise
interpretation is important, or where the variety of linguistic expression is large relative to the
amount of training data available. On the other hand, deep approaches to NLP have only
recently been able to achieve broad enough grammatical coverage and sufficient processing
efficiency to allow the use of HPSG-type systems in certain types of real-world applications.
Fully-automated, deep grammatical analysis of unrestricted text remains an unresolved
challenge. In particular, realistic applications of analytical grammars for natural language
parsing or generation require the use of sophisticated statistical techniques for resolving
ambiguities. We observe general consensus on the necessity for bridging activities, combining
symbolic and stochastic approaches to NLP...
An important recent advance in this area has been the application of log-linear models
(Agresti, 1990) to modeling linguistic systems. These models can deal with the many
interacting dependencies and the structural complexity found in constraint-based or
unification-based theories of syntax. The availability of even a medium-size treebank would
allow us to begin exploring the use of these models for probabilistic disambiguation of HPSG
grammars.”

And further on the webpage includes details of the implementation, which we report here
below,
"The key innovative aspect of the Redwoods approach to treebanking is the anchoring of all linguistic data captured in the treebank to the HPSG framework and a generally-available broad-coverage grammar of English, viz. the LinGO English Resource Grammar. Unlike existing treebanks, there will be no need to define a (new) form of grammatical representation specific to the treebank (and, consequently, less dissemination effort in establishing this representation). Instead, the treebank will record complete syntacto-semantic analyses as defined by the LinGO ERG and provide tools to extract many different types of linguistic information at greatly varying granularity. Depth of Representation and Transformation of Information Internally, the [incr tsdb()] database records analyses in three different formats, viz. (i) as a derivation tree composed of identifiers of lexical items and constructions used to construct the analysis, (ii) as a traditional phrase structure tree labeled with an inventory of some fifty atomic labels (of the type S, NP, VP et al.), and (iii) as an underspecified MRS meaning representation... While (ii) will in many cases be similar to the representation found in the Penn Treebank, (iii) subsumes the functor-argument (or tectogrammatical) structure as it is advocated in the Prague Dependency Treebank or the German TiGer corpus. Most importantly, however, representation (i) provides all the information required to replay the full HPSG analysis...”

Even though the overall tone of the researchers involved in the LinGO consortium is enthusiastic, the actual coverage of the PET parser in real texts is as usual limited by grammar and vocabulary coverage. However we find the approach in line with ours even though the underlying technical framework is totally different.

1.2. Shallow and Partial Parsing and Statistical Processing

In their Chapter - Language Analysis and Understanding (Karlsson & Karttunen, 1995) in the section dedicated to Shallow Parsing [ibid:113-114], they use the term shallow syntax as a generic term for analyses that are less complete than the output from a conventional parser. The output from a shallow analysis is not a phrase-structure tree. A shallow analyzer may identify some phrasal constituents, such as noun phrases, without indicating their internal structure and their function in the sentence. Another type of shallow analysis identifies the functional role of some of the words, such as the main verb, and its direct arguments. Systems for shallow parsing normally work on top of morphological analysis and disambiguation. The basic purpose is to infer as much syntactic structure as possible from the lemmata, morphological information, and word order configuration at hand. Typically shallow parsing aims at detecting phrases and basic head/modifier relations. A shared concern of many shallow parsers is the application to large text corpora. Frequently partial analyses are allowed if the parser is not potent enough to resolve all problems.

Abney (1996) comments on statistical methods applied to the problem of Part-of-Speech Tagging as being a fairly success story. People engaging in this kind of pioneering research effort at the beginning of the '90s showed that it was possible to "carve part-of-speech disambiguation out of the apparently monolithic problem of natural language understanding, and solve it with impressive accuracy" (Abney 1996:1). What the people (Church (1998), DeRose (1988), Garside (1987)), involved in that approach actually were interested in
showing was that even if the exact solution to the NLU problem is far beyond reach, a reasonable approximate solution is quite feasible.

In Abney (1996) the author discusses the feasibility of another important aspect of the NLU problem: that of syntactic analysis, by proposing as a solution what he defines "Partial Parsing". This is regarded as a cover term for a range of different techniques for recovering some but not all of the information contained in a traditional syntactic analysis. As he comments "Partial parsing techniques, like tagging techniques, aim for reliability and robustness in the face of the vagaries of natural text, by sacrificing completeness of analysis and accepting a low but non-zero error rate." (Abney 1996:3)

Further on in the same paper, we are told that a 5% error rate is certainly a remarkable achievement in terms of accuracy, and can be achieved in a very short term indeed - one month work of a computational linguist. However, if we consider the sentence as the relevant unit onto which to gauge the goodness of such an accuracy figure, we come up with a completely different figure: assuming an average of 20-word sentences and 4% per-word error rate we end up with a 56% per-sentence error rate. To get a 4% per-sentence error rate, we require accuracy figures which range beyond 99%, actually 99.98%. This is unfeasible for any statistically or even rule-based tagger presented in the literature.

Partial parsing tries to offer a solution to the problem posed by unrestricted texts to traditional parsers which, basically due to the incompleteness of both lexicon and grammar, are subject to failures and errors. Errors are also a subproduct of the length of sentences and the inherent ambiguity of grammars. What partial parsers do is recovering the nonrecursive core of constituent structure by factoring out the parse into those pieces of structure that can be reliably recovered with a small amount of syntactic information. This is usually done without using lexical information as would typically do all unification based parsers. Chunks and simplex clauses can then safely be used for bootstrapping lexical association information which is used to take decisions related to attachment of arguments and adjuncts. The output of any such chunkers can be regarded as a useful intermediate representation to be used for any further computation. In terms of efficiency, as Abney (1996:10) reports, the fastest parsers are all deterministic rule-based partial parsers.

We have developed our partial parser - that we discussed in the preceding chapter - as a finite-state cascaded machine that produces a final parser by cycling on the input and passing the output of each parse to the following FSA. The parser was originally a recursive transition network, and has been built expressly to eliminate recursion from the parsing process. However, even if the partial parser is good at recognizing constituent chunks or to do phrase-spotting, without having to analyze the entire sentence, when it comes to clauses the error rate increases a lot up to statistically valid threshold of 5-6%. As Church, Gale, Hanks & Hindle (1989) has shown, a partial parser can be put to use in a variety of ways, in particular in extracting subject-verb and verb-object pairs in order to provide a crude model of selectional restrictions.

1.3. Issues Related to the Use of Partial and Shallow Approaches

In his Chapter on Sentence Modeling and Parsing, Fernando Pereira (1995) defines what in his opinion are the main issues in applying linguistic theory to the development of computational grammars: coverage, predictive power and computational requirements.
However this is done in order to promote the use of statistically based approaches to parsing and thus the issues are highlighted as shortcomings.

As far as Coverage is concerned his comment is the following:

“Linguistic theories are typically developed to explain puzzling aspects of linguistic competence, such as the relationships between active and passive sentences, the constraints on use of anaphoric elements, or the possible scopes of quantifying elements such as determiners and adverbs. However, actual language involves a wide range of other phenomena and constructions, such as idioms, coordination, ellipsis, apposition and extraposition, which may not be germane to the issues addressed by a particular linguistic theory or which may offer unresolved challenges to the theory. Therefore, a practical grammar will have to go far beyond the proposals of any given theory to cover a substantial proportion of observed language. Even then, coverage gaps are relatively frequent and difficult to fill, as they involve laborious design of new grammar rules and representations.”

Then he continues by lamenting the lack of Predictive Power of linguistic grammars, which in his opinion

“… being oriented towards the description of linguistic competence, are not intended to model distributional regularities arising from pragmatics, discourse and conventional use that manifest themselves in word and construction choice. Yet those are the regularities that appear to contribute most to the estimation of relative likelihoods of sentences or analyses.” [ibid:137]

As far as Computational Requirements are concerned, recent implementations seem to have made progress in the direction towards tractable grammatical formalisms which are reported to constitute polynomial-time and space parsing algorithms: however he then asserts that, “… even polynomial-time algorithms may not be sufficiently fast for practical applications, given effect of grammar size on parsing time.” [ibid:138]

Eventually in his “Future Directions” paragraph, Pereira comments on the current challenge which we also endorse fully:

“The issue that dominates current work in parsing and language modeling is to design parsers and evaluation functions with high coverage and precision with respect to naturally occurring linguistic material (for example, news stories, spontaneous speech interactions). Simple high-coverage methods such as n-gram models miss the higher-order regularities required for better prediction and for reliable identification of meaningful relationships, while complex hand-built grammars often lack coverage of the tail of individually rare but collectively frequent sentence structures (cf. Zipf’s law). Automated methods for grammar and evaluation function acquisition appear to be the only practical way to create accurate parsers with much better coverage. The challenge is to discover how to use linguistic knowledge to constrain that acquisition process.” [ibid:140]

More or less of the same overall tone is the Chapter on Robust Parsing by Ted Briscoe (1995), where he comments on the question of disambiguation which will also be discussed by us further on in this chapter. Here is his comment:
“Despite over three decades of research effort, no practical domain-independent parser of unrestricted text has been developed. Such a parser should return the correct or a useful close analysis for 90% or more of input sentences. It would need to solve at least the following three problems, which create severe difficulties for conventional parsers utilizing standard parsing algorithms with a generative grammar:

1. chunking, that is, appropriate segmentation of text into syntactically parsable units;
2. disambiguation, that is, selecting the unique semantically and pragmatically correct analysis from the potentially large number of syntactically legitimate ones returned; and
3. undergeneration, or dealing with cases of input outside the systems’ lexical or syntactic coverage.

Conventional parsers typically fail to return any useful information when faced with problems of undergeneration or chunking and rely on domain-specific detailed semantic information for disambiguation.

The problem of chunking is best exemplified by text sentences (beginning with a capital letter and ending with a period) which contain text adjuncts delimited by dashes, brackets or commas which may not always stand in a syntactic relation with surrounding material… an analysis of the 150K word balanced Susanne Corpus… reveals that over 60% of sentences contain internal punctuation marks and of these around 30% contain text-medial adjuncts… Disambiguation using knowledge-based techniques requires the specification of too much detailed semantic information to yield a robust domain-independent parser. Yet analysis of the Susanne Corpus with a crude parser suggests that over 80% of sentences are structurally ambiguous… (statistically based) systems have yielded results of around 75% accuracy in assigning analyses to (unseen) test sentences from the same source as the unambiguous training material. The barrier to improvement of such results currently lies in the need to use more discriminating models of context, requiring more annotated training material to adequate estimate the parameters of such models. This approach may yield a robust automatic method for disambiguation of acceptable accuracy, but the grammars utilized still suffer from undergeneration, and are labour-intensive to develop. Undergeneration is a significant problem, in one project, a grammar for sentences from computer manuals containing words drawn from a restricted vocabulary of 3000 words which was developed over three years still failed to analyze 4% of unseen examples… This probably represents an upper bound using manual development of generative grammars; most more general grammars have far higher failure rates in this type of test. Early work on undergeneration focussed on knowledge-based manual specification of error rules or rule relaxation strategies… This approach, similar to the canonical parse approach to ambiguity, is labour-intensive and suffers from the difficulty of predicting the types of error or extragrammaticality liable to occur.” (Briscoe 1995:142)

In his Chapter on Statistical Parsing, John A. Carroll (2000) gives a rather pessimistic view of current and future possibilities for statistical approaches in NLP. This even though he is among the people working within the HPSG constrain unification framework quoted above in the LinGO project, who seem convinced of the contrary and are actually working with optimistic plans as the presentation of the Redwoods Treeback effort and subsequent parser creation and testing demonstrates. Carroll (2000:525) states what in his opinion, are the major problems that parsing of natural language should address and they are:

a. how to resolve the (lexical, structural, or other) ambiguities that are inherent in real-world natural language text;
b. how to constrain the form of analyses assigned to sentences, while still being able to return "reasonable" analyses for as wide a range of sentences as possible.
He then criticizes NLP approaches wrought within the generative linguistic tradition because in his opinion they have a number of major drawbacks that disqualify them as adequate and successful candidates for the analysis of real texts. These parsing systems - like ours - use hand-built grammars in conjunction with parsing algorithms which either,

c. return all possible syntactic analyses, which would then be passed on to detailed, domain-dependent semantic and pragmatic processing subsystems for disambiguation;

d. use special purpose, heuristic parsing algorithms that are tuned to the grammar and

e. invoke hand-coded linguistic or domain-specific heuristics to perform disambiguation and

f. invoke grammar relaxation techniques to cope with extragrammatical input

this being a position which I find totally in line with our approach except for c, being deterministic and as such using all possible semantic knowledge as soon as possible and in any case before any major constituent is being licensed.

However, Carroll (2000:526) finds that such an approach cannot be good because it has the following dramatic drawbacks:

g. computing the full set of analyses for a sentence of even moderate length with a wide-coverage grammar is often intractable (we also agree with this point)

h. if this is possible, there is still the problem of how to apply semantic processing and disambiguation efficiently to a representation of a (possibly very) large set of competing syntactic analyses (same as above);

i. although linguistic theories are often used as devices for explaining interesting facts about a language, actual text in nontrivial domains contains a wide range of poorly understood and idiosyncratic phenomena, forcing any grammar to be used in a practical system to go beyond established results and requiring much effort in filling gaps in coverage;

j. hand-coding of heuristics is a labour-intensive task that is prone to mistakes and omissions, and makes system maintenance and enhancement more complicated and expensive;

k. using domain-specific hand-coded knowledge hinders the porting of a system to other domains or sublanguages.

As to i.-k., seen that j.-k. are given as a logical/natural consequence to the widely acceptable and shared assertion contained in i. it needs looking into the quite obvious fact that neither handcrafted grammars nor statistical ones are exempt from being inherently language-dependent abstract representation of the linguistic structure of a specific language in a specific genre and domain. Statistically built will do that by tuning their grammar to a given training corpus with a certain number of caveats that we will discuss in detail below. Contrary to what is being assumed by Carroll, rule-based symbolic systems can take advantage of the generality of core grammars which as we will discuss further on offer core rules to be applied over an enormous gamut/range of natural languages, something which empirically built systems cannot take advantage of.

So it would seem that a lot of the current debate over the uselessness, inefficiency, inherent inability of rule-based symbolic systems is due to a fundamental choice in the type of parsing strategy and parsing algorithm, which as I understand it reflects Carroll's choice of constraint-based and unification-based formalisms. These formalisms supplanted ATNs and RTNs in the '80s and slowly came to fore of the linguistic audience supported by a number of linguistic theories, one of which, LFG is also at the heart of our system.
The effort devoted to the construction of hand-built lexica and rules are useful nonetheless to the theoretical linguistic community and certainly to students. These algorithms are unfit for the parsing of real texts required by systems for Information Retrieval and the more ambitious Natural Language Understanding community.

2. Linguistically-based Parsing and Linguistic Strategies

Ambiguity is one of the main problems faced by large-scale computational grammars. Ambiguities can arise basically from Part Of Speech (POS) tagging associated to any given word of the input sentence. Natural languages, with narrow tagset of say 100 tags, will come up with an average ambiguity ratio of 1.7/1.8 per word: i.e. each input word can be assigned in average to two different tags. This base level ambiguity has to be multiplied by rule interactions, via alternative subcategorization frames related to governing lexical entries, or simply from linguistically motivated syntactic ambiguities. As opposed to human speakers, computational grammars are not yet able to always (100%) determine the contextually correct or intended syntactic analysis from a set of alternative analyses. Thus, a computational grammar based on unification and constraint based formalisms, and covering a realistic fragment of natural language will, for a given sentence, come up with a large number of possible analyses, most of which are not perceived by humans or are considered inappropriate in the given context. In order to reduce the number of possible analyses, local tag disambiguation is carried out on the basis of statistical and syntactic algorithms. For a sentence like,

(1) John wanted to leave

there should only be one analysis available due to local linguistic and statistically derived restrictions that prevent the word "to" from being interpreted as a preposition after the word "wanted" a verb and not a noun and be assigned to the category of complementizers or verb particles. So, even though on the basis of a bottom up analysis, a word like "leave" could be analysed both as a noun and as a base verb, by means of disambiguation carried out in a topdown fashion, the word "to" will trigger the appropriate interpretation of the word "leave" as base verb and not as noun. This is not always ensured, in particular in case a chart parser with a bottom up policy is chosen and all possible linguistic analysis are generated in a parallel fashion.

Disambiguation should be carried out on a separate module, and not be conflated with parsing in case one wants to simulate Garden Paths while at the same time avoiding crashes or freezing of the parser to take place. This allows the topdown depth-first parser to backtrack and try the other analysis. However, backtracking should be allowed only whenever real Garden Path are in order. This kind of information is not hidden but can be derived from linguistic information.

In this chapter we will discuss our proposal to solve ambiguity by means of linguistically related lexical and structural information which is used efficiently in a number of disambiguation strategies. Since the parser we will present is a multilingual parser, strategies
will be also related to UG parameters in order to take advantage of the same Core Grammar and use Peripheral Rules for that task.

2.1. Shallow and Deep Parsing

The shallow parsing approach is very efficient and usually prevents failures. However the tradeoff with deep parsing is a certain percentage of text not being fully parsed due to local failures, especially at clause level. This may also result as a wrong choice of tag disambiguation, which carries on to constituent level. Another important shortcoming is the inherent inability of this approach to ensure a semantically consistent mapping of all resulting constituent structures. This is partly due to the fact that clause level analysis is only approximated and not always fully realized. In addition, all attachments are also approximated in lack of a stable clause level analysis. Eventually, also subcategorization frames cannot be used consistently but only tentatively matched with the available information.

As a counterpart to this situation, shallow parsers can easily be ported to other languages and so satisfy an important requirement, that of reusability. In theoretical linguistic terms, this concept is easily understood as a subdivision of tasks between the parameters and principles components vs the rule component which being universal, relies on X-bar based constituency.

Though X-bar based parsing may be inefficient, one way to improve it would be that of encoding lexical ambiguity, both at word level and at the ensuing structural level. We would like to assume that specialization in language dependent ambiguity resolution is one of the components of the language acquisition process.

The lexicon as the source of syntactic variation is widely accepted in various theoretical frameworks (see Bresnan, in press). We assume that be it shallow or deep, parsing needs to be internally parameterized in order to account for ambiguities generated both at structural and at semantic level.

The parser we present has been built to simulate the cognitive processes underlying the grammar of a language in use by a speaker, taking into account the psychological nuances related to the wellknown problem of ambiguity, which is a pervading problem in real text/life situation, and it is regarded an inseparable benchmark of any serious parser of any language to cope with.

As already discussed in the previous chapter and presented partially in chapter 3, in order for a parser to achieve psychological reality it should satisfy three different types of requirements: psycholinguistic plausibility, computational efficiency in implementation, coverage of grammatical principles and constraints. Principles underlying the parser architecture should not conform exclusively to one or the other area, disregarding issues which might explain the behaviour of the human processor. In accordance with this criterion, we assume that the implementation should closely mimick phenomena such as Garden Path effects, or an increase in computational time in presence of semantically vs. syntactically biased ambiguous structures. We also assume that a failure should ensue from strong Garden Path effects and that this should be justified at a psycholinguistic interpretation level.

In other words, looking at parsing from a performance-based perspective, to justify speakers' psycholinguistic behaviour and its simulation in a running parser, we think it should be organized as a toptdown depth-first symbolic rule compiler ordered according to efficiency
criteria and using Lookahead and a Well-Formed Substring Table (WFST) not to duplicate effort.

This is just the opposite of a Unification Grammar which uses Chart parsing in a bottom up breadth-first manner which is norm in Constraint-Based formalisms like HPSG or LFG. However, what's more important, the parser should know what kind of ambiguities could cause unwanted Garden-Paths and Crashes, to refrain from unwanted failures in order to mimic human processing. Constraint unification is in our opinion unable to satisfy the efficiency requirements and prevent unwanted failures: it is insufficient to simply have a list of lexical items with their features, and a grammar with a list of rules which obey to a certain number of principles and constraints. A "sound" parser needs to be told which ambiguous structures are expected in which language.

In general terms, ambiguity is generated by homophonous words in understanding activities and by homographs in reading activities. In both cases Garden Paths or Crashes may only result in a given language in presence of additional conditions which are strictly dependent on the structure of the lexicon and the grammar. But some UG related parameters, like the "OMISSIBILITY OF THE COMPLEMENTIZER" in English may cause the parser to crash or freeze. Generally speaking, all types of ambiguity affecting parsing at a clause level will cause the parser to go into a Garden Path. The typical example quoted in psycholinguistic literature is the reduced relative case, reported here below, determined by the lexical ambiguity of English verbs being at the same time interpretable as Past Participle - Past Tense and shown below in the Reduced Relative Clause well-known example,

(2) The horse raced past the barn fell.

Figure 2. The Deep Complete Parser of GETARUNS.
is one such case. The English speaker will attempt treating the verb "raced" as main tensed verb, but on discovery of sentence final verb "fell" which can only be interpreted as tensed past tense the whole sentential level analysis crashes and a Garden Path ensues causing a complete restart of the mental parser.

We assume that from a psycholinguistic point of view, parsing requires setting up a number of disambiguating strategies, basically to tell arguments apart from adjuncts and reduce the effects of backtracking.

Our parser is based on LFG theoretical framework (see Bresnan, [16]) and has a highly interconnected modular structure. It is a top-down depth-first DCG-based parser written in Prolog which uses a strong deterministic policy by means of a lookahead mechanism with a WFST to help recovery when failure is unavoidable due to strong attachment ambiguity.

It is divided up into a pipeline of sequential but independent modules which realize the subdivision of a parsing scheme as proposed in LFG theory where a c-structure is built before the f-structure can be projected by unification into a DAG. In this sense we try to apply in a given sequence phrase-structure rules as they are ordered in the grammar: whenever a syntactic constituent is successfully built, it is checked for semantic consistency, both internally for head-spec agreement, and externally, in case of a non-substantial head like a preposition dominates the lower NP constituent; other important local semantic consistency checks are performed with modifiers like attributive and predicative adjuncts. In case the governing predicate expects obligatory arguments to be lexically realized they will be searched and checked for uniqueness and coherence as LFG grammaticality principles require.

Whenever a given predicate has expectancies for a given argument to be realized either optionally or obligatorily this information will be passed below to the recursive portion of the parsing: this operation allows us to implement parsing strategies like Minimal Attachment, Functional Preference and other ones (see Delmonte and Dolci 1989; Delmonte and Dolci 1997).

As to multilinguality, the basic tenet of the parser is based on a UG-like perspective, i.e. the fact that all languages share a common core grammar and may vary at the periphery: internal differences are predicted by parameters. The DCG grammar allows the specification of linguistic rules in a highly declarative mode: it works topdown and by making a heavy use of linguistic knowledge may achieve an almost complete deterministic policy. Parameterized rules are scattered throughout the grammar so that they can be made operative as soon as a given rule is entered by the parser.

In particular, a rule may belong either to a set of languages, e.g. Romance or Germanic, or to a subset thereof, like English or Italian, thus becoming a peripheral rule. Rules are activated at startup and whenever a switch is being operated by the user, by means of logical flags appropriately inserted in the right hand side of the rule. No flags are required for rules belonging to the common core grammar.

Some such rules include the following ones: for languages like Italian and Spanish, a Subject NP may be an empty category, either a referential little pro or an expletive pronoun; Subject NPs may be freely inverted in postverbal position, i.e. preverbal NP is an empty category in these cases. For languages like Italian and French, PP or adverbial adjuncts may intervene between Verb and Object NP; adjectival modifiers may be taken to the right of their head Noun. For languages like English and German, tense and mood may be computed in CP internal position, when taking the auxiliary or the modal verb. English allows an empty
Complementizer for finite complement and relative clauses, and negation requires do-support. Italian only allows for a highly genre marked (literary style) untensed auxiliary in Comp position.

Syntactic and semantic information is accessed and used as soon as possible: in particular, both categorial and subcategorization information attached to predicates in the lexicon is extracted as soon as the main predicate is processed, be it adjective, noun or verb, and is used to subsequently restrict the number of possible structures to be built. Adjuncts are computed by semantic compatibility tests on the basis of selectional restrictions of main predicates and adjuncts heads.

Syntactic rules are built using CP-IP functional maximal projections. Thus, we build and process syntactic phenomena like wh-movement before building f-structure representations, where quantifier raising and anaphoric binding for pronominals takes place. In particular, all levels of Control mechanisms which allow coindexing at different levels of parsing give us a powerful insight into the way in which the parser should be organized.

Yet the grammar formalism implemented in our system is not fully compliant with the one suggested by LFG theory, in the sense that we do not use a specific Feature-Based Unification algorithm but a DCG-based parsing scheme. In order to follow LFG theory more closely, unification should have been implemented. On the other hand, DCGs being based on Prolog language, give full control of a declarative rule-based system, where information is clearly spelled out and passed on and out to higher/lower levels of computation. In addition, we find that topdown parsing policies are better suited to implement parsing strategies that are essential in order to cope with attachment ambiguities (but see below). We use XGs (extraposition grammars) introduced by Pereira (1981;1983). Prolog provides naturally for backtracking when allowed, i.e. no cut is present to prevent it. Furthermore, the instantiation of variables is a simple way for implementing the mechanism for feature percolation and/or for the creation of chains by means of index inheritance between a controller and a controllee, and in more complex cases, for instance in case of constituent ellipsis or deletion. Apart from that, the grammar implemented is a surface grammar of the chosen languages. Also functional Control mechanisms – both structural and lexical - have been implemented as close as possible to the original formulation, i.e. by binding an empty operator in the subject position of a propositional like open complement/predicative function, whose predicate is constituted by the lexical head.

Being a DCG, the parser is strictly a top-down, depth-first, one-stage parser with backtracking: differently from most principle-based parsers presented in Berwick et al. (1991), which are two-stage parsers, our parser computes its representations in one pass. This makes it psychologically more realistic. The final output of the parsing process is an f-structure which serves as input to the binding module and logical form: in other words, it constitutes the input to the semantic component to compute logical relations. In turn the binding module may add information as to pronominal elements present in the structure by assigning a controller/binder in case it is available, or else the pronominal expression will be available for discourse level anaphora resolution. As to the most important features of DCGs, we shall quote from Pereira and Warren (1980) conclusions, in a comparison with ATNs:

"Considered as practical tools for implementing language analysers, DCGs are in a real sense more powerful than ATNs, since, in a DCG, the structure returned from the analysis of a phrase may depend on items which have not yet been encountered in the course of parsing a
sentence. ... Also on the practical side, the greater clarity and modularity of DCGs is a vital aid in the actual development of systems of the size and complexity necessary for real natural language analysis. Because the DCG consists of small independent rules with a declarative reading, it is much easier to extend the system with new linguistic constructions, or to modify the kind of structures which are built. ... Finally, on the philosophical side, DCGs are significant because they potentially provide a common formalism for theoretical work and for writing efficient natural language systems.”(ibid,278).

2.2. Disambiguating Constituency with Functional Mapping

The parser is made up of separate modules:

i. The Grammar, based on DCGs, incorporates Extraposition to process Long Distance Dependencies, which works on annotated c-structures: these constitute the output to the Interpretation Module;

ii. The Interpretation Module checks whether f-structures may be associated to the input partially annotated c-structure by computing Functional Uniqueness, Coherence and Completeness. Semantic roles are associated to the input grammatical function labels at this level, after semantic selectional restrictions are checked for membership;

iii. The Mapping scheme, to translate trees into graphs, i.e. to map c-structures onto f-structures. The parser builds annotated c-structure, where the words of the input sentence are assigned syntactic constituency and functional annotations. This is then mapped onto f-structure, i.e. constituent information is dropped and DAGs are built in order to produce f-structure configuration.

Mapping into f-structure is a one-to-many operation: each major constituents may be associated with different functional values: this is why we activate grammatical function calls whenever possible in order to take into account the position of constituents to be built by the parser. This is particularly true for NPs, but can also be applied to other constituents as can be seen from the following discussion on constituent-grammatical function mapping:

a. NP --> SUBJect, both in preverbal and postverbal position - VP internally, VP adjoined and IP adjoined (see Delmonte, 1987) - with any kind of verbal category; OBJECT, usually in VP internal position, but also in preverbal position at Spec CP in case of reversed transitive structures; NCOMP predicative function - if not proper noun - occurring with copulative, and ECM verbs like "consider, believe"; closed ADJunct with [temporal] value, as the corresponding English example "this morning", which however in Italian can be freely inserted in sentence structure;

b. AP --> Modifier of an NP head, occurring as attribute in prenominal and as predication in postnominal position; ACOMP predicative function occurring with copulative, and ECM verbs; open XADJunct occurring freely at sentence level. Other examples of open adjuncts are: floating quantifiers, which however may only occur VP internally; doubling emphatic pronoun "lui" which also occurs VP internally and is computed as open adjunct;
c. AdvP --&gt; Open or closed Adjuncts according to its selectional properties, occurring anywhere in the sentence according to their semantic nature;
d. PP --&gt; OBLiques, when selected by a given predicate; PCOMP predicative function, when selected by a given predicate - both these two types of argument usually occur VP internally but may be fronted; open XADJunct or closed ADJunct according to semantic compatibility checks;
e. VP' --&gt; VCOMP infinitivals, when selected by a given predicate; SUBJect propositional clauses; closed ADJuncts with semantic markers like "for"; VP' gerundive and participial, which are always computed respectively as closed ADJuncts the former and as open ADJuncts the latter;
f. S' --&gt; CP as main clauses, or subordinate clauses, as well as sentential complements and SUBJect propositional clauses;
g. Clitics and Pronominal elements are also computed as NPs or PPs, because they are assigned grammatical functions when not associated to NP dislocation in preverbal position: in that case, the clitic is simply erased and TOPic function is associated with the binder NP.

2.3. Tracing c-structure Rules

The parser looks for syntactic constituents adjoined at CP level: in case of failure, it calls for IP level constituents, including the SUBJect which may either be a clause or an NP. This is repeated until it reaches the Verbal Phrase: from that moment onward, the syntactic category associated to the main verb - transitive, unergative, unaccusative, impersonal, atmospheric, raising, psych, copulative - and the lexical form of the predicate, are both used as topdown guidelines for the surface realization of its arguments. Italian is a language which allows for empty or morphologically unexpressed Subjects, so that no restriction may be projected from the lexicon onto c-structure: in case it is empty, a little pro is built in subject position, and features are left as empty variables until the tensed verb is processed.

The grammar is equipped with a lexicon containing a list of fully specified inflected word forms where each entry is followed by its lemma and a list of morphological features, organized in the form of attribute-value pairs. However, morphological analyzers for Italian and English are also available with big root dictionaries (90,000 for Italian, 25,000 for English) which only provide for syntactic subcategorization, though. The fully specified lexicon has been developed for Italian, English and German and contains approximately 10,000 entries for each language. In addition to that there are all lexical form provided by a fully revised version of COMLEX, and in order to take into account phrasal and adverbial verbal compound forms, we also use lexical entries made available by UPenn and TAG encoding. Their grammatical verbal syntactic codes have then been adapted to our formalism and is used to generate an approximate subcategorization scheme with an approximate aspectual and semantic class associated to it. Semantic inherent features for Out of Vocabulary Words, be they nouns, verbs, adjectives or adverbs, are provided by a fully revised version of WordNet in which we used labels similar to those provided by CoreLex.

Once the word has been recognized, lemmata are recovered by the parser in order to make available the lexical form associated to each predicate. Predicates are provided for all lexical categories, noun, verb, adjective and adverb and their description is a lexical form in
the sense of LFG. It is composed both of functional and semantic specifications for each
argument of the predicate: semantic selection is operated by means both of thematic role and
inherent semantic features or selectional restrictions. Moreover, in order to select adjuncts
appropriately at each level of constituency, semantic classes are added to more traditional
syntactic ones like transitive, unaccusative, reflexive and so on. Semantic classes are of two
kinds: the first class is related to extensionality vs intensionality, and is used to build
discourse relations mainly; the second class is meant to capture aspectual restrictions which
decide the appropriateness and adequacy of adjuncts, so that inappropriate ones are attached
at a higher level.

Grammatical functions are used to build f-structures and the processing of pronominals.
They are crucial in defining lexical control: as in Bresnan (1982), all predicative or open
functions are assigned a controller, lexically or structurally. Lexical control is directly
encoded in each predicate-argument structure, and in case shallow parsing does not make that
information available it will be impossible for the parser to bind the empty subject of all
predicative open functions built in all predicative structures (or small clauses) to the
appropriate syntactic controller (or binder). Grammar rules and structural configurations are
discussed at length in the following chapter.

2.4. Elliptical Structures

In a framework like this, all elliptical structures are left over at the end of grammar
traversal, simply because they cannot possibly be computed as any of the grammatically
complete sentence level analyses, either as main clauses, as complement clauses or as
subordinate or coordinate clauses. Just consider a simple case like the following:

(3) The nights must be cold and the days warm

which has been from a test text for NLUunderstanding distributed by Mitre – that we discuss in
some detail in Book 2. In order to compute the vp-ellipsis the rest of the previous
computation constituted by the string,

3.1 [and, the, days, warm]
must be evaluated in relation to the overall input sentence which is available from the
lookahead stack. This is done in order to check for some parallel pattern at the level of tag
assignment. This is also done in order to certify failures due to some form of agrammaticallity
present in the input sentence. When the parallelism has been ascertained, the main clause is
used in order to duplicate the governing elliptical verbal predicate and the rest of the sentence
is parser in its component constituents. This is done by accessing an iterative call which is
being used by all complements whenever a transitive verb has been detected or simply
whenever there is not enough information to decide on verb subcategorization frame. The
resulting list of constituents is then interpreted as in any normal non elliptical sentence by
adding all verb related syntactic and semantic information which is lacking in elliptical
sentences. The output will be a coordinate sentential structure which has the same verb
information as the main preceding clause.

The call to recover from failures with elliptical structures is also used in case of
ungrammatical structures with a feedback message being generated on the basis of the words
still to be processed. In one case we manage to recover from failure due to ambiguously computed constituents which however do not motivate any preliminary choice either from the tag disambiguation procedure or from parsing strategy. These are cases of adjunct PP or similar constructions which do not depend on lexical information for their interpretation. One example is a case of parasitic gap construction like the following,

(4) This is the kind of food that must be cooked before Mary eats.

In this example, “before Mary” will be computed as a PP which will then be appropriately interpreted as adjunct to the main verb “cook”. So the ending word “eats” will be left over to be filtered by the rule for elliptical constructions. This will be the trigger to recover the wrong analysis.

2.5. Parameters and Strategies

Here below is a short list of parameterized ambiguities: some of them are to be solved by parsing preferences which according to J.Fodor's latest work, are typological in nature. It appears that speakers of English prefer to adopt a Minimal Attachment strategy while this is not so per speakers of Romance languages. In particular, in the case of Relative Clause Attachment, this might be related to the influence of Latin on Romance language: Italian allows relative clauses as independent sentences to be attached in the discourse, just like Latin does.

A. Omissibility of Complementizer

• NP vs S complement
• S complement vs relative clause

B. Different levels of attachment for Adjuncts

• VP vs NP attachment of pp
• Low vs high attachment of relative clause

C. Alternation of Lexical Forms

• NP complement vs main clause subject

D. Ambiguity at the level of lexical category

• Main clause vs reduced relative clause
• NP vs S conjunction

E. Ambiguities due to language specific structural proprieties

• Preposition stranding
• Double Object
• Prenominal Modifiers
• Demonstrative-Complementizer Ambiguity
• Personal vs Possessive Pronoun

F. Clitic Pronouns

G. Aux-to-Comp
2.5.1. Linguistically-plausible Relaxation Techniques

With the grammar listed above and the parameters we are now in a position to establish apriori positions in the parser where there could be recovery out of loop with ungrammatical structures with the possibility to indicate which portion of the input sentence is responsible for the failure. At the same time, parsing strategies could be devised in such a way to ensure recovery from local failure. We will start by commenting on Parsing Strategies first and their implementation in our grammar.

The parser has been built to simulate the cognitive processes underlying the grammar of a language in use by a speaker, taking into account the psychological nuances related to the well-known problem of ambiguity, which is a pervading problem in real text/communicative situation, and it is regarded an inseparable benchmark of any serious parser of any language to cope with.

We also assume that a failure should ensue from strong Garden Path effects and that this should be justified at a psycholinguistic interpretation level (see Pitchett.

Differently from what is asserted by global or full paths approaches (see Schubert, 1984; Hobbs et al., 1992), we believe that decisions on structural ambiguity should be reached as soon as possible rather than deferred to a later level of representation. In particular, Schubert assumes "...a full paths approach in which not only complete phrases but also all incomplete phrases are fully integrated into (overlaid) parse trees dominating all of the text seen so far. Thus features and partial logical translations can be propagated and checked for consistency as early as possible, and alternatives chosen or discarded on the basis of all of the available information(ibid., 249)." And further on in the same paper, he proposes a system of numerical ‘potentials’ as a way of implementing preference trade-offs. "These potentials (or levels of activation) are assigned to nodes as a function of their syntactic/semantic/pragmatic structure and the preferred structures are those which lead to a globally high potential. Among contemporary syntactic parsing theories, the garden-path theory of sentence comprehension proposed by Frazier(1987a, b), Clifton & Ferreira (1989) among others, is the one that most closely represents our point of view. It works on the basis of a serial syntactic analyser, which is top-down, depth-first - i.e. it works on a single analysis hypothesis, as opposed to other theories which take all possible syntactic analysis in parallel and feed them to the semantic processor. From our perspective, it would seem that parsing strategies should be differentiated according to whether there are argument requirements or simply semantic compatibly evaluation for adjuncts. As soon as the main predicate or head is parsed, it makes available all lexical information in order to predict if possible the complement structure, or to guide the following analysis accordingly. As an additional remark, note that not all possible syntactic structure can lead to ambiguous interpretations: in other words, we need to consider only cases which are factually relevant also from the point of view of language dependent ambiguities.

The central goal of current theories of sentence comprehension is to identify the types of information people use when they hear or read sentences, and to identify the principles language users follow in using the information at their disposal. Adopting the hypothesis that knowledge of language is used in a "modular" fashion has led to substantial advances in our understanding of some of these principles. Meaning, discourse context, and world knowledge are generally thought to be used by modules other than the module(s) that use syntactic and lexical information.
We implemented in our parser a number of strategies that embody current intuitions on the way in which sentence comprehension mechanisms work at a psychological level. The parsing strategies are the following: Minimal Attachment/Late Closure (MA), Argument Preference (AP), Thematic Evaluation (TE), Referential Individuation (RI), Cross Compatibility Check (CCC). From the way in which we experimented them in our implementation it appears that they are strongly interwoven. In particular, MA is dependent upon AP to satisfy subcategorization requirements; with semantically biased sentences, MA and AP, and finally TE should apply in hierarchical order to license the phrase as argument or adjunct. RI seems to be required and activated every time a singular definite NP is computed. However, RI is a strategy that can only become operative whenever a full parse of possible modifiers is available and not before. In addition, subcategorization and thematic requirements have priority over referential identification of a given NP: a violation of the former is much stronger than the latter. Generally speaking, redundancies in referential properties might simply be accommodated by the speaker: but lack of consistency, uniqueness and completeness leads to ungrammaticality.

As discussed above, we follow a topdown depthfirst strategy which we believe better accounts for the way in which human psychological processes work. A drawback of this way of organizing parsing is backtracking actions that follow failure to complete the analysis of some constituent due to the presence of some ambiguity. In order to prevent failures and control backtracking, depthfirst analysis should be organized as much as possible deterministically. Nondeterminism can be very time consuming and it should be reduced or at least controlled according to the parsing strategy selected. So we assume that all possible linguistic information should be made available to the parser as soon as possible. In particular, subcategorization, semantic roles and all other semantic compatibility evaluating mechanisms should be active while parsing each word of the input string. We also assume that semantic role and selectional restrictions are dependent upon subcategorization information which is usually only available when a given head is parsed and not before.

Altmann(1989) offers a functional argument against a system in which choices are initially made by a syntactic processor, and later corrected by appeal to meaning and context. He says that if referential or discourse information is available, only a strange processor would make decisions without appealing to it. However, syntactic information is always available and always informative.

As the editor comments in his introduction (ibid.86), we also believe that it is an empirical question whether the constraints assumed by the thematic processor (single initial syntactic analysis, semantic evaluation only within the domain of this analysis) actually are constraints observed by the parser, or whether a less-constrained mechanism that makes appeal to context and meaning at the earliest stages of sentence comprehension is a more adequate description of the true state of affairs. It is our opinion that all lower level constraints should work concurrently with higher level ones: in other words, all strategies are nested one inside another, where MA occupies the most deeply nested level. The higher level strategy has control over the lower level one in case some failure is needed. Suppose we have the following examples which are disambiguated at the level of pronominal binding.

(5)i. The doctor called in the son of the pretty nurse who hurt herself.
ii. The doctor called in the son of the pretty nurse who hurt himself.
This level of computation takes place after f-structure has been completely checked and built in LFG - the same applies in GB framework, where S-structure gives way to L-structure and this is where binding takes place. In order to take pronominal binding into account, there should be a complete control over the previous levels of structural description so that a backtracking should be directed at the appropriate level of representation.

However, it is utterly inefficient and does not assure termination. We assume, instead, that RI should be activated in order to ascertain whether in the Discourse Model there is an entity which has the property of being hurt, associated to the individual "son" or to the individual "nurse". Relative clauses represent presupposed or known facts in case they are factual and these facts should be present in the DM. No syntactic and semantic constraints may be invoked in favour of one or the other interpretation: attachment of the relative adjunct is simply predictable from the DM – more on these two examples in a following chapter.

The following examples are all computed without any special provision for ambiguous structural position of the final temporal adverbial, simply by matching semantic information coming from verb tense and temporal configuration associated to the adverbial in its lexical entries in terms of a precedence relation between \( td \) (discourse time), and \( tr \) (reference time). Thus, in the case of "tomorrow" we shall have \( td < tr \) and the opposite will apply to "yesterday". In turn, this configuration is matched against tense, "past" or future" and a failure might result locally.

\( 6.\) (i). Mary will say that it rained yesterday.
     (ii). Mary said that it will rain yesterday.

3. TWO MECHANISMS AT WORK

In order to cope with the problem of recoverability of already built parses we built a more subtle mechanism that relies on Kay's basic ideas when conceiving his Chart (see Kay, 1980; Stock, 1989). Differently from Kay, however, we are only interested in a highly restricted topdown depthfirst parser which is optimized so as to incorporate all linguistically motivated predictable moves. Any already parsed NP/PP is deposited in a table lookup accessible from higher levels of analysis and consumed if needed. To implement this mechanism in our DCG parser, we assert the contents of the structure in a table lookup storage which is then accessed whenever there is an attempt on the part of the parser to build up a similar constituent. In order to match the input string with the content of the stored phrase, we implemented a WellFormed Substring Table(WFST) as suggested by Woods(1973).

Now consider the way in which a WFST copes with the problem of parsing ambiguous structure. It builds up a table of well-formed substrings or terms which are partial constituents indexed by a locus, a number corresponding to their starting position in the sentence and a length, which corresponds to the number of terminal symbols represented in the term. For our purposes, two terms are equivalent in case they have the same locus and the same length.

In this way, the parser would consume each word in the input string against the stored term, rather than against a newly built constituent. In fact, this would fit and suit completely the requirement of the parsing process which rather than looking for lexical information
associated to each word in the input string, only needs to consume the input words against a preparsed well-formed syntactic constituent.

Lookahead is used in a number of different ways: it may impose a wait-and-see policy on the topdown strategy or it may prevent following a certain rule path in case the stack does not support the first or even second match:

a. to prevent expanding a certain rule
b. to prevent backtracking from taking place by delaying retracting symbols from input stack until there is a high degree of confidence in the analysis of the current input string.

It can be used to gather positive or negative evidence about the presence of a certain symbol ahead: symbols to be tested against the input string may be more than one, and also the input word may be ambiguous among a number of symbols. Since in some cases we extend the lookahead mechanism to include two symbols and in one case even three symbols, possibilities become quite numerous.

Consider now failure and backtracking which ensues from it. Technically speaking, by means of lookahead we prevent local failures in that we do not allow the parser to access the lexicon where the input symbol would be matched against. It is also important to say that almost all our rules satisfy the efficiency requirement to have a preterminal in first position in their right-hand side. This is usually related to the property belonging to the class of Regular Languages. There are in fact some wellknown exceptions: simple declarative sentence rule, yes-no questions in Italian. Noun phrase main constituents have a multiple symbols lookahead, adjectival phrase has a double symbol lookahead, adverbial phrase has some special cases which require the match with a certain word/words like "time/times" for instance. Prepositional phrase requires a single symbol lookahead; relative clauses, interrogative clauses, complement clauses are all started by one or more symbols. Cases like complementizerless sentential complements are allowed to be analysed whenever a certain switch is activated.

Suppose we may now delimit failure to the general case that may be described as follows:

- a constituent has been fully built and interpreted but it is not appropriate for that level of attachment:
  failure would thus be caused only by semantic compatibility tests required for modifiers and adjuncts or lack of satisfaction of argument requirements for a given predicate.
  Technically speaking we have two main possibilities:
  A. the constituent built is displaced on a higher level after closing the one in which it was momentarily embedded.
  This is the case represented by the adjunct PP "in the night" in example that we repeat here below:

(7) The thieves stole the painting in the night.

The PP is at first analysed while building the NP "the painting in the night" which however is rejected after the PP semantic features are matched against the features of the governing head "painting". The PP is subsequently stored on the constituent storage (the WFST) and recovered at the VP level where it is taken as an adjunct.
B. the constituent built is needed on a lower level and there is no information on the attachment site.

In this case a lot of input string has already been consumed before failure takes place and the parser needs to backtrack a lot before constituents may be safely built and interpreted.

To give a simple example, suppose we have taken the PP "in the night" within the NP headed by the noun "painting". At this point, the lookahead stack would be set to the position in the input string that follows the last word "night". As a side-effect of failure in semantic compatibility evaluation within the NP, the PP "in the night" would be deposited in the backtrack WFST storage. The input string would be restored to the word "in", and analysis would be restarted at the VP level. In case no PP rule is met, the parser would continue with the input string trying to terminate its process successfully. However, as soon as a PP constituent is tried, the storage is accessed first, and in case of non emptiness its content recovered. No structure building would take place, and semantic compatibility would take place later on at sentence level. The parser would only execute the following actions:

- match the first input word with the (preposition) head of the stored term;
- accept new input words as long as the length of the stored term allows it by matching its length with the one computed on the basis of the input words.

As said above, the lookahead procedure is used both in presence and in absence of certain local requirements for preterminals, but always to confirm the current choice and prevent backtracking from taking place. As a general rule, one symbol is sufficient to take the right decision; however in some cases, more than one symbol is needed. In particular when building a NP, the head noun is taken at first by nominal premodifiers, which might precede the actual head noun of the NP. The procedure checks for the presence of a sequence of at least two nouns before consuming the current input token. In other cases the number of preterminals to be checked is three, and there is no way to apply a wait-and-see policy.

Reanalysis of a clause results in a Garden Path(GP) in our parser because nothing is available to recover a failure that encompasses clause level reconstruction: we assume that GP obliges the human processor to dummy all naturally available parsing mechanisms, like for instance lookahead, and to proceed by a process of trial-and-error to reconstruct the previously built structure in order not to fall into the same mistake. The same applies to our case which involves interaction between two separate modules of the grammar.

In relation to the following highly ambiguous example taken from a legal text:

(8) Producer means the manufacturer of a finished product, the producer of any raw material or the manufacturer of a component part and any person who by putting his name, trade mark or other distinguishing feature on the product presents himself as its producer.

In more detail, suppose we have to use the information that "put" is a verb which requires an oblique PP be present lexically in the structure, as results from a check in its lexical form. We take the verb in I position and then open the VP complement structure, which at first builds a NP in coincidence with "the book". However, while still in the NP structure rules, after the head has been taken, a PP is an option freely available as adjunct.

We have implemented two lookahead based mechanisms which are used in the PP building rule and are always triggered, be it from a position where we have a noun as head and we already built part of the corresponding constituent structure; be it from a position
where we have a verb as head and we want to decide whether our PP will be adequate as argument rather than as adjunct - in the latter case it will become part of the Adjunct Set.

The first one is called,

**Cross Compatibility Check (CCC)**

This mechanism requires the head semantic features or inherent features to be checked against the preposition, which in turn activates a number of possible semantic roles for which it constitutes an adequate semantic marker. For instance, the preposition "on" is an adequate semantic marker for "locative" semantic role, this will cause the compatibility check to require the presence in the governing heading of inherent or semantic features that allow for location. A predicate like "dress" is computed as an object which can be assigned a spatial location, on the contrary a predicate like "want" is computed as a subjective intensional predicate which does not require a spatial location. However, in order to take the right decision, the CCC must be equipped with the second mechanism we implemented;

The second one is called,

**Argument Precedence (AP)**

The second mechanism we implemented allows the parser to satisfy the subcategorization requirements in any NP constituent it finds itself at a given moment if the parsing process. Suppose that after taking "put" as the main verb, this mechanism is activated, by simply copying the requirements on PP oblique locative present in the lexical form associated with the predicate "put" in the lexicon, in the AP. As soon as the NP "the book" is opened, after taking "book" as N at the head position, the parser will meet the word "on", which allows for a PP adjunct. While in the P head position, the parser will fire the CCC mechanism first to see whether the preposition is semantically compatible, and in case it is, the second AP mechanism will be fired. This will cause the system to do the following steps:

i. check whether the requirements are empty or not;
ii. and in case it is instantiated, to control the semantic role associated with it;
iii. to verify whether the P head is a possible semantic marker for that semantic role: in our case, "on" is a possible semantic marker for "locative" semantic role;
iv. finally to cause the parser to fail on P as head of a PP adjunct of the head noun;
v. produce a closure of NP which obeys Minimal Attachment principle.

### 3.1. PP Attachment and Parsing

Consider now the wellknown problem of PP attachment or Syntactic Closure (see Delmonte, 1984), which concerns the way in which Prepositional Phrases modifiers or arguments of a given lexical head should be dealt with in a parsing scheme. In the literature we have two different proposals: a grammatical one called "garden-path" (see Frazer) and a purely semantic one, called "incremental-interactive" (see Steedman & Altmann). There is a crucial difference between the two theories: whereas garden-path theory claims that a single analysis of a syntactic structure ambiguity is initially constructed and passed to the semantic
module, incremental-interactive theory claims that the syntactic module offers all grammatical alternatives to the semantic one, in parallel, to be evaluated. There is evidence that is favourable to both sides on this issue.

The basic claims of the garden-path theory are that (a) the sentence processing mechanism (the parser) uses a portion of its grammatical knowledge, isolated from world knowledge and other information, in initially identifying the relationships among the phrases of a sentence; (b) the parser is sometimes confronted with an uncertain choice in how to relate a new phrase to the existing structure, and when it is, it initially commits itself to a single structure; and (c) the parser follows a very general and psychologically motivated principle in choosing the structure it builds. The principle is "choose the first available analysis". This principle is realized as the parsing strategies of Minimal Attachment ("do not postulate any unnecessary nodes" or, equivalently for present purposes, "use each incoming word into the structure currently being built") and Late Closure ("if consistent with the rules of the grammar, attach each incoming word into the phrase currently being analysed").

The incremental-interactive theory permits a far more intimate and elaborate interaction between the syntax and the semantic/referential modules; we are only interested in the grammatically based one, which we illustrate briefly here below:

A. The Garden Path (Gp) proposal has the following main features (see Kennedy et al, 1989):
   - choose the first available analysis, or words in a sentence are incorporated into complete syntactic structures at the earliest possible opportunity (i.e. the structural description is developed "on-line", word-by-word);
   - do not postulate any unnecessary node - attachment of words within a structure is invariably achieved in a way which minimises the number of nodes demanded (the principle of Minimal Attachment - MA);
   - if consistent with the rules of the grammar, attach each incoming word into the phrase currently being analyzed - or new words are incorporated, wherever this is possible grammatically, into the current clause or phrase being processed (the principle of Late Closure - LC).

In a typical minimal pair example such as the following,

(9)a. She positioned the dress on the rack
   b. She wanted the dress on the rack

the two principles mentioned above, MA and LC make conflicting predictions: in a. MA predicts that the PP "on the rack" shall not be assigned as adjunct of the head "dress" but as locative oblique argument of the main verb "position", thus complying with the general criteria of economicity and psychological efficiency based on grammatical issues; however the principle LC would make just the opposite prediction, since there is no grammatically motivated criteria to prevent the PP to be computed locally as a semantically compatible adjunct of the head "dress". In version b. the two principles will make just the opposite predictions, so that MA would wrongly predict that the PP attaches to the main verb as a locative adjunct or oblique argument, whereas only LC would apply correctly.

It is a fact that examples such as these are genuinely ambiguous cases and there is no way to state general principle which could apply to both equally well, producing the most efficient result. Even if we try to maximixe on the fact that in a. we already know what the Verb Guidance (see Mitchell, 1989) is, i.e. we expect an oblique PP to be present somewhere in the following structure, we still need to take a stance as to whether we rely on a
- selecting procedure, or structure assembly process;
- monitor procedure, or structure checking process;
where the second one may involve rejecting the initial structure, backtracking and reassembling a new structure (see Mitchell, 126).

3.2. Some Examples

In the texts reported below we give empirical evidence for the need to use lexical information in order to reduce parsing loads resulting from backtracking procedures: we mark decision points with a bar,


At the first boundary we have "of" which is non semantically marked and no prediction is available, so that the default decision is to apply Late Closure, which turns out to be the correct one. When the second preposition is found we are in the NP of the PP headed by "of", and we have taken the date "1985": this will cause the CCC to prevent the acceptance of the preposition "on" as a semantically compatible marker thus preventing the construction of the NP headed by "approximation".

Notice, that in case that would be allowed, the NP would encompass all the following PPs thus building a very heavy NP: "the approximation of the laws, regulations and administrative provisions of the Member States concerning liability for defective products". In case the parser had a structure monitoring strategy all this work would have to be undone and backtracking would have to be performed. Remember that the system does not possibly know where and how to end backtracking unless by trying all possible available combination along the path. In our case, the presence of a coordinate structure would render the overall process of structure recoverability absolutely untenable.

Another important decision has be taken at the boundary constituted by the participial head "concerning": in this case the CCC will take the inherent features of the head "States" and check them with the selection restrictions associated in the lexical form for the verb "concern". Failure in this match will cause the NP "the Member States" to be closed and will allow the adjunct to be attached higher up with the coordinated head "laws, regulations and administrative provisions". In this case, all the inherent features are collected in a set that subsumes them all and can be used to fire CCC.

Notice that the preposition "for" is lexically restricted in our representation for the noun "liability", and the corresponding PP that "for" heads interpreted as a complement rather than as an adjunct. We include here below the relevant portion of each utterance in which the two mechanisms we proposed can be usefully seen at work. We marked with a slash the place in the input text in which, usually when the current constituent is a NP a decision must be taken as to whether causing the parser to close (MA) or to accept more text (LC) is actually dependent upon the presence of some local trigger. This trigger is mostly a preposition; however, there are cases in which, see (11), (12), (14), (15), the trigger is a conjunction or a participle introducing a reduced relative clause. Coordinate NPs are a big source of indecision and are very hard to be detected if based solely on syntactic, lexical and semantic information. For instance, e. can be thus disambiguated, but h. requires a matching of
prepositions; In the case represented by (15) we put a boundary just before a comma: in case the following NP "the Member State" is computed as a coordination - which is both semantically, syntactically and lexically possible, the following sentence will be deprived of its lexical SUBJect NP - in this case, the grammar activates a monitoring procedure independently so that backtracking will ensue, the coordinate NP destroyed and the comma computed as part of the embedded parenthetical (which is in turn an hypothetical within a subordinate clause!!). Notice also that a decision must be taken in relation to the absolutes headed by a past participle which can be intended as an active or a passive past participle: in the second case the head noun would have to be computed as an OBJect and not as a SUBJect. The following examples are small fragment from biggest sentences which are use to enforce our point:

(11) a differing degree of protection of the consumer | against damage caused by a defective product | to his health or property
(12) in all member states | by adequate special rules, it has been possible to exclude damage of this type | from the scope of this directive
(13) to claim full compensation for the damage | from any one of them
(14) the manufacturer of a finished product, the producer of any raw material or the manufacturer of a component part | and any person
(15) The liability of the producer | arising from this directive
(16) any person who imports into the community a product | for sale, hire or any form of distribution | in the course of his business
(17) both by a defect in the product | and by the fault of the injured person
(18) However, if... the commission does not advise the Member State | concerned that it intends submitting such a proposal | to the council |, the Member State

3.3. Principles of Sound Parsing

- Principle One: Do not perform any unnecessary action that may overload the parsing process: follow the Strategy of Minimal Attachment;
- Principle Two: Consume input string in accordance with look-ahead suggestions and analyse incoming material obeying the Strategy Argument Preference;
- Principle Three: Before constructing a new constituent, check the storage of WellFormed Substring Table(WFST). Store constituents as soon as they are parsed on a stack organized as a WFST;
- Principle Four: Interpret each main constituent satisfying closer ties first - predicate-argument relations - and looser ties next - open/closed adjuncts as soon as possible, according to the Strategy of Functional Preference;
- Principle Five: Erase short-memory stack as soon as possible, i.e. whenever clausal constituents receive Full Interpretation.
- Strategy Functional Preference: whenever possible try to satisfy requirements posed by predicate-argument structure of the main governing predicate as embodied in the above Principles; then perform semantic compatibility checks for adjunct acceptability.
- Strategy Minimal Attachment: whenever Functional Preference allows it apply a Minimal Attachment Strategy.
The results derived from the application of Principle Four are strictly linked to the grammatical theory we adopt, but they are also the most natural ones: it appears very reasonable to assume that arguments must be interpreted before adjuncts can be, and that in order to interpret major constituents as arguments of some predicate we need to have completed clause level structure. In turn adjuncts need to be interpreted in relation both to clause level properties like negation, tense, aspect, mood, possible subordinators, and to arguments of the governing predicate in case they are to be interpreted as open adjuncts.

As a straightforward consequence, owing to Principle Five we have that reanalysis of a clause results in a Garden Path (GP) simply because nothing is available to recover a failure that encompasses clause level reconstruction: we take that GP obliges the human processor to dummify all naturally available parsing mechanisms, like for instance look-ahead, and to proceed by a process of trial-and-error to reconstruct the previously built structure in order not to fall into the same mistake.

4. TWO STRATEGIES IN COMPETITION

Let us turn to a discussion of the empirical evidence bearing on how context is used in sentence processing. Does it strongly support the claims of a parallel, interactive, incremental processor, as Altmann and Steedman (1989) suggest? Or is it more consistent with the serial, garden-path processor?

Even if one reaches the conclusion that some semantic factors can block the construction of some syntactic analyses, we do agree with A&S that referential and discourse information is used in making initial decisions.

The list of examples here below includes sentences used in the literature to support one or the other of the two parsing theories, garden-path and incremental-interactive theory (see Altman (ed), 1989). As will be shown below, the examples cannot be computed only by activating or not Minimal Attachment strategy.

(19) Mary put the book on the table
(20) Mary put the book on the table in her bag
(21) Mary saw the cop with the the binoculars
(22) Mary saw the cop with the the revolver
(23) The thief stole the painting in the museum
(24) The thief stole the painting in the night

A&S note that several of the ambiguities present in these sentences and resolved by Minimal Attachment involve contrasts between NP modification and other structures. However, (20), (21) and (23) require some more knowledge, linguistic one.

In example 1 we assume that the PP should be computed as an argument of the main predicate "put", thus following a MA strategy when parsing the NP "the book". However, the same strategy would lead to a complete failure in (20), where the PP should be taken as NP modifier, thus responding to a Late Closure strategy.
If we look at subcategorization requirements of the verb, example (19) constitutes a clear case of Verb Guidance and of semantic role satisfaction requirements: the main predicate requires an argument which is a locative, so at every decision point in which a PP might be taken, argument requirements should be accessed and a MA strategy imposed locally. In the following examples, argument structure plays no role whatsoever: instrumentals, comitatives, locatives with predicate "steal" are all cases of sentential adjuncts.

Differently from what has been discussed in the literature (see Altman, ed.), we assume that in (21) the instrumental is not an argument of the predicate "see", for the following reasons:

- only locatives may be computed as open complements (see Chapter III), in case their semantic role is not that of provenance or origin, as the following two examples show:

(25) John saw Mary in the kitchen
(26) John saw Mary from the bathroom

In (25) we understand that the location at which Mary was when the seeing event took place is the kitchen: we also understand that John might have been in the same location or in a different one, already provided by the previous context.

In (26), on the contrary, we understand that the location from which the seeing event took place is the bathroom and that John was certainly there; however we are given no information whatsoever about Mary's location.

However, in (21) and (22), the first decision must be taken when computing NP structure. In fact, a PP headed by preposition "with" is a semantically compatible NP modifier - a comitative - and the analysis should be allowed to continue until the PP is fully analysed. In other words, Late Closure might apply. Also MA could successfully apply to (21) with the result of computing the "with" PP as Instrumental adjunct. This procedure would succeed only in (21) and would fail in (22), where the PP can only be interpreted as NP modifier.

Finally, consider (23) and (24). Head preposition "in" constitutes a viable local PP modifier and there are no argument requirements from the main verb predicate. However, "in the night" is not a possible NP modifier and (25) is a clear case of minimal attachment sentence. On the contrary, (24) is ambiguous.

### 4.1. Three Different Activation Processes of Parsing Strategies

From our perspective, it would seem that parsing strategies should be differentiated according to whether there are argument requirements or simply semantic compatibility evaluation for adjuncts:

- Mode A: the main verb predicate projects its argument requirements onto the following NP structure building procedure - in case the predicate is a transitive verb;
- Mode B: there are no argument requirements and the head preposition may lead to a local PP attachment which is semantically compatible with the head noun.
- Mode C: in presence of semantic compatibility, an adjunct may be attached at different structural levels according to referential properties of the governing head.
When we are in the conditions to apply Mode A, a failure might also result seen that the adjunct "in her bag" does not make sense as sentence modifier - the putting event cannot take place in the bag!! In the case of examples (21) and (22), we should only prevent our parser from computing "with revolver" as sentence adjunct, because it could never be the case that revolvers may constitute adequate instruments for seeing events.

Obviously the same line of reasoning can be applied whenever we find ourselves without any argument requirement related to PP's as is the case with verbs like "steal".

As an additional remark, note that not all possible syntactic structure can lead to ambiguous interpretations: in other words, we need to consider only cases which are factually relevant also from the point of view of language dependent ambiguities.

Example (20) is an interesting case for explaining Mode C: the two final PP's have the same structure and the same semantic role, both can satisfy TE and AP, however MA should be made to fail in accordance with RI which checks for an entity "book" with the property of being on the "table". In this way, "in her bag" would be computed as Oblique/pp argument of "put".

4.2. Implementing Parsing Strategies

So we are left with five strategies: Minimal Attachment, Arguments Preference, Thematic Evaluation, Referential Individuation, Cross Compatibility Check, where the first one is just a completely syntactically driven strategy, and the remaining ones work at the interpretation level and by taking advantage of all subcategorization and semantic information available.

A. Minimal Attachment

A MA procedure is set up by blocking the NP structure building process as soon as the head noun is taken, i.e. after "cop" has been analysed in examples 3 and 4, and by disregarding entirely the following words of the input string. It is also activated whenever the verb requires a sentential complement or an oblique PP argument.

This strategy is also required in order to parse adequately genuinely ambiguous structures which do not have any specific requirements guided by verb's arguments', but only adjuncts' semantic adequacy requirements as for instance in (21-25) above. Since both "with the binoculars" and "with the revolver" can be construed as NP internal modifiers, only a Minimal Attachment strategy would prevent the parser from taking the PP while computing NP structure.

Obviously, when this strategy is activated example 4 would fail since the PP "with the revolver" would not be possibly computed as a higher sentential adjunct. Differently from what happens with the remaining examples.

In example (21), which we repeat here below,

(21) Mary saw the cop with the binoculars

the verb see allows for an open complement which however should be locative. This information is recovered from the prepositional head of the PP which is not an adequate head for locative complements. In other words, we use semantic role information to select the appropriate PP head at a syntactic level. So the PP headed by "with" is not taken as an open complement but is already recorded on the PP stack and recovered in the subsequent processing steps.
B. Strategy: Arg-Pref, Argument Preference

Consider the two sentential contexts: Mary put | the book on... / Mary promised | the man that...

The strategy requires the parser to look for arguments of a verb predicate, Oblique/PP and Propositional/s’, in order to project this information inside the Object NP rule. After the NP head has been taken, all following constituents will carry the information that expectations about some uninstantiated argument must be computed prioritarily in case such constituents are present in the input string. The constituent trigger in case of Obliques is the head preposition: the parser will take the preposition, then check in the argument list of the governing verb whether it matches the preposition contained in the lexical form and generate a local failure so that no PP modifier will be built inside the current NP. The same will apply for the trigger of sentential complements which is the complementizer, and this will also be encountered when trying to build a relative clause modifier inside the NP. As happens with obliques, when the parser is analysing the input sequence within the NP and is trying to build a relative clause it will look inside the argument information storage copied from the list of arguments of the governing verbal predicate contained in its lexical form, and in case a sentential propositional complement is present then a local failure will be produced and no relative clause will be built.

Consider that the word "that" is ambiguous between a number of lexical categories, which include, deictic pronoun, deictic adjective, complementizer. The parser should be able to predict the appropriate structural choice in many grammar rules, basically by using lookahead. Here below we give parsing times on a Mac G4 for example 1.

[Mary put the book on the table]

No parsing strategy activated, the PP is taken as adjunct in the NP constituent and the oblique argument is left indefinite.

Time: 0.13 sec.
Minimal Attachment activated.
Time: 0.087 sec.
Argument Preference activated.
Time: 0.105 sec.

C. Strategy: Fill-loc-arg, Fill Locative Argument

Fill Locative Argument activated.
Time: 0.28 sec.

No information is made available in our lexical forms for obligatoriness of oblique arguments. All of them may be left indefinite in case no input is found. So, we have built a parsing strategy that allows for the obligatory search of an oblique argument which is semantically interpreted as a locative. When this strategy is activated, the parser will fail the analysis at the interpretation level - i.e. when building f-structure - whenever no oblique is found - rather than accepting the incomplete input sentence and generating an indefinite argument label. This strategy only checks for grammatical completeness whenever all constituents have been built and the input sentence is fully consumed. In accordance with LFG grammatical principles of uniqueness, completeness and coherence, the parser produces a full parse and then checks for consistency.

The activation of this strategy may then result in backtracking in case the oblique was present in the input sentence but was taken as a PP adjunct or modifier of the object NP. On the contrary, the activation of the Arg-Pref strategy will be triggered right inside the object
NP in case there is such a constituent in the input string and no backtracking will apply. As can be seen from parsing time, semantic strategy based on completeness of semantic roles is the more time-consuming strategy.

4.3. Solving Ambiguities with Lookahead and Backtrack Storage

We implemented two simple enough mechanisms in order to cope with the problem of nondeterminism and backtracking. At bootstrapping we have a preparsing phase where we do lexical lookup and we look for morphological information: at this level of analysis of all input tokenized words, we create a stack of pairs input wordform - set of preterminal categories, where preterminal categories are a proper subset of all lexical categories which are actually contained in our lexicon. The idea is simply to prevent attempting the construction of a major constituent unless the first entry symbol is well qualified. When consuming any input wordform, we remove the corresponding pair on top of stack.

However, erasing one pair from stack top is a destructive non recoverable operation: we assert the lookahead stack and consequently we retract symbol pairs whenever the parser succeeds in its analysis. Problems may arise then any time a failure occurs and backtracking should restore previously consumed input: the lookahead stack cannot be recovered since the retract operation in Prolog is destructive.

In order to cope with this problem and in general with the problem of recoverability of already built parses we built a more subtle mechanism that relies on Kay's basic ideas when conceiving his Chart (see Kay, 1980; Stock, 1989). Differently from Kay, however, we are only interested in a highly restricted topdown depth-first parser which is optimized so as to incorporate all linguistically motivated predictable moves.

As we commented above, from a general point of view, we should distinguish between cases of ambiguity in which the parser may use argument requirements on the following analysis, and cases in which this does not apply. In this latter case, we are left with semantic adequacy evaluation to be performed between the constituent's head and the governing head. In case this leads to a subsequent failure, as would happen in examples (20), (23) and (24), the parser should not backtrack. The already parsed PP, should be deposited in a table lookup storage which is then accessed whenever there is an attempt on the part of the parser to build up a PP. In order to match the input string with the content of the store phrase, we implemented a WellFormed Substring Table (WFST) as suggested by Woods (1973).

Now consider the way in which a WFST copes with the problem of parsing ambiguous structure in his chart. He builds up a table of well-formed substrings or terms which are partial constituents indexed by a locus, a number corresponding to their starting position in the sentence and a length, which corresponds to the number of terminal symbols represented in a term. For our purposes, two terms are equivalent in case they have the same locus and the same length.

In this way, the parser would consume each word in the input string against the stored term, rather than against a newly built constituent. In fact, this would fit and suit completely the requirement of the parsing process which rather than looking for lexical information
associated to each word in the input string, only needs to consume the input words against a
preparsed well-formed syntactic constituent.

To give a simple example, suppose we have taken the PP "in the night" within the NP
headed by the noun "painting". At this point, the lookahead stack would be set to the position
in the input string that follows the last word "night". As a side-effect of failure in semantic
compatibility evaluation within the NP, the PP "in the night" would be deposited in the
backtrack storage. The input string would be restored to the word "in", and analysis would be
restarted at the VP level. In case no PP rule is met, the parser would continue with the input
string trying to terminate its process successfully. However, as soon as a PP constituent is
tried, the storage is accessed first, and in case of non emptiness its content recovered. No
structure building would take place, and semantic compatibility would take place later on at
sentence level. The parser would only execute the following actions:
- match the first input word with the (preposition) head of the stored term;
- accept new input words as long as the length of the stored term allows it by matching its
length with the one computed on the basis of the input words.

So the PP storage is a stack in which PP's are stored as soon as they are parsed in order to
prevent backtracking from taking place, and used in higher level of analysis in case some
failure takes place. Differently from the lookahead stack, the PP storage stack should not be
destroyed as phrases are consumed at some level of analysis since there is no way to be sure
that the result of that given choice be successful until both c-structure and f-structure building
have been completed.

4.4. Lookahead and Computation

Pritchett(1992) believes computational properties of the parser are irrelevant, they may
simply contribute a more efficient implementation and nothing more; they do not modify nor
decide what parsing strategy should be better than another simply because strategies are
decided at a grammatical linguistic level. However, even though she assumes that her ideas
are implementable I have some doubts about it. In particular, in relation to Object - Subject
Ambiguity she describes the disambiguating procedure as being triggered by the fact that the
parser detects the lack of a NP at subject position when computing the VP structure for a
subjectless verb, a configurational requirement of English, and then by means of the Steal-NP
strategy recovers the NP without problems for reanalysis.

Differences in reanalysis are determined by structural requirements and by analysis load
imposed on the parser by backtracking: in case a sentential adjunct has to be destroyed/broken
up and reconstructed it represents a far lighter load than a subordinate/main clause as is the
case in real Gps. Let’s say, that whenever a clausal structure has to be destroyed/broken up a
whole set of semantic decisions have to be dismantled, and structure erased.

Consider the following example taken from one of our texts,

(27) Al seguito di Alberti, che era diventato vicepresidente del senato, Franco Avveduti
nello immediato dopoguerra si trasferì a Roma.

which might be translated roughly as follows,

"At the suite of Alberti, who has become vicepresident of the Senate, Franco Avveduti in
the immediate post-war transferred himself to Rome".
The preposed adjunct PP, modified by a nonrestrictive relative clause, is followed by a noun phrase "Franco Avveduti" which is tentatively computed as apposition of the noun "Senate"; in turn, the following PP headed by "in" is also computed in the wrong place, after taking "Franco Avveduti". In other words, before starting to close any constituent and to interpret it, the parser is situated on the verb "transferred" which causes a local failure, the closing of the PP "in the immediate post-war", and the attempt at interpreting it as adjunct modifier of "Avveduti" a proper noun which is the nonlocal failure causing backtracking.

Consider now what has happened on the lookahead stack: the parser has taken all input symbols, 16 actually, and is now looking at the 17th word "transferred" which is a "verb - v" - in the original version the clitic pronoun "si" also counts as a "v". When backtracking, the length of the first constituent is correctly computed against the lookahead symbol which is still available on top of stack. However, all remaining constituents need the information locally both of starting place and ending place in the input string in order to compute their length.

Consider also the fact that in this situation, backtracing takes the parser back in the input string only up to the point in which there has been a wrong attachment, i.e. after the second comma at the end of the nonrestrictive relative clause. This is so simply because the remaining part of the analysis is correctly interpreted in the position in which it has been analysed as clause initial adjunct PP.

As a result of this situation, there is a mismatch between the input string position of the parser - in front of "Franco", the position of the lookahead stack - as follows, "18-[v-transferred]" - and the constituents on stack for wellformed substrings, which has all the NP's and PP's intervening between the verb "transferred" and the first word in the input string "at". Each constituent is built as a term which has the first word as functor or predicate and in its internal representation it has the preterminal symbol associated with the terminal on the stack, then a number indicating the position in the string, another number indicating the length of the constituent built, and finally the constituent itself. The input string is internally represented schematically as follows:

[al, seguito, di, alberti, ',', che, era, diventato, vicepresidente, del, senato, ',', franco, avveduti, nello, immediato, dopoguerra, si, trasferi, a, roma, .]

**LOOKAHEAD-STACK**


**INPUT TERMINAL SYMBOLS**

- a(3, 2-pp(PP))
- il(2, 2-np(NP))
- di(2, 4-pp(PP))
- alberti(1, 4-np(NP))
- vicepresidente(1,9-np(NP))
- di(2,11-pp(PP))
- il(2, 11-np(NP))
I shall comment now on some of the main procedures used to implement lookahead in our parser. This procedure is used both in presence of certain local requirements for preterminals and in absence, but always to confirm the current choice and prevent backtracking from taking place. In particular when building a NP, the head noun is taken at first by nominal premodifiers, which might precede the actual head noun of the NP. The procedure checks for the presence of a sequence of at least two nouns before consuming the current input token.

\[
doublelook(Mc,Y,Z):-
\{\text{doublelook(sn_mod,n,n)}\},
\text{head(Head,Gen,Num,Pers,Case,Type)}.\]

In this case the number of preterminals to be checked is three, and there is no way to apply a wait-and-see policy, since adverbial phrases are computed in a separate rule. Besides, to be confident about the current rule choice we need to have a past participle to complete the rule of the compound verb.

\[
triplelook(Mc,Y,Z,Q):-
\text{verb_lex_nf(Ainf,LMood,Vinf,Mood,Tense,Pers,Num,Gen,CatGr1,CatV,LArgsV,Adv)}
\text{-->
\{on(Mood,ListMood)\},
\text{auxiliary(aux(Ainf,Mood,Tense,Pers,Num))},
\{triplelook(verb_lex_nf,v,av,q)\},
\text{retraction(auxiliary, termin(S-Z))},
\text{sint_adv(Tense,Adv,_)},
\text{participle(Vinf,Gen1,Num1)}.\]

The following procedure is used to disambiguato multiple preterminals assigned to the same word in order to prevent backtracking. The procedure specifies a set of possible disambiguating categories (Cats) which might follow the current symbol (Y). This procedure is used for an Italian word like "decise" which might be computed both as a verb, thus meaning "decided", past tense, third person singular, or past participle, plural feminine; or else as an adjective, plural feminine.

\[
disambiguate_tok(Y,Cats):-
\text{agg(Agg,Gen,Num)} --\rightarrow \{X\},
\{gr(romance),agg_x(X,Agg,Gen,Num),
\text{disambiguate_tok(a,[p,c])},
\text{retraction(agg_x,termin(S-Z))}.\]

This is used for a special parsing strategy: it takes the current terminal on top of stack and matches it with a list of words which are either taken from the predicate-argument list or are locally indicated. It will prevent parsing from continuing in that rule. In the first case, it is
used to activate minimal attachment on the basis of subcategorization information: the current terminal symbol (Head) should be a preposition (Y), and it should not belong to the set of prepositions passed by the lexical form of the current governing predicate.

In the second implementation of the predicate we have a similar problem: the governing predicate waits for a sentential complement but the complementizer is ambiguous with a deictic pronoun/adjective.

The information gathered from the search on the top of stack also makes available the current location which is then used to compute the length of the PP constituent to be written on the PP stack.

After having analyzed a modal verb we look for an auxiliary verb in order to support the rule for compound auxiliary which makes up a passive verb: retraction of the modal takes place after the auxiliary is found on the stack. Otherwise backtracking is allowed.

4.5. Graceful Recovery Actions from Failures

As we discussed above, recovery from garden-path requires a trial and error procedure, i.e. the parser at first has to fail in order to simulate the garden-path effect and then the recovery will take place at certain conditions.

Now consider the well-known case of Reduced Relatives which have always been treated as a tough case (but see Stevenson & Merlo, 1997). From an empirical point of view we should at first distinguish cases of subject attachment reduced relatives from all other cases, because it is only with subject attachment that a garden-path will actually ensue. This is easily controllable in our parser given the fact that NPs are computed by means of functional calls. In this way the information as to where the NP is situated in the current sentence analysis is simply a variable that is filled with one of the following labels: subj, obj, obj2, obl, adj, ncomp, where the last label stands for predicative open complements. Again from a purely empirical point of view, we also visited the WSJ corpus in order to detect cases of subject attachment vs all other cases for reduced relatives and we came up with the following figures:

\[
\text{SUBJect-Attachment} \quad 530 \quad + \quad \text{Others} \quad 2982 \quad = \quad \text{Total} \quad 3512
\]

From the total number we must subtract present participle cases of reduced relatives which do not constitute ambiguous words: the total number is lowered down to 340. Subject-attachment thus constitute the 9.68% of all cases, a certainly negligible percentage. In addition, 214 of all subject-attachment are passive participles and lend themselves to easy computation being followed by the preposition “by”. So there will reasonably be only 116 possible candidates for ambiguous reduced relatives. The final percentage comes down 3.3% which is very low in general, and in particular when computed over the whole 1 million occurrences, it comes down to a non classifiable 0.0116%. The same results can be obtained from an investigation of the Susanne Corpus, where we found 38 overall cases of reduced relatives with ambiguous past participles, 0.031% which is comparable to the 0.035% of the WSJ.

If we look into matter closely, then we come up with another fairly sensible and easily intuitive notion for reduced relatives disambiguation: and it is the fact that whenever the
governing Noun is not an agentive, nor a proto-agent in any sense of the definition (see Stevenson & Merlo, 1997), no ambiguity may arise simply because non agentive nominal governors may end up with an ambiguous interpretation only in case the verb is used as ergative. However, not all transitive verbs can be made ergatives and in particular none of the verbs used in WSJ in subject-attachment for reduced relatives can be ergativized apart from “sell”. We report here below verb-types, i.e. verb wordforms taken only once. As can be easily seen none of the verbs are unergative nor unaccusatives.

Table 23. List of 27 verb-types used in WSJ in subject-attached reduced relatives

<table>
<thead>
<tr>
<th>accused</th>
<th>afforded</th>
<th>based</th>
<th>boosted</th>
</tr>
</thead>
<tbody>
<tr>
<td>bought</td>
<td>canceled</td>
<td>caught</td>
<td>caused</td>
</tr>
<tr>
<td>completed</td>
<td>contacted</td>
<td>derived</td>
<td>designed</td>
</tr>
<tr>
<td>filed</td>
<td>honed</td>
<td>involved</td>
<td>led</td>
</tr>
<tr>
<td>listed</td>
<td>made</td>
<td>managed</td>
<td>owned</td>
</tr>
<tr>
<td>paid</td>
<td>purchased</td>
<td>related</td>
<td>represented</td>
</tr>
<tr>
<td>requested</td>
<td>sold</td>
<td>unsettled</td>
<td></td>
</tr>
</tbody>
</table>

If we look at the list of 36 verb-types used in Susanne Corpus we come up with a slightly different and much richer picture:

Table 24. List of 36 verb-types used in SUSANNE in subject-attached reduced relatives

<table>
<thead>
<tr>
<th>altered</th>
<th>become</th>
<th>bent</th>
<th>burned</th>
</tr>
</thead>
<tbody>
<tr>
<td>charged</td>
<td>clouded</td>
<td>compared</td>
<td>cooled</td>
</tr>
<tr>
<td>cut</td>
<td>deserted</td>
<td>distilled</td>
<td>dominated</td>
</tr>
<tr>
<td>estimated</td>
<td>fed</td>
<td>figured</td>
<td>filmed</td>
</tr>
<tr>
<td>focused</td>
<td>frozen</td>
<td>internalized</td>
<td>intertwined</td>
</tr>
<tr>
<td>known</td>
<td>left</td>
<td>made</td>
<td>opened</td>
</tr>
<tr>
<td>posted</td>
<td>proposed</td>
<td>puckered</td>
<td>put</td>
</tr>
<tr>
<td>removed</td>
<td>reported</td>
<td>seen</td>
<td>shown</td>
</tr>
<tr>
<td>shut</td>
<td>soiled</td>
<td>studied</td>
<td>torn</td>
</tr>
</tbody>
</table>

The number of ergativizable verbs increases and also the number of verb types which is strangely enough much higher than the one present in WSJ. We also underlined verbs that can be intransitivized, thus contributing some additional ambiguity. In some cases, the past participle is non ambiguous, though, see “frozen, seen, shown and torn”. In some other cases, the verb has different meanings with different subcategorization frames: this is case of “left”.

In any case, the parser will proceed by activating any possible disambiguation procedure, then it will consider the inherent semantic features associated to the prospective subject: in order to be consistent with a semantic classification as proto-agent, one of the following semantic classes will have to be present: “animate, human, institution, (natural) event, social_role, collective entity”.

In the affirmative case, and after having checked for the subject position/functional assignment, the analysis will proceed at NP internal adjunct modifier position. If this is
successful, the adjunct participial clause will be interpreted locally. Then the parser will continue its traversal of the grammar at i\_double\_bar position, searching for the finite verb.

In case no finite verb is available, there will be an ensuing failure which will be recovered gracefully by a recovery call for the same main constituent expected by the grammar in that position. Two actions will take place:

i. the current input word will have to be a nonfinite verb;
ii. the already parser portion of the input sentence must contain a possibly ambiguous finite verb;
iii. this token word should correspond to the predicate lemma heading the modifier adjunct clause computed inside the NP which is scanned to search for the appropriate structural portion.

The first two actions are carried out on the lookahead stack, while the third action is carried out on the NP structure already parsed and fully interpreted by the parser. The input string is then reconstructed and the ambiguous word is tagged as finite tensed verb. Parsing will then proceed smoothly.
Chapter 5

PARSING 3:
DEEP PARSER BETWEEN GRAMMAR AND STRUCTURE

1. INTRODUCTION

As discussed in the previous chapter, human language is highly ambiguous at all levels of analysis and interpretation thus entailing non determinism: this notwithstanding, we have shown how a lookahead mechanism is able to disambiguate lexical categories in order to direct the moves of a parser as much deterministically as possible. This strategy follows a topdown schema where grammar rules rather than input data guide the parsing process.

In a bottomup schema the data would be used instead: this could end up in X-bar like constituency where all possible alternative structures are explored in parallel before attachment to a higher major constituent may take place. Since phrase structure rules must be accessed in some order local failures would be the way to cope with lack of determinism; a global failure would be made to coincide only with the end of all grammar rules and input string left unconsumed. In all other cases, and supposing to have a grammar large enough to be able to encompass all possible surface structures of the language, the net result will be a certain number of output structures only one of which will be the most adequate to the context.

This is what happens in most unification-based parser which might incorporate a chart-like internal structure with an agenda(see Stock). In this formalism rewriting rules are substituted by attribute-value pairs and grammar and lexicon are viewed as data-structures (Kay, 1985). In such an approach, however, data-structures become the repositories of all grammatical knowledge and are very big and hard to maintain. Constraints would assume the same weight irrespective of their role in a given syntactic configuration. Linear order and hierarchical relations in terms of constituency would matter less to the wellformedness of the final structural configuration.

On the contrary we assume that linear order is an important parameter for human languages and as such it should be accounted for in the underlying grammar rules order. Besides, we also believe it important to fire semantic restrictions as soon as possible in order to let syntax and semantics mutually constrain each other. For semantics here we intend
properties of predicates and constituents as expressed in the underlying lexical form. As to
adjuncts, we apply cross compatibility checks to allow for a certain adjunct or modifier to be
accepted locally, explained in more detail in the previous chapter.

2. Grammar and Ambiguity

In our system, the Parser builds c-structure representations, which undergo grammatical
wellformedness tests by which lexical semantic information is appended to each constituent.
Finally constituent information is dropped and DAGs (Direct Acyclic Graphs) are built in
order to produce f-structure configuration.

Structural information is essential for the assignment of functions such as TOPic and
FOCus. Questions and relatives, (Clitic) Left Dislocation and Topicalization are computed
with the Left Extrapolation formalism presented by Pereira(1981;1983). Procedurally
speaking, the grammar is implemented using Prolog Horn Clauses. In particular,
Extrapolation Grammars allows for an adequate implementation of Long Distance
Dependencies: restrictions on which path a certain fronted element may traverse in order to
bind its empty variable are very easily described by allowing the Prolog variable associated to
the element in question - a wh-word or a relative pronoun - to be instantiated in a certain c-
structure configuration. Structural information is then translated into functional schemata
which are a mapping of annotated c-structures: syntactic constituency is now erased and only
functional attribute-value pairs appear. Also lexical terminal categories are erased in favour of
referential features for NP's determiners, as well as temporal and modal features. Some
lexical element disappears, as happens with complementizers which are done away with and
substituted by the functional attribute SCOMP or COMP i.e., complement clause - in Italian
FCOMP.

Italian is a highly structurally ambiguous or underdetermined language (see Delmonte,
1985), so that semantic or thematic checking seems necessary at this level: in particular, long
distance dependencies activate all kind of functional restrictions available, since they may be
used to prevent backtracking which is time-consuming. We use Case, Gender and Person, as
well as semantic categories of the bindee whenever available, to restrict the choice of the
binder, as will be shown in detail later on.

It is worth while reminding that f-structures coincide partly with lexical forms, i.e. a
predicate-argument structure paired with a grammatical function assignment; in other words
an fframe PRED whose f-value is a lexical form. Usually clause nuclei are the domain of
lexical subcategorization, in the sense that they make available to each lexical form the
grammatical functions that are subcategorized by that form (see Bresnan, 1982:304). In case
also nouns are subcategorized for, the same requirement of coherence and completeness may
be applied. Not all nouns however take arguments. As a consequence, "...an f-structure is
locally coherent iff all of the subcategorizable functions that it contains are subcategorized by
its PRED; an f-structure is then (globally) coherent iff it and all of its subsidiary f-structures
are locally coherent. Similarly, an f-structure is locally complete iff it contains values for all
of the functions subcategorized by its PRED; and an f-structure is then (globally) complete
iff it and all of its subsidiary f-structures are locally complete."(ibid.,305). In this sense f-
structure is a notion absolutely parallel to Chomsky's(1986) Complete Functional Complex,
with the difference that in LFG grammatical functions are all made available in the lexical
form - in particular the SUBJect -, whereas in a CFC this must be stipulated, and then recovered from s-structure.

3. **C-structure BUILDING**

In a language like Italian, at least three clause structural organizations are possible:

- a canonical organization, corresponding to the standard case in which constituents occupy their canonical positions; subjects come in preverbal position, objects and obliques in postverbal positions and adjuncts may alternate in preverbal or postverbal positions - although they may alternate freely also between verb and object NP;
- an inverted organization, corresponding to presentative constructions in which the subject occupies postverbal inverted position and an expletive may be present, "ci", or an oblique locative may be preposed in the subject place; or else nothing which relates to the arguments of the predicate be present in preverbal position. The latter case being allowed in Italian but not in other languages;
- a marked organization, corresponding to a complete reversal of constituents, allowed only in Italian, in which the object NP comes in preverbal position and the subject in postverbal position. The subject in this case, might also be an empty category.

Other structures occur with psychic verbs which subcategorize for an open proposition, an infinitival clause as open complement; copulative constructions with a closed tensed or untensed proposition as subject which might be anaphorically controlled by an adjunct PP headed by "for". Also to this lot, belong left dislocation constructions with clitics as topic variables; topicalized impersonal structures, and other constructions.

Even though LFG does not independently provide the tools to build a richer c-structure configuration, we think it highly important to organize c-structure rules for sentence level representation in line with the latest chomskyan framework: by extending the X-bar system for the syntactic representation of constituency by the introduction of functional major constituents it identifies at least the following basic levels:

\[
CP \rightarrow Spec, C' \\
C' \rightarrow C, IP \\
IP \rightarrow Spec=NP(subject), I' \\
I' \rightarrow \text{Inflected Tensed Verb Form, VP}
\]

According to this configuration, adjuncts and constituents like wh- words for questions and topicalized NPs, adjoined at sentence level, will be computed at first in a CP constituent and then passed down to the lower level of analysis. This organization of constituency allows for complementizers, i.e. the head of CP, to be kept separate in C' level so that a nice interaction may be possible, if needed.

When IP is reached, the NP subject or sentential subject should be computed: at this point there are at least two possible parsing strategies to be followed, both theoretically plausible. The former is in line with LFG traditional view that no empty category should be produced
unless it is strictly required by language typology. The latter is in line with Chomsky's assumption of the necessity to pose a basic structural or deep structure configuration which is equal for all languages. In the former case no empty subject NP should arise in case the structure to be analysed is an inverted construction: this is justified by the fact that the Subject NP is actually to be found in inverted VP internal, or VP adjoined position. Since no NP movement is postulated in LFG there would be no possibility to adequately bind the empty category previously generated in preverbal position. Thus, the sentential structure of inverted, presentational constructions corresponds directly to a VP (see Baker, 1983).

In the latter case, the subject position is filled by an empty category and it should be done away with when parsing the actual lexical subject NP in postverbal position. In case we choose the first strategy, see how the reasoning proceeds with parsing: since Italian freely allows the subject to be left lexically empty, and since we do not want to produce an empty little pro in case the lexical subject is present in postverbal position, the rule for marked presentational IP must be accessed first. In case the sentence has a canonical structure, failure would have to take place in order to start the second rule for canonical IP. The reason to let the presentational structure come first is due to the fact that in case the sentence starts with a lexical NP before the VP (computed at first as subject), a fail is performed very soon. Here we should note exceptions like bare NPs with a head noun homograph with a verb - which is a common case in English - less so in Italian. In case no lexical NP is present, there are still two possibilities: we either have a canonical structure with an empty little pro as subject, or we have a fully inverted structure.

At first we must assume that no subject is available and try to compute an inverted Subject: this might fail, in case the NP computed in the VP is not interpretable as Subject but as Object of the main predicate. However, we take the marked option to be more frequent and less extendible than the other way round: not every verb class may undergo subject inversion, which is not completely free (see Delmonte, 1991). And even if it does, there is quite a number of restrictions that may be made to apply to the inverted subject, as to its referential features (definiteness, etc.), which do not apply to the canonical object NP.

As can be easily gathered, the number of drawbacks from the point of view of parsing strategies is quite high: failure requires backtracking to be performed and this might be very heavy, depending mainly on what has been computed as inverted Subject. Not to mention the fact that VP rules should be duplicated in part.

As to the second choice, there will be only one general procedure for parsing grammatical sentence structure, which would postulate the existence of a subject position to be filled either by lexical material or by an empty constituent. In other words, in case the sentence starts with a verb we let typologically determined parameters decide whether it is possible to build an empty subject NP or not: in case we are parsing Italian texts this parameter would be active, but in case we are parsing a text belonging to Germanic languages, it would be disactivated. When we generate an empty category in subject position it remains to be decided what to do with it in case a lexical NP in postverbal position is computed, and this is interpreted as the actual Subject function of the sentence, the trace should be discarded.

In our parser C-structure building corresponds to a partial interpretation of each constituent: in fact, when a parse is completed, we assign a structurally determined grammatical function label which could match semantic checking procedures performed when annotated c-structure is built, or it might be rejected as semantically inappropriate, due
to selectional restrictions associated to that NP. Grammatical functions assignment is required in all cases in which a presentational construction has been parsed: it is just on the basis of the structural position of a given constituent, the postverbal NP, that we know what is the pragmatic import of the entire utterance. And this will be registered only in the grammatical function assigned to one of the arguments of the predicate, which is computed either as Subj_Foc, or Subj_Top according to whether it is an indefinite or definite NP respectively. The empty NP subject is not bound to the actual lexical NP found in inverted position, and it is simply discarded from the final representation. In this way, the annotated c-structure outputted by the parser is cp rewritten as vp, but the postverbal subject is computed with an adequate grammatical function. Backtracking is thus totally eliminated, and there is only one single procedure which applies to all sentential structures.

Here below, the main calls to utterance level are listed: rules will all be presented as formally represented in Definite Clause Grammars to simplify the description.

1. utterance(X) --> assert_dir(X, nil).
   utterance(X) --> standard_utterance(X, nil).
   standard_utterance(X, nil) --> wh_question(X, nil), [?].
   standard_utterance(X, nil) --> yes_no_question(X, nil), [?].
   standard_utterance(X, nil) --> assert_cp(X, nil), [?].

At the highest level we want to differentiate between direct speech and other utterances, which are all called by standard_utterance. Only simplex utterances are described here and not complex utterances. A simple utterance can either be started by the SPEC of CP containing a ±wh element, i.e. it can be a question, a topicalization or a left dislocation, or a yes-no question. These are fairly general rules applying to all languages: there is a call to adjuncts at cp level, and a call to aux-to-comp elements which however is typologically restricted. It applies to Germanic languages in particular, where auxiliaries may be computed in comp position, as will be discussed below in more detail. Rules for questions are listed below in the section on long distance dependencies. In case the call to canonical structures fails, we try topicalized and dislocated constructions, as shown by 3 below.

2a.assert_cp(ass(ind(Simb),v(Verb, Voice, Lex_form,[Funct_Feats]),Support, CatV, Args, Adjs), H0) --> aux_to_comp(Aux, Verbo2, Support, CatV, CatG, SubClIn),
   adjunct_cp([Sent_Adjs]),
   i_double_bar([Sent_Adjs], Verbo2, Aux, CatG, SubClIn, L, Supp, CatV, Args, Adjs, H0).

3a.assert_cp(ass(ind(Simb),v(Verb, pass, Lex_for,[Funct_Feats]),Supp,CatV,Args,Adjs),H0) --> object(NP1,np,_,_,nil,nil),
   adjunct_cp(Sent_Adjs),
   pro(cl(TypePron,Part,Pro,[Funct_FeatsC]),
   {Pro=si, NP=np(ind(Ind),cat([hum]),cl(pro_imp_,si,[Funct_Feats],[nom]),
   spec(def("-"),part("+")), _ , _ , _ ),
   verb_phrase_imp([np(NP1,[Funct_Feats1])],np(NP,[Funct_Feats]),Sent_Adjs],
   v(Verb,[Funct_Feats]),Lex_form,Supp,CatV,Args, Adjs,H0).
The first of these calls, is a call to impersonal SI reverse constructions which are usually associated to passive voice. Then we have reverse constructions with transitive verbs which may have the object in sentence initial position: this NP cannot be used to trigger Agreement with the Verb, and must be taken at Top level. Two possibilities exist now: in the first case, we have a typical left dislocation construction, which has the following essential structure: NP Object, NP Subject, resumptive clitic, VP structure, and may be exemplified by the sentence, "Il libro Gino lo ha comprato"/The book John it has bought. In the second case, left dislocation is accompanied by subject inversion, i.e. the essential structure, NP Object, resumptive clitic, tensed verb, NP subject, as in the following example, "Il libro lo ha comprato Gino"/The book it has bought John. Thus, when a clitic is present and the Subject is in inverted postverbal position, this is captured by the rule where the topicalized Object NP is linearly followed by a clitic which has accusative case, and no intervening lexical NP can be computed.

From this structural level, either a VP could be straightforwardly computed, or else, an empty NP Subject be postulated and then discarded. We prefer the first option. From structural representation we can already tell that the subject must be empty, owing to the presence of an object clitic. In the former case, the clitic is present but the Subject is in preverbal position. Or else, which is the option available in all languages, as in "Ski John loves", we have a Topicalization or focalization, i.e. the Object is in Top CP, and the Subject in preverbal position. No clitic appears. This is achieved partly by constituent check when building annotated c-structure, and partly by Interpretation at sentence level, when all constituents have been recovered and constructed. The presence of a bound clitic for clitic left dislocation, or else the absence of a clitic and the type of constituent can now be adequately dealt with respectively, as a case of left clitic dislocation with subject focalization in the first case, left clitic dislocation in the second and topicalization in the third case. In the former case, the inverted subject will be interpreted as Foc; in the latter case the preposed object will be interpreted as Top; and in the third case the preposed object as Foc. Notice also that no lexical subject might be present, thus resulting in a simple clitic left dislocated structure with an empty NP subject.

It is interesting to note that all this will follow independently by letting the adequate structure building and constituent check at VP level. After cp has been correctly built, we
activate the call to ip where subject NP and negation may be parsed; then a call to i_one_bar, will activate calls to Clitics and Infl, for all inflected verbal forms. The call to Clitics, is allowed both for German and Italian; it also applies exceptionally to English "there", provided no NP subject has been analyzed. Infl is a call which is specialized for different languages and the subsequent typologically marked constructions of Italian.

4a.i_double_bar([Sent_Adjs],Verb,Aux,CatG,SubCln, SubCOut,Supp,CatV, Args, Adjs, H0) 
---> subject(SN,[Funct_Feats], H0, H1),
        negat(Ne),
        adjs_preverb(Avv2,_,)
        parent(Par),
        i_one_bar([np(SN, [Funct_Feats]), [Sent_Adjs], Neg, Verb, Aux, CatG, SubCln, SubCOut, Supp, CatV, Args, Adjs, H1]).

Parsing the appropriate VP structure requires the instantiation of the appropriate syntactic verb class of the main predicate: in this case, it may either belong to the class of psychic or copulative verbs. Theoretically speaking, c-structure is now represented with a verbal phrase which contains no verb, which has been raised to infl, in case it is a tensed finite verb. Notice that in order to enter the call for inchoativized verb_phrase, aspectual class is needed; in addition, Subject NP should be an empty pro, in Italian.

All subject inverted constructions in the table are constrained by a check on the subject NP: it must be an empty category. This check also applies to impersonal-si constructions and to dislocated constructions. In this way, no backtracking will be allowed. In addition, syntactic category of the main verb should always be checked accordingly. In addition, inchoative constructions and impersonal-si constructions are also typologically marked, since they are only allowed in Romance languages; also fully inverted transitive constructions and intransitive reflexive structures are only present in Romance languages. The call to intransitive verbal phrases is subsequently further split into the four syntactic classes listed in brackets. Transitive structures are differentiated according to the complement type: i.e. adverbal objects require a separate treatment owing to differences in the interpretation of its NP, see "John spent three days in Venice"; "Mary weighs 45 kilos" and so on. Transitive verbs with open complements are also special in that their object is nonthematic and is interpreted in the open complement, see verbs like "believe John stupid", "see Mary in the shower",("consider" and so on. The presence of syntactic classes in verbal entries listed in the lexicon, used as a filter in the construction of VP, might be regarded as redundant information; but from a computational point of view it turns out to be a very powerful tool. Also remember that Italian verbs select auxiliaries according to syntactic class! In particular, unaccusatives require "essere/be" and unergatives "avere/have".

Table 25. VP Types Realization Rules

<table>
<thead>
<tr>
<th>Syntactic Category</th>
<th>Subcategorization</th>
<th>Aspectual Category</th>
<th>Auxiliary Mood</th>
<th>Clitics</th>
<th>Subj.NP</th>
<th>VP Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>transitive</td>
<td>passive</td>
<td>auxiliary</td>
<td>passive</td>
<td></td>
<td>empty</td>
<td>inchoative</td>
</tr>
</tbody>
</table>


Rather than looking into all the phrase structure rules implemented in the parser we shall consider the verb_phrase for copulative constructions and look in more details into it. This is the main rule:

\[
5.\text{verb_phrase_cop}(\text{VarSN}, P, \nu(\text{Mood},\text{Tense},[\text{Funct_Feats}]), \text{Lex_form}, \text{Supp}, \text{CatV}, \text{Args}, \text{Adjs}, \text{H0})
\]

\[
\rightarrow \text{adv_phr(Adv3)},
\text{check_clitic_object}(P, \text{Xcomp}, \text{Subj}),
\text{xcomp}(\text{Xcomp}, \text{Gen1}, \text{Num1}, \text{TipoXcomp}),
\text{prep_phrases}(\text{LSP}, \text{H0}, \text{nil}),
\text{interpret_cop}(\text{Verb}, \text{Lex_form}, \text{NP}/\text{Subj}, \text{Xcom}/\text{TypeXcomp}, \text{LSP}, \text{CatV}, \text{Args}, \text{Adjs}, \_x), \text{Adjs}).
\]

The rule starts by checking whether a "lo" clitic has been found, in that case this will constitute the open complement, as in sentences like "Gino lo è" = John it is (happy), where "lo" is the resumptive invariable clitic for open complements in Italian. In case another clitic has been computed, this can only be treated as a complement or adjunct of the open complement, and is consequently included as first element in the list of constituents passed
onto the open complement call. The XCOMP call can be instantiated with any of the allowable lexical heads X=P,A,N,V,Adv, and its associated main constituents. Finally, there is a check on the specifier and referentiality of the preverbal NP computed: in case it is a deictic pronoun, or the Xcomp is a proper noun, this structure will be locally computed as inverted structure as appears in sentences like: The murdered is John, and This is a spy story.

4. FROM ANNOTATED C-STRUCTURE TO F-STRUCTURE

4.1. Lexical Control

Lexical control is activated by the lexical form and integrated by the redundancy rule for functional control into the annotated c-structure level. It requires the creation of a SUBJect variable bound by the lexically assigned controller. The result is a lexical chain.

Here below, we give the rules for the interpretation of structures associated with copulative constructions where the rule for function control is called.

6. `interp_cop(Verb, Lex_form, NP1, Xcom/TyXcom, ListSP, CatV, Args, NewAdjs_vp)`
   ---> `pred_v(verb,Verb,CatV,[FirstArg|ListArgs])`,
   `Lex_form=[FirstArg|ListArgs]`,
   `interpret_subj(NP1,FirstArg,Subj)`,
   `find(cat(ListCat),NP1)`,
   `interp_xcomp(Xcom/TyXcom,[FirstArg|ListArgs],NewArgs,X,Contr,Arg_contr,ListCat)`,
   `(X=form(ci))`,
   `assign_control_xcomp([Subj,X|Arguments_vp],Contr,Arg_contr,ListCat);`
   `X=form(ci))`,
   `Args = [Subj,X|Arguments_vp]`,
   `interp_adjuncts(Args,Adjuncts_vp,NewAdjs_vp),` !.

4.2. Open complements

This is the redundancy rule to assign lexical functional control to locative pcomps:

7. `interpret_xcomp(SP1/pcomp,[FirstArg|ListArgs],NewArgs,X,Contr,Arg_contr,CatSubj)`
   ---> `nonvar(SP1)`,
   `find_controller_xcomp(pcomp,Role,ListCat,[subj=Arg_contr],ListArgs,NewArgs)`,
   `SP1=pp(prep(P,cat(Cat),Mod),NP)`,
   `member(CC,Cat)`,
   `Role=locative`,
   `adjunct_type(Role,P,ListCat1)`,
   `member(CC,ListCat1)`,
   `Subj =arg(function(subj),role(nil),np(ind(Simb), pr(vbl,vbl,_,_,_),controller(Contr)))`,
   `gen_sym(np,Simb)`,
   `Obj = arg(function(obj),role(Role), NP)`,
The preposition is checked by means of the same call "adjunct_type" which is used to check for semantic compatibility in adjuncts. In case the preposition is consistent with the locative role interpretation assigned to the Pcomp open function, a Subject function is created which contains a lexical variable controlled by the lexically assigned functional controller. In order to find the controller index, we access the lexical form and the associated functional control equation.

The functional control equation is accessed once functional completeness and uniqueness is verified. As appears, in case a predicate contains "xcomp" as functional attribute for the open complement this is verified as default specification.

When these procedures have been completed, control is passed onto the open complement and features are passed from the controller to the controlled argument. This is the rule for control assignment to xcomps (predicative complements acomp, ncomp, vcomp, pcomp):

- assign_control_xcomp(+Args,-Control,+Arg_control)

where: Args, are the arguments of the matrix sentence, Control is a variable created within the xcomp. Arg_control indicates which is the controller (default, obj2, obj, subj) and is returned by control_xcomp. Control can take place

a. according to a default rule (the first argument present is selected between obj2, obj, subj).

b. the controller can be already present in the lexical form as [arg_controlled = Arg_control], ex: [subj = obj]

c. the control argument is not present, it is = indefinite

There are two possibilities: the controller is a theme_bound, corresponding to a nonargument function, and it is either an OBJect or a SUBJect. In this case features and semantic categories must be passed up from the open complement to the controller to interpret it. This rule is prevented from applying whenever the open complement is a Pcomp, because in that case the copulative construction must assign a location to the controller, and not a property as is usually the case with the remaining complements.

4.3. Syntactic Control

Structural control is activated by Extraposition whenever a nonargument function like TOPic or FOCus is met by the parser. The extraposed constituent looks for variables in a given structural domain and if it finds one it extends its path into it. In this way, extensions to the path are explicitly marked in the grammar and ungrammatical sentences may be easily filtered out. The extraposition device checks for functional as well as semantic information, as made available by the lexical form associated to a given predicate. To this aim, SCOMPs
or non predicative sentential complements are semantically separated into five different classes: PROP - or simple assertions, PROPF - or factive propositions, PROPINT - intensional propositions, PROPQ - questions, PROPE - exclamations. Only PROP constitutes an adequate domain for the extension of syntactic control in Italian. As the extraposition operator looks for extension variable at CP level, semantic constraints require the argument role to belong to a certain type, thus automatically excluding ungrammatical sentences. A language like English would require also Questions and Exclamations equipped with the extension extraposition variables. Notice that at lexical level, constituent information attached to propositional arguments might be differentiated into C', for sentential complements which are simple propositions, and CP for the remaining cases, in which SpecCP should be filled with some lexical material. However, no such possibility exist in our theory, since only major constituents may be associated to grammatical functions, C' being barred as an intermediate projection.

We take syntactic control to represent the best example of how a theory can be put into a psychologically realistic application. Differently from what happens in principle-based parsers, where syntactic configurations are at first built according to X-bar theory and then checked for consistency with principles and parameters, in our framework, syntactic constituency is built only in case structural configurations are consistent: principles and parameters are contained within the rules, rather than without, being context-sensitive!

Coming now to syntactic control which applies both to relative and interrogative constructions, Italian requires functional and semantic information to be made available as soon as possible in order to prevent undue backtracking to take place. In particular, as to functional features, Person and Gender are used but not Number. The reason for not using Number is simply due to the fact that in case the sentence in which the topic relative pronoun has to find its controllee is a construction with an inverted subject, we simply cannot possibly allow the parsing to fail, because failure might be too time-consuming. This might happen in sentences like "Che cosa stanno facendo (i bambini)/What are (the children) doing?" where the topic has Number set to singular, and the Auxiliary's Number is plural. Early failure would simply be caused by a mismatch in functional features. Since Agreement precedes all other syntactic and semantic controls, it should be more flexible and be independently activated rather than automatically instantiated.

The case of reversed constructions is only possible with transitive verbs and can be easily dealt with at the level of interpretation: selectional restrictions attached to "what" include [+object], whereas those attached to "who" include [+human]. In this way, the topic controller is bound to the first available controllee: thus, in case the subject NP is missing, the extraposition variable will be bound locally. Then, if the NP found in object or VP internal position is the actual object it will be assigned the object semantic role; on the contrary, if it is an inverted subject, it will be interpreted accordingly and the grammatical function associated will be FOC. Notice that in case it is empty, no FOC may be generated, since a little pro cannot constitute an adequate focalizer!

In English, to care for complementizerless relative clauses, we activate the same call to ecp_check which we assume is required for complement clauses, where case is set to be accusative. As a side effect, given that the subject should be lexically expressed (the empty subject rule being barred in this language), this call will force a grammatical structure to be generated.
8. \texttt{check\_that\_trace\_eff(Co, [Funct\_FeatsC]) \rightarrow}
\begin{align*}
&\{\text{gr(english), }\{\text{Co=[]}\}, \text{on(Case, [Funct\_FeatsC]), on(acc, Case) ; }\{\text{Co=}[]\}\}.
\end{align*}

4.4. Extending the Binding Domain

Extension of the binding domain requires the presence of extraposition variables, i.e. traces of wh- movement or metavariables in LFG. In addition, the governing predicate must select for its complement in terms of its semantic role: a declarative proposition is a valid extension, but a question, an exclamation or a factive proposition is computed as an island. Notice that the rule for sentential object is just a special case of the general rule for the OBJect construction. Also, in our parser we allow for an object to be left unexpressed and be interpreted as an implicit argument, i.e. as a dummy quantifier "exist" which is then dealt adequately by the following modules: in particular it will be assigned lower scope than any other quantifier in Logical Form and will become a generic class identifier in the semantics with semantic features restrictions. The theoretical and practical implications of this choice have been discussed in the specific chapters in Book 2.

In that case restrictions apply at the level of semantic role, in the sense that the complement type is selected by means of semantic rather than syntactic category, cp, or functional category, scomp. At this level selection applies across language typologies; however, at a lower level both language type and syntactic information should be checked. In particular, German is a language that does not allow complementizerless complement and requires a comma to precede it. As to the treatment of that-trace effect in English, we check whether Case transmitted by the CP specifier level is accusative whenever the Complementizer is empty. Notice that also in Italian the complementizer can be omitted, with a special class of predicates and with certain moods.

\begin{align*}
9b. &\text{scmp}(Fcomp,H0,nil) \\
\rightarrow &\{\text{gr(deutsch)}, [','], \text{comp(Co), }\{\text{Co=}[]\}, \text{assert cp}(Fcomp,H0).}
\end{align*}

\begin{align*}
c. &\text{scmp}(Fcomp,H0,nil) \\
\rightarrow &\text{comp(Co),} \\
&\{\text{gr(romance)}, \text{assert cp}(Fcomp,H0) ; \text{on(Case, H0), check that trace eff(Co, Case), assert cp}(Fcomp,H0) ; \{\text{Co=}[]\}, \text{assert cp}(Fcomp,H0))}.
\end{align*}

5. Building Grammars for Other Languages

If we consider other languages, we see that the grammar written in DCG format for Italian may be easily extended to languages like French, English or German, which are usually more restricted as to variations in structural configuration.

Consider English, which like French and other European languages does not allow the SUBJect to be left morphologically unexpressed - except for sentential coordination. This can be easily taken care for in that the call to NP subject is performed by a call to SUBJect which
can be instantiated either as NP, or as a variable, or as an empty little pro in Italian. Other language simply drop the last option.

The rule tries to build a lexical NP first, then it tries to bind a syntactic controller variable built by interrogative and relative clauses, which rewrite the empty string, and generates a bound empty category. Else it will build a little pro and rewrite the empty string. However, the rule for the empty little pro is typologically marked: it can only be accessed by the Italian grammar!

5.1. A Grammar for English

In English, wh- words may be computed in situ, i.e. rather than being a case of long distance binding, they are simply computed as if they were a case of pronominal NP. This can in turn be subject to a number of restrictions in case another +wh element has been already processed at CP level. Generally speaking, constituents which can be omitted without causing agrammaticality are more easily extractable than the ones which must be obligatorily present in the final structure. From this assumption, we see that adjunct wh- words are more easily extractable than complements; indirect object are more easily extractable than direct object and subject wh- elements. All these facts can be easily implemented in our system, given that both subject and object NPs are accessed directly from functional labels.

As to wh- adjuncts which can cross other wh- islands this will cause the Extraposition system of variable storage collapse if it is treated as a single element storage. We implemented a stack where the last element inputted is extracted first, i.e. a LIFO policy. Since the arguments will be met before the rules for PP adjuncts at VP level in English, the requirement that grammatical combination for double wh- words will be of the form ADJunct + Argument and not the reverse is automatically taken care of. Here are some relevant examples taken from Lasnik, Saito(1984):

10a. I wonder who will bring what
   b. *I wonder who what will bring
11a. Why did you buy what
   b. *Why do you wonder who left
   c. *How were you wondering what to say
12a. Why did you see who
   b. *Who did you see why

In all these cases there is a restriction that works in such a way as to prevent a Spec CP from being stored onto the stack in case it is an NP and the stack is non empty. If it were a PP, no violation would ensue, as the following examples from (Frampton,1992) show,

13a. What do you know why he wanted to see
   b. Who do you know why I think you like
   c. Who do you wonder why we fired
   d. Who do you know why I need to see
   e. Who do you wonder why I think left
   f. *Who do you wonder whether came
   g. *Who do you wonder whether John said came
   h. *Who do you wonder whether John said that came
As can be seen from these examples, a wh-word which is a PP adjunct does not cause agrammaticality; on the contrary, the complementizer "whether" does cause it. To handle this, we simply check the nature of the semantic role associated to the SCOMP of the main predicate WONDER and then we see that CP has its SPEC empty, by calling directly Cbar. No extraposition variable, or controller will be passed down in the following structure, which is an island. There is another set of constructions where the discriminating factor is the presence of a complementizer, the well-known that-trace effect,

14a. Who do you think left early
b. *Who do you think that left early
c. Why do you think (that) he left early
d. Who do you believe that Mary said left early
e. When does he think that John left

As can be easily gathered, the restriction applies only when a SUBJect NP has been questioned. This can be captured nicely in our system given the fact that when we call Cbar, where the complementizer is taken, we simply must check whether it is lexical or not in the subsequent structure of IP. In case it is lexical, i.e. Comp is not an empty list, we prevent the SUBJect NP from hosting Extraposition variables which have Case set to [nom]: in case it is accusative we still want to allow for the possibility to bind in the extended domain. Principles are thus distributed throughout the grammar by means of a set of rules: assign_empty_subject, and check_that_trace_eff which are both language sensitive.

Now consider the well-known CNPC (Complex NP Constraint), i.e. the impossibility to host a controllee of a wh-word inside a relative clause, as shown by,

15. *Who do you believe the claim that John said that came

In this case, we simply prevent extraposition variables from being made visible to the relative clause which is a subconstituent of the NP. At the NP level, however, wh-controller must appear since English allows for PPs to be extracted, from a certain nominal head though, "picture NPs"

16a. Who would John buy a/any picture of
b. *Who would John buy some/the picture of

The examples show that the NP must be restricted also according to its Specifier. English does not allow the construction Relative Clause+Interrogative Clause which is allowed in a language like Italian (see Rizzi, 1983; Delmonte, 1991) also known as Double wh-island constraint, exemplified in,

17a.*the book which I wonder who bought...
b. il libro che mi chiedevo chi avesse comprato...

5.2. A Grammar for German

Now we will look at German. It is a matter of debate whether German should be typologically treated as an SOV or verb-final language, rather than as an SVO language. In case the former is chosen, it follows that daß sentences should be treated as the clauses which have their constituents in canonical order. However, since the SOV option is based on Verb movement, and since no functional category can be "moved" in our framework, it would seem that a grammar for German is difficult to implement within the SVO perspective.
The main problem in building CP level rules for German lies in the fact that the verb must be placed in a position which sometimes precedes the SUBJ and the OBJ NPs. Thus, while keeping the general framework as for other languages, we should allow for the tensed verb to appear under a different constituent in case the sentence to be processed requires it. The first example is then a case of canonical order,

18. ... daß ich das Buch gelesen habe

The following cases require the tensed verb to appear under CP,

19a. Wer will das Buch lesen
b. Wen bist du begegnet
c. Wem hast du das Buch gegeben
d. Wer fand das Buch
e. Das Buch, habe ich nicht gelesen
f. Heute habe ich schon gegessen

In all of these examples, a wh-word, a topicalized NP, or an adjunct has already been computed in Spec CP and consequently the C-bar level has the head C empty. The tensed verb will then be taken in C, and simply passed down to IP in order to fill the variables associated to verbal functional features like Tense, Mood, Person, Gender, Number. The interpretation will be vacuously satisfied. Notice that separable particles will be taken by the same rule that in canonical sentences will accept a tensed verb, and this is done under V-bar.

20. C" --> Spec {±wh-}; ADJuncts} , C'
C' --> C {daß; (Aux, Verb [+tense])} , IP
IP --> Spec {SUBject NP} , I'
I' --> Verb {empty if copied from CP} , VP
V" --> OBJect NP , V'
V' --> (Neg) , V''
V" --> {Trennbar Particle; V[-tense]; zu + V[-tense]; (Aux, Verb [+tense])}

As was the case with English and Italian Aux-to-Comp (which is limited to gerundive adjuncts), we are presented with a situation in which there are different slots that can host an inflected Verb.

Provision is made in the rules for inverted structures due to the presence of a preposed ADJunct. Also notice that the Verb is copied from CP in case nonvar(Verb) applies, otherwise it is taken in sentence final position. The remaining rules care for usual SVO sentences, in case CP is found to be empty.

### 6. Computing Adjuncts

The parser may compute PPs either as Obliques or as Adjuncts. Adjuncts may be Open or Closed: Closed Adjuncts are circumstancials like Locatives or Temporal modifiers of the event or state. On the contrary, Open Adjuncts may see one of the participants in the action specified by the matrix predicate, and modify it consequently. Obliques are closed complements and as such are listed in the lexical form of their predicate governor. However, not all Obliques behave in the same way: some may be omitted without causing
ungrammaticality due to incompleteness, while others must be lexically expressed. See for instance the difference existing between TALK and PUT, which is based on that fact, as shown by the following examples.

21a. John talked to Mary (about her father)
   b. John talked about her father (to Mary)
   c. John is talking
   d. John put the book *(on the shelf)

TALK is a verb which does not need its oblique arguments to be lexically specified; on the contrary, PUT requires its oblique to be lexically present. To cope with this problem, two possible strategies may be envisioned: either we mark in the lexicon all predicates allowing their oblique arguments to become implicit arguments and as a consequence we get the requirement that the remaining predicates do not allow their oblique arguments to be left implicit; or we compute it as a default. Since we are analyzing sentences, we may simply assume that every time a predicate has an oblique in its lexical form this may be left implicit, unless it is required to complete the meaning of the predicate itself. Motion verbs require the oblique to be lexically expressed for semantic reasons, i.e. the oblique specifies the location to which the subject NP and motion are directed, or come from. Another interesting case of omissible oblique is constituted by "by/da" PPs in passive constructions. This is how we interpret obliques in passive sentences:

The first rule tries to compute a lexical PP which has a preposition "da/by" and the semantic features required from the corresponding SUBJect inherent features. In case the rule has success, the lexical form is changed from the active transitive form into the passive corresponding transitive form due to the functioning of the lexical redundancy rule which turns the Subject argument into an Oblique and the Object argument into a Subject while semantic roles are kept in the same order. If the rule fails to apply, it means that no PP "by" has been found in the input c-structure level and thus an implicit argument must be generated by the interpretation level.

The implicit argument is generated as an oblique with a given semantic role passed by the lexical form and associated to the Subject argument of the predicate, and is a generic existential quantifier which will be assigned lower scope than any other quantifier in the logical form.

More complex cases are constituted by PPs which behave like obliques semantically but can be freely omitted: in case they are present, they add pragmatic information which is relevant to the interpretation of one of the arguments of the predicate. Perception verbs like SEE, are treated as transitive verb which may have a locative PP or a proposition: the PP is computed as an open complement, a PCOMP. This PP is then not computed as an adjunct, and the relevant tests are as follows,

22a. John has seen Mary in the kitchen
   b. John has seen a beetle in the soup
   c. In the kitchen John has seen Mary
   d. In the soup John has seen a beetle

where the location is always predicated of Mary/beetle and not of John. However if we consider verbs such as HIT, BREAK, they may be constructed with a locative PP, but have no oblique in their lexical form; THRUST, INSERT have an oblique in their lexical form; OPEN, FIGHT may be construed with a PP instrumental, but have no oblique in their lexical form.
In our model, locative PPs headed by "against" are computed as open adjuncts with predicates like HIT, THROW, BREAK. Differently from other preposition, "against" indicates the impingement - the forceful contact of an object/instrument against a place, the goal or the location where the result PP describes the ultimate destination or trajectory of the Obj/Theme_Affect after being struck: this is detected by the Control Rule for Open Adjunct interpretation (differently thus from Copredication in Gawron, 1987). The same would apply in case the verbs were transitives like BANG, KNOCK (...the hand against the table), or CUT in "John cut his foot against the rock", where the rock is at the same time both the CAUSER and the LOCATION. This is why we might as well say: "This rock cut John's foot - it happened by accident." "Against" always carries the meaning of forceful contact whenever an Object/Theme_Affected is the controller, and the Subject in that case will certainly be an Agent; but in case the controller is not an Affected Theme, only "contact" will be left as residual meaning, as in,

23a. The trees were black against the morning star (PCOMP)  
    b. He placed the ladder against the tree (PCOMP)  
    c. They put the piano against the wall (PCOMP)  
    d. Jack's head was leaning against the window pane (PCOMP)

In sentence a. and d. the main predicates may be dubbed as "states" describing the position or the colour of the subject from a given perspective; however sentences b. and c. may be dubbed as punctual processes, achievements involving motion of the Object by the subject Agent, who CAUSES the object NP to be in a given location. The Pcomp structure introduces an open complement or proposition whose subject is the object NP itself.

On the contrary, Instrumental PPs will always be computed as closed Adjuncts, thus modifying the event described by the main predicate, as in,

24a. John hammered against the door with a stick  
    b. Mary broke the vase with a hammer  
    c. George hit the fence with the car's bumper  
    d. Jack hit/throw the ball against the fence with the wooden bat  
    or the debated ambiguous example,  
    e. John saw the man in the park with the telescope

where we want to say that the location of the man is in the park, but the location of the telescope is with the subject. In other words, the spatio-temporal location of the main predicate is inherited by the subject and the event modifiers; the spatio-temporal location of the object may be the same or be different. In the following text,

A. "John took Mary to the cinema with his old Mustang. But at a crossroad they met Frank who made them change their mind. So they went to a pijama party instead."

both John, Mary and the Mustang share the same location. However, since TAKE can be regarded as an accomplishment, and not an achievement, the action might not be completed: so along the way to the cinema, there is a provisional location and, when they stop at a crossroad, the location changes again. TAKE is like SEND, MAIL and other processes which require the completion of the action to take place: this is expressed by achievements like ARRIVE, REACH, GET, RECEIVE and so on.

The same things might be said about verbs like FILL, BRUSH, REMOVE, where we have a change of state in the object NP brought about by a change of location: these verbs have alternates in English but not in Italian. You can "brush the coat from stains" in English,
and "brush stains from the coat", but in Italian only the second action is acceptable. You fill the glass with water, but in Italian you fill the glass "di/of" water (See Cognition, 1988).

6.1. Locatives Adjuncts and Adverbials

The same intuitions can be applied to adverbials and locative adjuncts. Consider adverbials discussed by Jackendoff (1972), as belonging to five distributional classes, according to whether they can appear initially in a clause, in auxiliary position, or final. Additionally, in Italian, they may be inserted between main verb and NP object, and between NP subject and main verb:

Class I: can appear in all five positions but with varying meaning (cleverly, carefully, deliberately, voluntarily, intentionally, clumsily...)

Class II: appears in all five positions with no meaning change (quickly, slowly, quietly, frequently...)

Class III: does not appear in final position (evidently, probably, unbelievably...)

Class IV: does not appear initially (completely, easily, totally...)

Class V: final position only (hard, well, more... plus the two Italian variants);

Class VI: auxiliary position only (merely, virtually... plus the two Italian variants).

We assume that adverbs are predicates which do not have arguments nor select theta-roles, but are predicat ed of a certain grammatical function: class I adverbs are functionally controlled by the SUBJect; class II and III are bound by tense and aspect at clause level; class IV adverbs are temporally dependent and predicate only within the VP, thus they predicate of the result of the action or event expressed by the verb. As Roberts (p.81) proposes, thematic properties are relevant for the selection of some class: in particular, class I adverbs require an Agent or an Eventive predicate, while class II adverbs only require that the predicate be eventive.

6.2. Functional Control

The grammar provides rules for open Adjuncts which are subject to Structurally determined Functional Control. Adjuncts are checked for semantic compatibility with the Verb, the SUBJect or the OBJect in order to establish the more likely or preferred level of attachment. When the Fusion mechanism is activated and the controller is assigned, further semantic checking procedures are used to reject inappropriate or inadequate binders.

All adjuncts must be semantically selected: this amounts to saying that the semantic category of the head NP, as well as the prepositional head must be checked and matched against the semantic category of the modified head, be it a verb or a noun. This applies to adjectives, either attributive or predicative, to adverbs and to PP. Sentential adjuncts are simply attached at a VP node in case they are taken after the main clause has been parsed, or at a higher node in case they precede the main clause. Adverbs are also computed as PP in case they belong to the class of closed adjuncts: i.e. temporal and locative adjuncts. The PP structure allows the parser to spell out properties of the SPECifier which may be important for interpretation and logical form: this applies to such adverbs as "whenever, everywhere, etc." which contain a universal quantifier.
6.3. Closed Adjuncts

Temporal and locative adjuncts are restricted by a predicate that matches aspectual features of the predicate and type of adjuncts: these are computed as closed adjuncts which modify the main predicate. However, locatives individuating origin by the preposition "from", are computed as open adjuncts, modifying one of the arguments of the main predicate.

Open adjuncts like matter, and impingement are treated as predicative functions which are only visible at f-structure and are assigned a controller according to a rule which resembles the default rule of lexical control, except that it is made sensitive to semantic roles, as shown here below.

25a. assign_control_xadj(Args,from,Contr,default) :-
    member(arg(function(subj),_role,Struct),Args), !,
    find(ind(Contr),Struct), !.

b. assign_control_xadj(Args,Prep,Contr,default) :-
   (member(arg(function(obj2),role(patient),Struct),Args); 
    member(arg(function(obj2),role(addressee),Struct),Args); 
    member(arg(function(obj2),role(goal),Struct),Args); 
    member(arg(function(obj),role(theme_aff),Struct),Args); 
    member(arg(function(obj),role(theme_eff),Struct),Args); 
    member(arg(function(subj),_Role,Struct),Args); 
    member(arg(function(subj_top),_Role,Struct),Args)), !,
    find(ind(Contr),Struct), !.

c. assign_control_xadj(Args,Prep,indefinite,default).

26a.assign_contr_xadj_adjectv(Args, cat([CatAdj]), Gen, Num,Contr,default) :-
    stage_level([CatAdj]),
    legal_fun_role(Fun, Role, ConstraintOnRole),
    member(arg(function(Fun), role(Role), ArgInfo), Args),
    call(ConstraintOnRole),
    match_gen_num(GenAdj, NumAdj, ArgInfo),
    find(cat(CatNoun), ArgInfo),
    control_modifier_saa1(CatNoun, cat([[CatAdj]])), !,
    find(ind(Contr),ArgInfo).

b.legal_fun_role(-LegalFunction, -Role, -ConstraintOnRole).
    legal_fun_role(obl, Role, Role=agent).
    legal_fun_role(obj2, Role, Role=benef).
    legal_fun_role(obj, Role, member(Role, [theme_aff, theme_eff])).
    legal_fun_role(subj, _, true).
    legal_fun_role(subj_top, Role, true).
\textit{legal\_fun\_role(subj\_foc, Role, true)}.

\texttt{c.match\_gen\_num(+Gen, +Num, +ArgInfo).}
\texttt{match\_gen\_num(Gen, Num, ArgInfo) :- find(n(_,Gen,Num), ArgInfo).}
\texttt{match\_gen\_num(Gen, Num, ArgInfo) :- find(npro(_,Gen,Num), ArgInfo).}
\texttt{match\_gen\_num(Gen, Num, ArgInfo) :- find(pr(_,_,_,_,Gen, Num,_), ArgInfo).}
\texttt{match\_gen\_num(_, _, ArgInfo) :- find(cl(_,_,_,_,_,_), ArgInfo).}

d. stage\_level(emotive).
\texttt{stage\_level(evaluative).}
\texttt{stage\_level(state\_temp).}

As appears from the control rule, adjectives require their controller to be agreed in Gender and Number, agreement performed by "find"; also semantic features associated to the adjective must be compatible with those of the controller, this control being performed by "control\_modifier". Semantic roles are then accessed in line with what happens with PP modifiers. Finally, there is a general restriction which only applies to adjectives and is computed from the semantic selectional restriction associated with the lexical form: the semantic class is reinterpreted in terms of the subdivision introduced by Carlson(1977) of stage level vs individual level predicates. Stage level predicates can be used for secondary predication, whereas individual level ones can only be used for primary predications. As can be seen from the lines of code listed above, stage-level predicates include adjectives which require some subjective relation or some feeling or emotion. In addition, also temporary state predicates are included. Real stative predicates are excluded, together with the majority of semantic classes like for instance, temporal, provenance, both geographical, theoretical, religious, racial, scientific etc.

We take open adjuncts to adequately represent all instances of secondary predication; resultatives, on the contrary, as well as causatives and "small clauses" constructions as the ones occurring with perception verbs or ECM verbs like "consider" are predicted in the lexicon, where the lexical form requires the (non)thematic or nonargument OBJect to be interpreted in the open function XCOMP. Differences in the interpretation of small clauses and resultatives are encoded in the semantic role associated to the OBJect NP, which in the case of resultatives has an independent semantic role, as shown by the example included in the following section.

As to adjuncts, they are all computed in the semantics according to both their functional status, either open or closed, and their role. In particular, open adjuncts can be interpreted as location of their controller which is a participant in the event/state denoted by the main predicate; or else they can be computed as property predicators, in case they specify matter or some other intrinsic quality. Closed adjuncts, on the contrary, can either modify the event/state or the situation spatiotemporal location; or they can modify the verbal predicate in case they indicate modality.
7. **F-structure Representations**

It is worth while reminding that f-structures representations are the input to the rest of the system: in other words they serve as the structural input both for the binding module, for the logical form, for the semantic interpreter. Even though we may naturally suppose that f-structures may be pragmatically inconsistent with the rest of text being analysed, and that backtracking could be performed from the semantic interpreter, we have actually never been forced to such a procedure. Rather we consider only the possibility that the parser produces its f-structure representations deterministically. Here below are a selected set of such representations.

### 7.1. Adjuncts

The first example is a case of resultative with two adjectives in a sequence: notice that in Italian the first adjective cannot be positioned in front of the head Noun and must appear after it, as in *"Gino dipinse la rossa macchina"*. Thus the parser decides what to do on the basis of the lexical form: in particular, since the open function ACOMP is present, an adjective must be computed as argument and this must be the second, because otherwise, in case the first would be chosen as the argument, there would be no way to attach the second one as head modifier.

```
[gianni dipinse la macchina rossa gialla]/John painted the red car yellow
index:f4
pred:dipingere
lex_form:[np/subj/agent/[human], np/obj/theme_aff/[object], acomp/result/[_]_]
mood:ind
tense:pass_rem
cat:accomp
subj/agent:index:sn2
  cat:[human]
  pred:gianni:[gen:mas,num:sing]
  spec:def:'0'
  tab_ref:[+ ref, - pro, - ana, - class]
obj/theme_aff:index:sn4
  cat:[object]
  pred:macchina:[gen:fem,num:sing]
  spec:def:+
  mods:mod_r:index:saa22
    cat:[estens]
    pred:rosso:[gen:fem,num:sing]
    tab_ref:[+ ref, - pro, - ana, + class]
acomp/result:index:saa24
  cat:[estens]
  pred:giallo:[gen:fem,num:sing]
subj/nil:index:sn5
  pred:vbl
  controller:sn4
  tab_ref:[+ ref, - pro, - ana, - me]
aspect:accomp
```
7.2. Copulative Constructions

We shall look in more detail at an example with the verb "avere"/have, which is also treated as a case of copulative predicate. The first example is a complex utterance with a predicate "passare"/spend in the matrix clause that has an adverbial object. The structure of the copulative sentence is embedded in the sentential complement of "dire"/say and is an interrogative clause which says, "chi ha paura del lupo cattivo?"/who is afraid of the big bad wolf. The English translation requires the verb BE in place of "avere": as to semantic roles, the subject is interpreted on the basis of the contents of the open complement, Ncomp, which, according to the semantic features of its head, selects a semantic role for its controller. The analysis of the NP structure for psychic Noun "paura"/fear: "paura del lupo cattivo"/fear of the bad wolf" is interpreted as having a SUBJect "the wolf" who has as semantic role "malef(active)", an information encoded in the lexical form associated to the head noun "fear". Notice that the system only encodes information related to the arguments of a nominal predicate, whenever necessary, i.e. with deverbal nouns: either because the preposition is different from the one occurring with the corresponding verbal predicate, or for some special case, like this one, when the semantic role associated to the SUBJect NP modifier is not generally available. In all other cases, selectional inherent features are used to compute all PP modifiers/head relations. Thus, differently from BE, the verb HAVE will turn out with a subject which has been assigned a semantic role from the semantic categories of the predicative function it controls – but see chapter 1. The example is the following, "The two lazy little pigs spent their days playing and singing a song which said who is afraid of the big bad wolf"

[i due porcellini pigri passavano le loro giornate suonando e cantando una canzone che diceva chi ha paura del lupo cattivo]

index:f8
pred:passare
lex_form:[np/subj/actor/[human, animate], np/obj_avv/theme_unaff/[time], sadv/adv/modal/[mood, activity, state]]
mood:ind	tense:imp
cat:activity
subj/actor:index:sn1
cat:[edible, animate]
pred:porcellino:[gen:mas,num:plur,pers:3]
spec:def:+
card:2
mods:mod_a:indice:sa410
cat:[evaluative]
pred:pigro:[gen:mas,num:plur]
tab_ref:[+ ref, - pro, - ana, + class]
obj_adv/theme_unaff:index:sn4
cat:[time, duration]
pred:giornata:[gen:fem,num:plur,pers:3]
spec:def:+
subj/poss:index:sn7
cat:[]
pred:loro
cat:[]
spec:def:+
In addition, we show an example with the same copulative verb, which has an idiomatic predicative function, a locative predicate of the object. The structure of AVERE would in this case look very similar to the one of verbs like SEE and CONSIDER, which however are transitive predicates and may be constructed in the passive. The sentence is the following, "Quick, little brother, open us [the door]! we have the wolf at our heels [we are chased by the wolf]". In the structure of the first sentence the lexical form of the verb AVERE/have where the idiomatic function is included. In order to compute the locative Pcomp as idiomatic, there is a further requirement in the lexical form that says what the actual pred should look like. We consider FORM as a dummified semantic role which prevents the argument of the predicate to be seen from the semantic module, and be consequently dropped. In other words, the idiomatic meaning is computed on a par with the literal meaning, except for the lexical form which contains a specific word form in order to trigger the required interpretation: it is exceptional if compared with the procedures for literal interpretation. The underlying
psychological motivations are thus the following: in order to understand an idiomatic expression we need an extra work to be done by the mental processor, in case the idiomatic meaning has not already been learnt by the speaker: however, when it is, it will be listed among the lexical forms associated to the same predicate AVERE and it will be scanned before the lexical entry corresponding to the literal meaning. No extra work will be needed and the meaning will be built quicker for idiomatic expressions than for literal ones.

7.3. NP OBJecT Deletion and English Coordinate Structure Subject

In the utterance we showed above: "The two lazy little pigs spent their days playing and singing a song which said who is afraid of the bad wolf" there is a case of NP object deletion which is treated as a case of instantiation of variables in Prolog: the sentence has a coordination of gerundives, where the first gerundive has an intransitivized object NP which is built as a big Pro carrying the same index of the controller, the lexical NP object constructed in the following gerundive. In this case, we simply instantiate a variable which we make available at different levels of computation. Consider now the well known problem of empty NP subject in English coordinate constructions: since no empty category may be freely generated by the English grammar, unless it is a syntactic variable, the parse would fail. However, we assume that in the case of coordinate constructions, the empty subject is a bound empty pronoun which is taken care of by the level of annotated c-structure. When the parse of the first clause has been completed, we make available the index of the higher NP subject in case it might be used by the subsequent parse at sentence level. However, there must be a specific rule to parse a sentence which is an I_one_bar, i.e. a sentence which starts with a finite verb in English. This rule is required in order to prevent overgeneration: in this case we know that the subject NP is a controlled empty pronoun.

The example below, "she took it and put it in her hair", is taken from one of text we have analyzed, Text.1:

```
[she took it and put it in her hair]
main/prop:index:f7
  coord:index:f5
    pred:take
      lex_form:[np/subj/agent/[human], np/obj/theme_aff/[object, activity]]
mood:ind
tense:past
cat:result
subj/agent:index:sn3
  cat:[human]
    pred:she:[pers:3,gen:fem,num:sing]
case:[nom]
spec:def:+
tab_ref: [+ ref, + pro, + ana, + me]
antecedent:external
interpretation:specific
obj/theme_aff:index:sn7
  cat:[object, animate]
pred:it:[pers:3,num:sing]
case:[nom, acc, obl]
spec:def:+
tab_ref: [+ ref, + pro, + ana, + me]
antecedent:external
```
The following example shows the structure of a left clitic dislocated sentence which also has an inverted subject. This construction cannot be computed as a normal case of Clitic Left Dislocation, because the pragmatic overall import is taken up by the contrastive or emphatic inverted Subject. Thus, rather than representing a case of Topic reinforcement by Left Dislocation, where the Subject would be in its canonical preverbal position, this construction is a case of Focalization on the Subject. The proper noun indicates that the individual being introduced at this point of the story is already known in the world, this is why the function Subj_top is used rather than Subj_foc, which is used for indefinite NPs. Focalization will cause a Shifting to take place at discourse level and Trabucchi will become the new main topic of discourse. Notice that the resumptive clitic has been discarded from the representation after binding has appropriately taken place at c-structure level. In this way semantic roles are directly transferred onto the left dislocated NP, which also preserves a pragmatically marked functional label. In a canonical Clitic Left Dislocation, however, the clitic would be computed as the argument of the matrix predicate and binding would take place as with wh- words.
The example says, "In Verona, Alberti's college had been inherited by Trabucchi, and with the college he had inherited the chair of the Fair".

The example says, "In Verona, Alberti's college had been inherited by Trabucchi, and with the college he had inherited the chair of the Fair".

[In Verona il collegio di Alberti lo aveva ereditato Trabucchi e col collegio aveva ereditato la presidenza della fiera]
7.5. There Sentences, Locative Inversion and Reversed Constructions

We will now discuss an example of presentative utterance, a case of There sentence, where the locative clitic "ci" appears, and is the following: "There were once three little pigs who lived happily in the countryside". In order to compute "ci" as expletive clitic we interpret it as Pcomp thus satisfying the lexical form requirements at structural level, while at the same time assigning it the dummy function Form. The adjectival phrase "felici" corresponding to the adverbial phrase "happily" is computed as a case of open adjunct which is controlled by the subject which in turn is in a control chain with the higher matrix subject. The same thing would happen with the corresponding English example, where "ci" is rendered by "there", and the adverb is a case of subject controlled open adjunct.

[ci erano una volta tre fratelli porcellini che vivevano felici nella campagna]
7.5. Long Distance Dependencies

The following example is a question, where the syntactic controller, the FOC of the indirect question is singular in number and the SUBJect NP in inverted object position is also singular: the extraposition variables are bound in the preverbal NP and the interpretation call causes this NP to be interpreted as object. The utterance says: "Then tired to make merry, they decided to go and see what was doing their little brother".

```plaintext
scomp/proq:focus:type_focus:question
index:sn248
  cat:[object]
  pred:cosa:[pers:3,gen:,num:sing,case:acc]
  tab_ref:[+ ref, - pro, - ana, - me]
index:f48
pred:fare
  lex_form:[np/subj/agent/[human,animate], np/obj_adv/theme_unaff/[]]
mood:ind
tense:imp
support:stare
cat:extensional
foc/agent:index:sn343
  cat:[human, animate, relational]
  pred:fratello:[pers:3,gen:mas,num:sing]
  spec:def:+
  subj/poss:index:sn344
  cat:[]
pred:loro
  spec:def:+
  tab_ref:[+ ref, + pro, - ana, + me]
antecedent:sn244
interpretation:specific
```
mods:mod_a:index:sa431
cat:[evaluative]
pred:piccolo:[pers:3, gen:mas, num:sing]
tab_ref:[+ ref, - pro, - ana, + class]
obj/theme_unaff:index:sn257
cat:[object]
pred:vbl:[case:acc]
controller:sn248
tab_ref:[+ ref, + pro, - ana, - me]
aspect:activity
Chapter 6

ANAPHORIC BINDING

1. INTRODUCTION

LFG theoretical framework assumes that pronoun-antecedent relations are computed at f-structure level: quantifiers, on the contrary, are computed at s-structure level. That is not our approach. In non modular frameworks like DRT or Situation Semantics pronouns are visible at the same level of processing in which wh- words and quantifiers are. This is also partially the view held by GB scholars, who regard binding as a single phenomenon which however applies diversified principles for anaphors, pronouns and "variables", be they wh- or quantified variables. In traditional LFG framework f-structures are mapped into s(semantic)-structures where quantifiers are made visible. However, as we shall see, since quantifiers may modify the number of indefinite NPs in the scope of universal quantifiers, and this could happen at intrasentential level, we think it important to compute quantifier scope before anaphoric binding. In this way, pronouns which are contained within the same utterance may be syntactically bound to their indefinite NP antecedent even though their grammatical number would not allow it, as for instance in "The women who were carrying a baby each dropped them". Here we see that the indefinite NP "a baby" has singular grammatical number: however, by virtue of scope dependence on the universal distributive floating quantifier "each", it comes to be interpreted as a set with the same cardinality assigned to the NP "the women" which acts as a syntactic controller of the quantifier "each" computed as open adjunct, at discourse level. More about this and related problems in the following chapter.

In our system, wh- elements are bound at c-structure level and make available syntactic chains to the next level of computation, f-structure. Quantifier raising is applied at f-structure level, seen that scope domains coincide with the f-domains of closed grammatical functions. In turn, quantifier scope is made available to discourse level procedures in order to compute discourse anaphora adequately. All this happens before semantic interpretation is triggered, where Logical Form is constructed. Since quantifiers are paramount both for utterance and discourse level we discuss them in one separate chapter, Chapter VII.

We shall now deal with anaphoric binding at utterance level.
At the core of the algorithm there is a dichotomy between pronouns, P and anaphors, A. We might consider the following properties as relevant for a separation of P or pronominal elements into classes according to a feature system, where the a. property is the basic one distinguishing A from P as listed in Table 26.

Properties from a. to c. are only sensitive to anaphors; property d. sets anaphors apart from pronouns. Properties e. to h. are only sensitive to pronouns. Italian differs from languages like English because it possesses a greater number of lexical items to express the properties listed below. For instance, English has only one kind of pronoun to express properties f., h., i.e. lexical pronouns; and another kind of pronominals which expresses property e. - i.e. big Pro. In most languages, only non anaphoric pronominals possess property g..

Contrary to what expected, in English we can have 1st and 2nd person reflexives which can be bound freely in the discourse; and 3rd person reflexives which can be bound in the discourse according to Zribi-Hertz(1990) principles of the Subject of Consciousness as antecedent and Domain of Point of View as discourse relevant domain. This cannot be covered by local utterance level criteria like the ones introduced here - but see Chapter VIII.

Italian splits properties f-h which in English are associated with a single class, into two different classes: one kind of pronominals expresses properties h. - i.e. independent lexical pronouns. Another kind of pronominals expresses property f. - i.e. little pros and clitics. In turn, clitics and little pros can be treated as syntactic variables and be bound at c-structure level: in this case they would be seen as foot of a syntactic chain.

Table 26. Anaphoric Binding Constraints and Referential Properties

<table>
<thead>
<tr>
<th>Type of Anaphora</th>
<th>a.</th>
<th>b.</th>
<th>c.</th>
<th>d.</th>
<th>e.</th>
<th>f.</th>
<th>g.</th>
<th>h.</th>
</tr>
</thead>
<tbody>
<tr>
<td>short anaphora</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Long anaphora</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>proprio</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Independent pros</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>clitics</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>pro, PRO</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
</tr>
</tbody>
</table>

Italian is more restrictive with reflexive pronouns, which cannot be used in the way Zribi-Hertz has shown: perhaps this is due to the fact that possessing a higher number of lexical items has a more restrictive system of referentiality as a counterpart. Thus, it would seem that a system for pronominal binding in English would necessarily treat reflexive pronouns ambiguously, and in case they are left unbound at sentence level it would require the computation of discourse structure, where such discourse notions as Subject of Consciousness
and Domain of Point of View apply – but see the following chapters. Italian, on the contrary allows modularity to the system, keeping the domain of reflexive pronouns - only sentence level - distinct from the domain of lexical free pronouns. However, the morphological word-structure of English reflexive pronouns is different from that of Italian counterparts: the former are constituted by a free lexical pronoun "him", and a reflexive element "self", as in "himself"; the latter are built up with two reflexive elements "sé"+"stesso" as in "se stesso", but is otherwise similar in remaining cases, i.e. "me stesso", "lui stesso" etc.

Italian has three reflexive elements, one of which is a possessive anaphoric pronoun, "proprio", then a short distance reflexive pronoun, "se stesso", and a long distance one "sé". The short distance reflexive "se stesso" has a distribution that is somewhat similar to the English reflexive "himself", though there are differences between the two: "se stesso" may corefer with a coargument and its antecedent must appear in the same minimal finite domain. This does not apply to "himself" which could be bound from the text, as shown by Zribi-Hertz(1990) and others(Pollard & Sag). On the contrary with long distance reflexive "sé" the antecedent must be a subject: or else it must be "governed" by a preposition, i.e. it must be contained in an OBLique or an ADJunct PP. As to long distance possessive anaphoric pronoun "proprio", it is subject oriented and clause bound; but in lack of an adequate antecedent it may search out of its clause (complement or adjunct or coordinate) for its antecedent. In addition, there is the multivalued clitic "si" which may be assigned the following functions "middle", "passivizing", "reflexive", "impersonal or arbitrary", its behaviour being strictly determined by the verb predicate to which it is associated. It may be assigned nominative case if it is computed as impersonal SUBJect; none of the remaining reflexive elements may be used as SUBJects.

Italian also has four pronominal elements, one of which is a possessive pronoun, "suo", then a Null Subject pronoun which behaves very closely to the English personal pronouns; finally a set of lexically independent pronouns which are used for contrast or emphasis. All these pronouns look for their antecedent outside their minimal containing clause. As to the possessive "suo", it behaves quite differently from the corresponding English "his". "His" can be bound by an OBJect coargument, when it is contained in the SUBJect NP as for instance in "His daughter loves John". This is not allowed in Italian, the SUBJect being a strong domain for reference. The same applies to "proprio", which being a possessive anaphoric pronoun is sensitive to the grammatical function it is contained in. However, there is one exception, and this is the case constituted by psychic verbs, whose SUBJect is characterized by a thematic role which is very low in the hierarchy of theta-roles: it is an (emotional) Theme, as for instance in "La propria salute preoccupa ognuno-Gianni"/self's health worries everyone-John. Coreference between "proprio" and "ognuno" is allowed, but is banned with "Gianni" as antecedent. This does not apply to the corresponding "La sua salute preoccupa ognuno-Gianni"/His health worries everyone-John, where no such coreference is allowed.

As Dalrymple(1990) comments, "constraints on anaphoric binding are lexically associated with each anaphoric element. In fact generalizations have been noted that deal specifically with the lexical form of the anaphoric element: elements of a particular morphological form are usually or always associated with particular sets of anaphoric binding constraints"(ibid.,2). It is interesting to note that such functional notions like "subject", "tense" and "predicate" are essential in defining these constraints, they all "denote some syntactically or semantically 'complete' entity"(ibid.3). As Dalrymple comments, "In a complete, consistent f-structure, a PRED denotes a syntactically saturated argument structure;
presence of a SUBJ entails a predication involving some property and the subject; and presence of TENSE indicates an event that has been spatiotemporally anchored. The 'complete' entities are the relevant domain for binding conditions" (ibid.3).

The grammatical function of the antecedent is part of the antecedent constraints: an anaphor must or may be bound to a SUBJECT. Also the domain in which an anaphor must find its antecedent is always constrained relatively to either the main predicate which the anaphoric element is an argument of: the minimal domain with a subject containing the anaphor; or the minimal tensed domain containing the anaphor. These can be regarded as domain constraints. Moreover, we may think of two kinds of binding constraints: positive and negative constraints. In line with Binding Theory of Chomsky(1981), 'reflexive' is an element which must be bound or must have an antecedent within some syntactically definable domain.

On the contrary, 'pronominal' is an element that must be free, or be noncoreferent with elements in some syntactically definable domain.

However if we look at "proprio", we see that it must be bound in its minimal tensed domain, but in case no suitable antecedent is locally available, it may look outside and be assigned an antecedent or even receive arbitrary reading at certain semantic conditions, definable in terms of tense, subject, aspect. As Dalrymple suggests, there may be a typology of constraints rather than a typology of anaphoric elements (ibid.,4). In previous works (Hobbs, 1978; Ingris, 1989) only syntactic constituency and c-command was considered, but recent work in linguistics has clearly proven this approach to be insufficient. In particular, both Chomsky's (1981) and Manzini's (1983) theory wrongly predict the grammaticality of sentences such as,

1. *I persuaded/told the boys that each other's pictures were on sale.

2. The boys thought that each other's pictures were on sale.

were the reciprocal anaphor "each other" lacking an accessible subject in its Domain Governing Category (we will not enter into a discussion of Chomsky's binding principles nor in Manzini's modifications - see Giorgi(1984)), its sentence (1) is predicted to corefer freely, hence the object NP of the matrix clause is treated as a possible antecedent on a par with the subject in 2. Since it is wrong to say that anaphors can corefer freely, what is needed is a theory of Long Distance Anaphor, which is able to explains how the anaphor is still subject to a number of binding constraints.

Here crucially, the terms long-distance and short-distance are not used in the way in which Ingris does, and do not apply to pronouns: in particular personal pronouns, cannot be treated as long-distance anaphors (see, ibid.263) since they can pick up an antecedent in any domain whatsoever, outside their minimal domain, the clause in which they are contained - including their matrix clause and the discourse. On the contrary possessive anaphors and reflexive anaphors which count as long-distance anaphors must be bound by an antecedent before leaving their matrix clause - in other words they cannot be bound by a discourse-level antecedent with the notable exceptions of LDRs. This applies to lexical personal pronouns as well as to morphologically unexpressed personal pronouns like PRO/pro which can be bound in a superordinate clause or in the discourse. However, in constructions involving picture noun phrases, reflexives allow non-local antecedents, and rather than being subject to syntactic constraints they seem to obey discourse constraints as Pollard and Sag(1989) discuss in their work.
In the same way it is possible to explain why in example 3. below, with an experiencing verb, the anaphor contained in the subject NP can be bound by the object which does not c-command it, showing that this notion is not sufficient in itself to tell it apart from 4. where the same structural conditions do not apply:

3. Each other's pictures pleased the boys.
4. *Each other's wives murdered the men.

In other words, each other seems to behave like a long distance anaphor, i.e. a possessive pronoun like proprio in Italian, with some exceptions. The lack of c-command is clearly shown in case a quantifier appears as experiencer,

3i. La propria salute preoccupa ognuno/One's health worries everyone

In sum, the two Italian anaphors seem to behave in the same way: sé, must always be governed by a preposition and se stesso can also be governed by a verb and it can look for a subject in a superordinate clause being classified as subject-oriented, i.e. [+SUBJECTIVE]. In addition, se stesso can be bound by coargument grammatical functions and is strictly local. Proprio, being a mixture of both, can be bound by coarguments beside the subject, and can look for a binder in a superordinate clause.

In addition, with psychic and experiencing verbs the anaphor contained in the theme/subject can be bound by the experiencer/object - the same does not apply to the pair agent/subject and theme/object of transitive verbs. In other words, candidates for antecedenthood must be selected in accordance with their status as grammatical function and thematic role. The same applies whenever the experiencer is the subject of raising verbs - when better antecedents lack - like seem/sembrare.

5a. *La propria salute preoccupa Marco/ *La figlia della propria moglie odia Gino
b. La propria salute preoccupa ognuno/ *La figlia della propria moglie odia Gino

6a. La malattia della propria moglie preoccupa molto Marco/The illness of self's wife worries Mark a lot
b. *La propria moglie odia Gino/ *La figlia della propria moglie odia Gino

7a. His wife hates John
b. *His father hates/worries everybody

As these examples show, quantifier status is a very important parameter in order to assess the status of candidates for antecedenthood. Also, language dependent differences are visible from the paradigm: Italian possesses a wider range of pronominals and anaphors and allows binding of a possessive within the same clause as it becomes more deeply embedded. However deep embedding does not rescue 6b: thematic relations are the relevant criterion in this case. In the corresponding English examples, binding is performed at reverse conditions: not being a quantifier is the only requirement.

Belletti and Rizzi(1988) propose for these kind of examples and for others that Principle A of the Binding Principles be an "anywhere" principle (ibid.,314), in the sense that it can apply at D-structure, where the subject NP is contained within the VP, thus justifying the fact that the anaphor contained in the Subject is bound before it moves to its S-structure position. Obviously, this is also relevant for sentences like

8. Which picture of himself do you think [that Bill likes e best]?

where Move-a has destroyed the well-formed binding configuration by extracting (the constituent containing) an anaphor from the c-domain of its antecedent. In a framework like
LFG, however, no such "anywhere" principle could be made to work since categories which must be bound are only visible at one level of representation. In particular, syntactic variable are visible at c-structure and this is where they must be bound by their controller; lexical anaphors are only visible at f-structure where they must be given an antecedent in their nuclear f-structure. For examples like 7. above, there is a variable binding operation that takes place at c-structure level between the FOCus wh- phrase and the empty element in the embedded clause; when we get to the next level of representation, the anaphor contained in the FOCus is part of a syntactic chain, i.e. is included in a non-argument function, the discourse function FOCus, and is bound to an argument function the OBJect of the predicate "LIKE" which also assigns it its theta-role. Since the argument function is the place in which the FOCus will be interpreted - they bear the same index - they can be bound under f-command, as we shall see.

3. F-COMMAND, OPERATOR BINDING, AND BACKWARD PRONOMINALIZATION

In order to perform binding procedures, all functional structures are transferred into a graph with arcs and nodes, where arcs contain grammatical function. Arcs also relate each function to its mother node, in this way allowing to compute all functions contained in an upper function: this is the crucial notion for the definition of f-command domains (see Bresnan,1982).

The algorithm uses f-command rather than c-command and obviation to prevent clitics and lexical pronouns to look for antecedents in the same f-structure in which they are contained. Formally it is expressed as follows:

9. F-command

For any occurrences of the functions \( \alpha, \beta \) in an f-structure F, \( \alpha \) f-commands \( \beta \) iff \( \alpha \) does not contain \( \beta \) and every f-structure of F that contains \( \alpha \) contains \( \beta \)

As for obviation, it applies to big PROs, to little pros, and to lexical pronouns: it follows the more elaborate framework resulting from Bresnan et al. (1985) where pronouns which must obey the Coargument Disjointness Condition (i.e. they may not be bound to an argument of the same predicate) are obviative and are marked [±NUCLEAR], thus meaning that they may or may not appear in the same syntactic nucleus as their antecedent. However, an ADJunct is never part of the nucleus so that a pronoun is allowed, in English but not in Italian.

10a. John talked to Mac about himself/him
    b. John talked to Mary about herself/her
11. *Gino ha parlato a Mario di lui/ sé / se stesso
12. John wrapped a blanket around him/himself

The English reflexive pronoun himself is [±NUCLEAR] and must find an antecedent within the same nucleus containing the pronominal and a subjective function; while him is [-NUCLEAR]. The open ADJunct "around himself" however has a subjective function bound to a SUBJECT whenever it is preposed, and bound to functions included in the VP, or to the
SUBJect when sentence final. In the example, the OBJect is not a semantically adequate controller, and is thus eliminated in favour of the SUBJect, John. However, in order to explain why both *him* and *himself* are allowed in an XADJunct we need to allow for binding both by the OBJect and by the SUBJect, which amounts to saying that PP may be attached within the VP or at a higher level. When attached to the SUBJect, also the event or state of affairs described by the predicate will automatically be in the scope of the ADJunct. However, when they are attached within the NP, the SUBJect is not in the scope of the ADJunct. In addition, prepositions may be regarded as either semantically attracted by a given predicate or semantically independent thus imposing attachment at a higher level. In case the XADJ is bound by the OBJect, as in the examples above, only the reflexive would be bound to the SUBJect, the pronoun would receive both a bound and an obviative reading. Open adjuncts are computed differently from OBLiques which are closed functions and do not allow for functional controllers - see Chapter V. Both reflexive and pronoun can be bound by the SUBJect. Beside, English pronoun *him* is not obviative like the corresponding Italian one.

13i. Gino *ha visto un serpente vicino a lui/*i/*j (John has seen a snake near him)
   ii. Gino *ha visto un serpente vicino a sé/*i/*j (John has seen a snake near "sè")
   iii. Gino *ha visto un serpente vicino a se stesso/*i (John has seen a snake near himself)

14. They saw snakes near them/themselves/each other

Thus, the relevant domain for anaphors and pronouns contained in nominal f-structures is not the f-structures directly containing them: this is due to their functional nature and not simply to structural reasons. As to reciprocals, reflexives, and possessives anaphors, they are all assigned SUBJECT function, thus counting as possible candidates for antecedenthood: but a conflict is raised here by the referential nature of anaphors which is marked in our system as minus referential [-ref] in their feature matrix, hence unable to become antecedents of themselves. This conflicting result works as a filter for anaphors at the structural level, erasing their ranking as candidates for antecedenthood but raising them out of their subordinate f-structure into the upper one: in this way, anaphors cannot be bound within their minimal f-domain but must be bound in the upper one, pronouns are left free to corefer.

At clause level, reflexive pronouns look for binders in the same f-structure in which they are contained. As said above, two kinds of anaphors must be taken care of: long anaphors like sé and short anaphors like *se stesso*. Only short anaphors can be bound by non-subjects and only long anaphors can be bound in an upper clause if no suitable binder appears in the local minimal one. The possessive anaphor *proprio* on the contrary partakes of features belonging to both short and long anaphors: it can use both a short and a long distance strategy; it is not SUBJective. We have established then that the lexical feature [-SUBJCTV] distinguishes short anaphors from long anaphors, which are marked [+SUBJCTV]. Summarizing, we have two sets of reflexive pronouns,

15a. subjective reflexive pronouns [+SUBJCTV] "sé"
   b. non-subjective reflexive pronouns [-SUBJCTV] "se stesso"

In addition, long distance anaphors like the possessive *proprio*, non specified as to SUBJectivity, behave both as long and short anaphor, according to the domain in which they can be bound, and are positively marked for [+pro, +ana]. As to Germanic reflexive
pronouns, they are all computed [+ref], thus allowing them to become discourse level anaphors.

An opposite behaviour is also testified by anaphors like lexical reflexives and possessives contained in a SUBJ ect, which can be bound within the same clause in English but not in Italian:

16a. *his father hates John [*Each other's wives murdered the men]
b. *his father hates everyone
17. his father saw John ≠ *He saw John ≠ Suo padre ha visto Gino

However as b. shows, the antecedent cannot be a quantifier.

4. OUR PROPOSAL

Our proposal takes into account the facts of Italian but also those of English, Norwegian and other languages as discussed by Enç (1989) or Dalrymple (1990). Binding is expressed by coindexation of a controller α and a controllee β, just like coreference between antecedent and pronoun, in a domain F - a complex f-structure, at the following conditions:

18. β is an f-structure [+ana] and is bound in its F-domain
19. β is an f-structure [+pro] and is not bound in its F-domain

The first part of the formulation accounts for the fact that an anaphor is in complementary distribution with a pronoun, i.e. that in the domain in which the anaphor must be bound the pronoun must be free, or not be bound. Now, the smaller domain is an f-structure with a SUBJ ect, be it an open or a closed f-structure. Obviation could be used to tell obviative pronouns or pronominals in a certain domain, an obviative proposition, that is a clause nucleus; however either formulations of obviation do not account for the behaviour of NPs. Nothing seems required for referential expressions at this level, where no mention is made about the antecedent.

20. F is an F-domain iff
   α f-commands β in F and there is an I between α and β such that I is licensed

The second part of the formulation, says that the structure in which the antecedent and the anaphor must be bound is the one containing a SUBJ ect function - this is derived from licensing condition: in an NP the F containing the head, in a clause, the F containing the SUBJ ect of the clause, in an ADJunct the one containing the PRO, in an open function, the open function itself.

21. F-command:
   A function α f-commands a function β in F iff
   a. α is not contained in β, and β is not directly contained in α, β = SUBJ ect
   b. every f-structure of F that contains α contains β
   b1. β may contain α in F iff α is in a weak RD
   c. a function β is directly contained in a function α if
      β is a subsidiary f-structure of a function α

The reference to the subject in a. captures the well-known restriction that the subject is not accessible to itself - the remaining arguments/adjuncts of the head Noun may be bound by the subject; as well as the i-within-i principle recast.
In a., the antecedent/binder cannot be contained in the f-structure of its bindee: in other words, the relation is asymmetric; also the bindee cannot be directly contained in the f-structure of the antecedent but it must constitute a separate f-structure. This is trivial, but requires the formulation of a notion, "directly contained", which divides f-structures contained in complements and adjuncts of a head from their governors.

The b. clause only applies when the bindee is contained in the same F that contains the binder, but the binder is down in a separate f-structure which is open. However, for licensing conditions on F given below, obliques are not regarded as possible F-domains.

22. Licensing conditions for an Indexing I of \( \alpha \) with \( \beta \)
   1. i. must be lexically free;
      ii. it is the SUBJect
      iii. it is in a strong RD
      iv. its \( \Theta \)-role is superior in the following hierarchy:
         agent > benefactive > recipient/experiencer/goal > causer_emotional >
         instrumental > theme/patient > locative

   Here we see that iii. differentiates between an ADJunct PP and a predicative one, in the sense that the anaphor contained in an adjunct PP is bound to the SUBJect of the higher strong RD, whereas an anaphor contained in an open PP is bound locally to the closer function.

   2. otherwise,
      A. a function \( \beta \) is free in the discourse if F is a weak RD,
      B. a function \( \beta \) is coreferent in the discourse if \( \beta \) is in a strong RD.

23. A function is lexically free iff,
   - it is argumental
   A function is lexically bound iff,
   - it is \( \Theta \)-empty, the existentially bound argument
   - it is an expletive (no PRED, but FORM)
   - it is a quasi-argument

   An empty function is an implicit argument of a predicate and is always computed as a generic existential quantifier: arbitrary reading may ensue from the RD; an expletive is always a Form; a quasi-argument is the dummy pronominal generated as SUBJect of atmospheric weather verbs.

24. A R(eferential)-D(omain) is an f-structure specified for referential energy:
   i. it is strong iff a. it is a closed function;
      b. it is referentially transparent;
   ii. it is weak iff a. it is an open function;
      b. it is referentially opaque.

   iii. Referential energy:
      a. for clause nuclei(where a SUBJect is obligatory) is expressed by atomic attribute-value pairs: TENSE=[±REF] {past tense individuates a specific reference time}, MOOD [±REAL] {real mode is assertive and implies the truth of the proposition - at least on part of the speaker}, CLASS[±IMPLIC] {implicative verbs imply the truth of their complements and may be interpreted referentially - also factivity is included}, ASPECT [±PERF] {perfective aspect implies the existence in the world of the object predicated by the verb};
      b. for NP heads of relative predicative adjuncts
SPEC = [±DEF/O], INDIV [±GEN], [±ref].
c. transparency obtains whenever the features have positive value.

5. THE ALGORITHM FOR ANAPHORIC CONTROL

Two structures are built from the output of the grammar: annotated c-structures, i.e. a directed graph which can be traversed primarily through syntactic constituents; and a list of the functional schemata associated with semantic forms - in other words, all PRED expressions with a list of semantic attribute-value pairs, i.e. the f-structure mapped from the previous structure, where pronominal binding is computed. The algorithm applies to a completely parsed structure which is a graph translating the annotated c-structure of LFG into the f-structure. The algorithm uses all functional information associated to the f-structure of a grammatical function and its semantic role. The definition of domains is based crucially on the notion of f-structure and governors are derived from grammatical function and semantic role, as we shall describe in more detail below.

When a pronoun is encountered, the algorithm moves up to the left of its minimal domain, the closest f-structure containing it and stops in the first superordinate f-structure; on the contrary, with anaphors, the search is to the left within the same f-structure containing it, unless it is contained in a SUBject. It is worthwhile reminding that at f-structure level the VP node disappears and an OBJect NP appears at the same level of a SUBject NP. F-structures contained in a nominal f-structure behave differently due to their grammatical function as discussed below.

In line with Bresnan et al. (1985) and contrary to the proposal contained in Dalrymple (1990) we use functional features as lexically specified properties of individual anaphoric elements. These features both account for and translate lexical category, in this way directly triggering the binding algorithm that fires a certain procedure whenever either a [+pro] or a [+ana] feature is met in the referential table associated to a certain f-structure. Features also serve to restrict the type of possible antecedents in terms of reference to the SUBject; to set up a hierarchy for antecedenthood in which possible antecedents are ranked according to their associated grammatical function and semantic role; to unify morphological features checking for agreement in person and number, and selectional restrictions imposed by inherent semantic features; to tell apart quantifiers and quantified NPs which cannot be used as antecedents in backward pronominalization. A complete list of features is given below. The notion of f-command is central to the working of the algorithm. A pronoun must look for an antecedent in a certain F(unctional)-domain which is local for the anaphors and non-local for pronominals.

Rather than defining F-domains as nuclear functions, containing a SUBject function, we simply incorporate the reference to the SUBject in the f-command definition. In this way we obtain a similar behaviour both for anaphors contained in an NP and those contained in a S.

In order to do this, we require for an f-structure which is marked as [+ana] to be raised out of its level into the upper one where it will look for the antecedent. This works out right both for anaphors contained in open or closed functions with two notable exceptions; whereas anaphors which are SUBjects and are contained in a SUBject function must be raised out of the higher SUBject, before looking for their antecedent; anaphors which are objects or
adjuncts can be bound by a SUBJect within a NP structure in case there is one or else look out of this f-domain. Consider the three examples below:

25a. John’s book about himself/*herself

b. Pictures of himself upset Bill/*My pictures of himself upset Bill

c. His pictures of himself upset Bill

Beside we also must ensure that an anaphor contained in an ADJunct does not pick up the head noun as its antecedent: this is captured by the notion of subsidiary f-structure or directly contained as we defined it.

When the closest f-domain is individuated, the algorithm will look for the antecedent which must be semantically compatible and adequate. In case there is none - at Sentence level - another f-domain may be present and will be accessible to all anaphors which are also marked [+pro], i.e. long distance anaphors and big PROs. The antecedent may be any function available at that level, unless [+subjective] is present as feature within the referential table associated to the anaphor. In this case, only the SUBJect will be selected as antecedent.

Anaphors contained in SUBJects undergo a number of restrictions both in Italian and English to the effect that only SUBJects associated to certain semantic roles may look for antecedents within the same f-domain: in particular they must be computed lower in a semantic roles hierarchy where for instance, Experiencer is higher than a Causer_Emotion which are the two roles associated respectively to the OBJect “Bill” and the SUBJect “pictures of himself” in example 25.

Other exceptional procedures are required in case the antecedent is a quantifier or a quantified NP, or in case the anaphor or the pronoun is contained in an opaque Referential Domain - i.e. an f-domain which has its referential attributes, TENSE, MOOD, ASPECT, marked negatively.

Whenever an antecedent is found - selected by the procedure poss_ante which checks for the presence of the feature [+ref] - its ranking is checked for agreement as well as its features: the interaction with binding principles determines the possibility for an OBJect referential expression to act as binder of long distance anaphors. In other words, binding works by default according to the principle "bind anaphors as soon as possible". On the contrary pronominal coreference imposes the algorithm to pick up a certain referential expression as possible candidate and to reject other referential expressions owing to their ranking in the hierarchy. Only one antecedent is selected for [+ana] elements; with [+pro] more than one antecedent is selected according to the rules and to the antecedents available.

Whenever a pronoun is left unbound the algorithm adds an instruction "resolve(x)", which is used to trigger the anaphoric binding algorithm at discourse level(see Chapter VIII). The remaining pronouns and anaphors are assigned a couple of indices: their own and the one of their antecedent and binder. Following recent work by Enç (1989) who discusses a pronominal system for natural languages made up of seven classes, we built one made up of four classes for Italian - Chomsky's system based on two classes, anaphors and pronouns is insufficient. To be added to these four classes - which include anaphors and nouns (common, proper) - there is one class for pleonastic lexically unexpressed pronouns constituted by verbal agreement in Italian, deprived of deictic import. Pronouns can be lexically specified or not, this being expressed by a feature introduced in Bresnan (1982), [±MU] (Morphologically Unexpressed), which however we use positively as [±me], i.e. morphologically expressed or not. Thus, big PRO's resulting from tense specification which can be subject to anaphoric
control - in LFG PROs are subject to structural or lexical functional control - are differentiated from little pro's by the fact that the former are marked [+ana], and the latter are marked [-ana]. These are differentiated from clitics and independent lexical pronouns by the fact of being [-me], whereas the latter are [+me]. Besides, clitics - which are always unstressed - are marked [+ana], whereas stressed personal pronouns are [-ana]. Epithets contain a deictic or a determiner feature specification. Deictic pronouns like "this, that" contain the feature [+class] in place of [-me]. Pronominal quantifiers are marked [+pro] [+PART]. We give below a complete classification in features of all pronominal and nominal expressions as computed by the system, as a translation of lexical category together with features from SPEC, and NUMBER.

Other features are attributed to nouns by their SPECifier: in particular articles are translated into [+DEF], numbers into [+CARD], quantifiers into [+PART]. The lack of determiner or the null determiner is marked by the presence of the feature [0 DEF]. The feature [+PART] is also assigned when a prepositional marker "di/"/of is used to indicate an indefinite or a definite unspecified quantity (corresponding to the English "some, a (little) bit of"). This information is recorded under a separate functional node, the SPECifier: it is recovered by the discourse module and passed on to the semantic module for interpretation.

<table>
<thead>
<tr>
<th>Category</th>
<th>Referential</th>
<th>Pronominal</th>
<th>Anaphoric</th>
<th>Morphologically Expressed</th>
<th>Subject-bound</th>
<th>± Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>big PROs</td>
<td>+ref</td>
<td>+pro</td>
<td>+ana</td>
<td>-me</td>
<td></td>
<td></td>
</tr>
<tr>
<td>little pros</td>
<td>+ref</td>
<td>+pro</td>
<td>-ana</td>
<td>-me</td>
<td></td>
<td></td>
</tr>
<tr>
<td>clitics</td>
<td>+ref</td>
<td>+pro</td>
<td>+ana</td>
<td>+me</td>
<td></td>
<td></td>
</tr>
<tr>
<td>independent pronouns</td>
<td>+ref</td>
<td>+pro</td>
<td>-ana</td>
<td>+me</td>
<td></td>
<td></td>
</tr>
<tr>
<td>common nouns</td>
<td>+ref</td>
<td>-pro</td>
<td>-ana</td>
<td></td>
<td>+class</td>
<td></td>
</tr>
<tr>
<td>variables</td>
<td>+ref</td>
<td>-pro</td>
<td>-ana</td>
<td>-me</td>
<td></td>
<td></td>
</tr>
<tr>
<td>expletives</td>
<td>-ref</td>
<td>-pro</td>
<td>-ana</td>
<td>-me</td>
<td>-class</td>
<td></td>
</tr>
<tr>
<td>proper nouns</td>
<td>+ref</td>
<td>-pro</td>
<td>-ana</td>
<td>-me</td>
<td></td>
<td>+class</td>
</tr>
<tr>
<td>quasi-arguments</td>
<td>+ref</td>
<td>-pro</td>
<td>-ana</td>
<td>-me</td>
<td></td>
<td></td>
</tr>
<tr>
<td>quantifiers</td>
<td>+ref</td>
<td>-pro</td>
<td>-ana</td>
<td>+me</td>
<td></td>
<td></td>
</tr>
<tr>
<td>long anaphors</td>
<td>-ref</td>
<td>+pro</td>
<td>+ana</td>
<td>+subjectv</td>
<td></td>
<td></td>
</tr>
<tr>
<td>short anaphors</td>
<td>-ref</td>
<td>-pro</td>
<td>+ana</td>
<td>-subjectv</td>
<td></td>
<td></td>
</tr>
<tr>
<td>proprio</td>
<td>-ref</td>
<td>+pro</td>
<td>+ana</td>
<td>+me</td>
<td>+subjectv</td>
<td></td>
</tr>
<tr>
<td>deictic</td>
<td>+ref</td>
<td>+pro</td>
<td>-ana</td>
<td></td>
<td>+class</td>
<td></td>
</tr>
</tbody>
</table>

In addition, common nouns are differentiated from proper nouns by the feature +CLASS for the former and -CLASS for the latter, indicating that common nouns are used to denote classes or properties of individuals, as opposed to proper nouns which should pick out names for individuals. Moreover, common nouns are specified in reference by definiteness, whereas proper nouns use definiteness only redundantly - in Italian a proper noun may be preceded by
Anaphoric Binding

a definite article. When a noun is recognized as proper, this feature is discarded. Proper nouns are assigned a higher score than common nouns, as candidates for antecedenthood. Cardinality is marked by Number, which adds the information that a Singular, Definite, Specific noun phrase is to be interpreted as a unary set or singleton of the class of objects or individuals denoted by the noun, i.e. there is only one member referred to by the noun phrase in universe of discourse that we want to pick up. Plural noun phrases are treated differently, i.e. as quantified NPs.

6. THE CORE ALGORITHM

We list here below the basic algorithm in its Prolog formulation: as we said previous it applies to f-structures which are compiled as a directed graph, and accessed by an algorithm with performs graph search.

As to f-command, the algorithm should indicate all the nodes containing the pronominal or anaphoric node within their f-domain, i.e. all the nodes that f-command them. The f-commanding nodes may constitute potential antecedents or binders of the bindee.

The rule implementing f-command recovers the index and the grammatical function of the f-commanding element. In addition, it also marks the distance in a numerical index in terms of levels of f-structures intervening between the binder and the bindee.

In this way, it is possible to impose conditions on the levels in the graph to be searched for by the rule of f-command: for instance, in case one needs to exclude from the search all the nodes belonging to the same level, it is sufficient to establish that the level distance index be different from zero, as a condition to the output of the rule. In this way, the rule will fail when trying to bind a given element at zero distance, and the backtracking mechanism of Prolog will cause the algorithm to start out a new execution of the rule at a different node of the graph.

The following constitutes the basic realization of f-command in DAG’s terms:

\[
\text{f-command}(\text{Alpha}, \text{Beta}, \text{Level}):= \\
\text{Fs} : \text{Function} : \text{Beta}, \\
f\text{-command}\text{-subs}(\text{Fs}, \text{Alpha}, 0, \text{Level}).
\]

this predicate is used to express the fact that the f-structure Alpha must f-command the f-structure Beta, where colon (:) is an infix right-associative operator. Then we note with Fs, the f-structure which immediately dominates Beta.

\[
\text{f-command}\text{-subs}(\text{Fs}, \text{Alpha}, \text{Lev}, \text{Lev1}):= \\
\text{Fs} : \text{Function} : \text{Alpha}.
\]

which has success in case there is an f-structure Alpha also directly dominated by Fs, whereas the following,

\[
\text{f-command}\text{-subs}(\text{Fs}, \text{Alpha}, \text{Lev}, \text{Lev1}):= \\
\text{L1 is Lev}+1,
\]
Fs1 : Function1 : Fs,
f-command-subs(Fs1,Alpha,L1,Lev1).

will be verified in case the f-structure Alpha is dominated by an f-structure Fs1 which in turn dominates Fs thus including Beta. This is the complete version of the f-command predicate in the parser:

\[ f_{\text{structure}}(\text{Index},F_R,\text{Node}) :- \]
\[ \text{node}(\text{Node})::F_R::\text{index}::\text{Index}. \]

\[ f_{\text{command}}(\text{Alpha},\text{Alpha}_Funct,\text{Beta},\text{Level}) :- \]
\[ f_{\text{structure}}(\text{Beta},F,N), F= \text{subj}/_, \]
\[ \text{node}(\text{N1})::F1::\text{node}(\text{N}), F1 = \text{subj}/_, \]
\[ \text{node}(\text{N2})::F2::\text{node}(\text{N1}), \]
\[ f_{\text{c}}(\text{N2},F2,\text{Alpha},\text{Alpha}_Funct,0,\text{Level}_x), \]
\[ \text{Level is Level}_x + 2. \]

\[ f_{\text{command}}(\text{Alpha},\text{Alpha}_Funct,\text{Beta},\text{Level}) :- \]
\[ f_{\text{structure}}(\text{Beta},F,N), F= \text{subj}/_, \]
\[ \text{node}(\text{N1})::F1::\text{node}(\text{N}), F1 \neq \text{subj}/_, \]
\[ f_{\text{c}}(\text{N1},F1,\text{Alpha},\text{Alpha}_Funct,0,\text{Level}_x), \]
\[ \text{Level is Level}_x + 1. \]

\[ f_{\text{command}}(\text{Alpha},\text{Alpha}_Funct,\text{Beta},\text{Level}) :- \]
\[ f_{\text{structure}}(\text{Beta},F,N), \]
\[ F \neq \text{subj}/_, \]
\[ f_{\text{c}}(\text{N},F,\text{Alpha},\text{Alpha}_Funct,0,\text{Level}). \]

\[ f_{\text{c}}(\text{N},F,\text{Alpha},\text{Alpha}_Funct,0,0) :- \]
\[ \text{node}(\text{N})::\text{Alpha}_Funct::\text{index}::\text{Alpha}, \]
\[ \text{Alpha}_Funct \neq F. \]

\[ f_{\text{c}}(\text{N},F,\text{Alpha},\text{Alpha}_Funct,\text{Lev},\text{Lev}) :- \]
\[ \text{Lev} > 0, \]
\[ \text{node}(\text{N})::\text{Alpha}_Funct::\text{index}::\text{Alpha}. \]

\[ f_{\text{c}}(\text{N},F,\text{Alpha},\text{Alpha}_Funct,\text{Lev},\text{Level}) :- \]
\[ \text{node}(\text{N1})::F1::\text{node}(\text{N}), \]
\[ \text{Lev} \text{ is Lev} + 1, \]
\[ f_{\text{c}}(\text{N1},F1,\text{Alpha},\text{Alpha}_Funct,\text{Lev1},\text{Level}). \]

\[ f_{\text{domain}}(\text{Node},\text{Dominated}) :- \]
\[ \text{same_level}(\text{Node},\text{List_same}), \]
\[ \text{delete}(\text{Node},\text{List_same},\text{List_other}), \]
\[ \text{maplist}(\text{depth},\text{List_other},\text{D}), \]
\[ \text{flatten}(\text{D},\text{DF}), \]
append(List_other,DF,Dominated).

same_level(Ind,List_other) :-
    node(Node)::index::Ind,
    node(X)::Y::node(Node),
    findall(N/A,(node(X)::A::node(N),A \= Y),List_same_level),
    delete(Node/Y,List_same_level,List_other).

depth(Node,List_domain) :-
    bagof(Sub_node,sub_node(Node,Sub_node),List_domain).

sub_node(Node,Sub_node) :-
    node(Node)::X::node(Sub_node),
    node(Sub_node)::index::_.

sub_node(Node,Sub_node) :-
    node(Node)::path(X)::node(Sub_node),
    node(Sub_node)::index::_.

The algorithm has two main rules which activate the search,
26a. resolve_anaphor(Net,Index,WeightedList).
    b. resolve_pronoun(Net,Index,WeightedList).

The first rule is used to look for binders of elements which carry the feature [+ana], whereas the second one will care for those elements carrying the feature [+pro]. In this way, we introduce a hierarchy in the general binding mechanism, to the end that even though no given order can be established in the application of the rules, anaphoric elements will be resolved before pronominal ones.

Looking back at the two rules, we see that the two first arguments are input parameters, and the last one is an output parameter. More precisely,

- Net - is the net index, or the input symbol of the graph of the current sentence;
- Index - is the index of the referring expression containing the element to be bound or coreferred, and is the same index used to label the corresponding NP in the graph of the sentence;
- WeightedList - constitutes the output of the algorithm, i.e. a list of the possible antecedents to be assigned as appropriate binders. These two rules proceed to operate with a verification phase, a search phase and a selection phase.

The verification phase extracts from the corresponding file the graph of the current sentence, consults it and then asserts arcs and leaves in memory. When accessing the graph, the node related to the referring expressions to be bound is extracted together with its referential table containing the referential features. At this point the rule verifies whether the feature [+ana] or [+pro] is present.

In the second phase, the search phase, the appropriate rule for binding or coreferring is activated. The program contains a different rule for each lexical elements, be it morphologically expressed or not, and a different set of conditions associated to it. This rule
carries the node index and its referential table as input parameters. The rule called \( \text{refer} \), recovers a list of all possible antecedents available within the sentence.

Three possibilities may arise: there can be no antecedent available, just one and more than one. In the first case, which may only apply to \([+\text{pro}]\) elements, we can either have arbitrary reference assigned - in case the element is also marked \([+\text{ana}]\); or else, external reference, in case the \([+\text{pro}]\) is marked \([-\text{ana}]\). In the second case, the element is picked up as the binder or antecedent. In the third case, the selection phase takes place and an ordering between the possible antecedents is established by means of scores: in other words, the elements of the list are weighted so that the heaviest can be selected as adequate antecedent.

This is how the main algorithm is triggered by the presence of a certain feature in the referential table associated to a certain f-structure node:

27a. \( \text{resolve\_anaphor}(\text{Net}, \text{Index}, \text{WeightedList}) :\)
\[
\text{node}(\text{Node}):\text{index:Index},
\text{node}(\text{Node}):\text{tab\_ref:List},
\text{member}(+\text{ana},\text{List}),
\text{bagof}(\text{Outref},\text{refer(\text{Node,\text{List,\text{Outref}}},\text{List})},\text{Listref}),
\text{maplist}(\text{scoring,\text{Listref,WeightedList}}).
\]

b. \( \text{resolve\_pronoun}(\text{Net}, \text{Index}, \text{WeightedList}) :\)
\[
\text{node}(\text{Node}):\text{index:Index},
\text{node}(\text{Node}):\text{tab\_ref:List},
\text{member}(+\text{pro},\text{List}),
\text{bagof}(\text{Outref},\text{refer(\text{Node,\text{List,\text{Outref}}},\text{List})},\text{Listref}),
\text{maplist}(\text{scoring,\text{Listref,WeightedList}}).
\]

Now, consider how "se stesso" is bound:

28a. \( \text{refer}(\text{Node},[-\text{ref},-\text{pro},+\text{ana},+\text{me}],\text{Ante/N}) :\)
\[
\text{node}(\text{Node}):\text{index:Ind},
\text{f\_command(\text{Ante,F\_ante,Ind,N})},\text{N = 0},
\text{F\_ante = subj/\_},
!.
\]

b. \( \text{refer}(\text{Node},[-\text{ref},-\text{pro},+\text{ana},+\text{me}],\text{Ante/N}) :\)
\[
\text{node}(\text{Node}):\text{index:Ind},
\text{f\_command(\text{Ante,F\_ante,Ind,N})},\text{N = 1}.
\]

Two examples are shown here: the first is a simple case of a possessive anaphor contained in a SUBJect NP of a psychic verb: \( \text{f\_command} \) is used to raise \( \text{proprio} \) out of the SUBJect f-structure and the presence of an OBJect Experiencer triggers binding. In the second example \( \text{proprio} \) is contained in the SUBJect NP of a sentential complement: only the SUBJect of the higher clause is chosen as antecedent; the nuclear NP OBJect is discarded from the list of possible candidates because it is an Unaffected Theme (in case it were an Experiencer it would have been included). Pronominal binding at discourse level is discussed at length in Chapter VIII.
EXAMPLE 1. La salute della propria moglie preoccupa Mario (the health of self's wife worries Mario)
Net index:f2
pred:preoccupare
mood:ind
tense:pres
sem_cat:psych
subj/causer_emot:index:np34
  pred:preoccupare
  sem_cat:state
  spec:df+
  tab_ref:[+ref,-pro,-ana,+class]
subj/poss:index:np35
  pred:moglie:[gen:fem,num:sing,pers:3]
  sem_cat:human
  spec:df+
  ref_tab:[+ref,-pro,-ana,+class]
subj/poss:index:np36
  pred:proprio:[gen:fem,num:sing,pers:3]
  tab_ref:[-ref,+pro,+ana,+me]
obj/experiencer:index:np37
  pred:mario:[gen:mas,num:sing,pers:3]
  sem_cat:human
  spec:df0
  tab_ref:[+ref,-pro,-ana,-class]
PRONOMINALS: np36/subj:[-ref,+pro,+ana,+me]
F-COMMAND: np37/2
Possible antecedent/s of np36: [np37/101]

EXAMPLE 2: lui ritiene che la propria sorella ami Gino (he believes that self's sister loves John)
Net index:f2
pred:ritenere
mood:indic
tense:pres
sem_cat:attitude
subj/agent:index:np4
  pred:lui:[pers:3,gen:mas,num:sing]
  sem_cat:human
  case:[nom]
  spec:df+
  tab_ref:[+ref,+pro,-ana,+me]
fcomp/prop:index:f4
  pred:amare
  mood:subjunct
tense:simple/pres
  sem_cat:state_emot
subj/experiencer:index:np11
  pred:sorella:[pers:3,gen:fem,num:sing]
  sem_cat:human
  spec:df+
  tab_ref:[+ref,-pro,-ana,+class]
subj/poss:tab_ref:[-ref,+pro,+ana,+me]
  index:np12
  pred:proprio:[gen:fem,num:sing]
obj/theme_unaff:index:np13
  pred:gino:[pers:3,gen:mas,num:sing]
  sem_cat:human
  spec:df0
  tab_ref:[+ref,-pro,-ana,-class]
7. More Complex Structures

7.1. Assigning Antecedents to Obviative Pronouns

Obviative pronouns in Italian can be subdivided into three different types: clitics, null Subject pronoun, lexical pronouns. Clitics can be differentiated from lexical pronouns by two basic properties: they are unstressed and they can be bound in the syntax by a TOPic function. In case they are left unbound at c-structure, they can be assigned an antecedent at f-structure. Lexical pronouns are always stressed, and can never be long-distance bound in the syntax. However, they can be used in doubling a local NP, as follows,

29. Il presidente ha promosso un candidato che lui, da semplice commissario, aveva bocciato. / The president passed a candidate which he, as a mere commissioner, had failed.

Lexical pronouns can also be used across sentences or within the text, for contrastive or emphatic aims (see Bresnan & Mchombo(1987) on Chichewa). Finally, the Null Subject is lexically empty and behaves in a manner very close to clitic pronouns: it can be bound in the syntax or be unbound and be assigned an antecedent at f-structure. obviously, it cannot be stressed nor be used for emphatic, contrastive use, nor for doubling. Being lexically empty, it is different from clitics in relation to binding domain: it can be bound from within a complement clause or an adjunct clause by a lexical pronoun, but not by a common or proper Noun. The Null Subject Pronoun originates from discourse: it must be associated to or bound by the current Main Topic of discourse - see Chapter VIII.

30a. pro Ha detto che lui non verrà. / pro said that he will not come.
   b. pro Ha detto che Mario non verrà.
   c. pro Ha parlato di guerra perché lui ama le armi. / He has told about war because he likes weapons.
   d. pro Ha parlato di guerra perché Mario ama le armi.

Only the a.- c. examples allow for coreferentiality between little pro and the lexical pronoun in the COMP - the lexical pronoun being also free to look for an external antecedent in the discourse. The same would happen in case a clitic was introduced in place of the lexical pronoun,

31a. pro Ha parlato di guerra perché Mario lo conosce. / He told about war because Mario knows him.

If we front the adjunct clause, both the lexical pronoun and the clitic are available as antecedents of little pro; and also the common or proper Noun is available, since it f-commands it. However, the lexical pronoun is only available if a list of referents is intended and is not available to continue the discourse topic, being contrastive.

32a. Poiché pro ama le armi, lui ha parlato di guerra.
   b. Poiché pro ama le armi, la polizia lo controlla. / Since pro loves weapons, the police controls him.
   c. Poiché pro ama le armi, Mario ha parlato di guerra.
It is a well-known fact that adjunct clauses can be attached to a lower level, within a complement clause or they can be fronted therein. Here are more examples:

33a. Gino ha detto che Maria verrà all'incontro dopo PRO aver parlato a Tom. / John said that Mary will come to the meeting after having talked to Tom.

   b. Dopo PRO aver parlato a Tom, Gino ha detto che Maria verrà all'incontro. / After having talked to Tom, John said that Mary will come to the meeting.

The difference between a. and b. lies both in semantic interpretation and in the availability of antecedents for big PRO. As to semantic interpretation, the adjunct clause modifies the complement predicate in the a. example, and the matrix predicate in the b. example. As to big PRO, Mary will be the antecedent in a. example and John in the b. example. The skeletal f-structures for the two examples captures the different behaviour of f-command in a straightforward way:

34a. SUBJ/actor: Pred: Gino
     PRED: DIRE <SUBJ/actor, SCOMP/prop>
     SCOMP/prop: Pred: VENIRE <SUBJ/agent>
     SUBJ/agent: Pred: Maria
     ADJ/temporal: sem_mark: Dopo
     SUBJ/actor: PRO
     SCOMP/prop: Pred: PARLARE <SUBJ/actor, OBL/addressee>
     SUBJ/actor: PRO
     OBL/addressee: Pred: Tom

b. ADJ/temporal: sem_mark: Dopo
   SUBJ/actor: PRO
   SCOMP/prop: Pred: PARLARE <SUBJ/actor, OBL/addressee>
   SUBJ/actor: PRO
   OBL/addressee: Pred: Tom
   SUBJ/actor: Pred: Gino
   PRED: DIRE <SUBJ/actor, SCOMP/prop>
   SCOMP/prop: Pred: VENIRE <SUBJ/agent>
   SUBJ/agent: Pred: Maria

In the a. example only Mary can be reached by f-command from the position of big PRO; in the b. example on the contrary, only John can be reached. The same behaviour can be predicted for little pro in tensed clauses. However, note the contrast with corresponding English complex sentences:

35a. John beats her because he hates Mary
   b. Gino la picchia perché egli/pro odia Maria
   c. Gino la picchia perché Maria odia il gatto / John beats her because Mary hates the cat

As usual, we indicate with italics purported coreference between the two items; now, whereas in the English example coreference between her in the matrix and Mary in the subordinate is possible, no such thing may apply to the corresponding Italian version, the b. example – with normal intonation and no emphatic stress. Only the c. example allows it because the NP coreferent with the clitic pronoun is a SUBJECT. Now, why the SUBJECT should be privileged over the OBJECT NP as possible antecedent for pronouns contained in a preposed subordinate clause? This is only explained in a theory of anaphora in discourse, and
in particular by the fact that SUBJects are naturally used as topic of discourse or else some non canonical constituent order must be introduced in the sentence. For instance, in

36a. Dopo che pro è arrivato, Maria ha sgridato Franco / After arrived, Mary scolded Frank

b. Dopo che pro è arrivato, è stato sgridato Franco/ After arrived, has been scolded Frank

c. Dopo che pro è arrivato, Maria lo ha sgridato/ After arrived, Mary has scolded him

Coreference for little pro is only allowed in c.: the passive form in b. with a postposed SUBJect does not permit the NP to be used as coreference, being computed as a FOCus. Being a FOCus requires a new topic of discourse to be set up and the previous references to be discarded. This is clearly shown by the specular structure in,

37a. Dopo che è arrivato Gino, pro si è seduto. / After has arrived John, self sat down.

b. Dopo che Gino è arrivato, pro si è seduto. / After John has arrived, self sat down.

c. Dopo che pro è arrivato, Gino si è seduto. / After pro has arrived, John sat down.

Where coreference in a. between Gino and pro is blocked because Gino is a focussed constituent and ARRIVARE has a lexical form with a focussed OBJect at lexical level (see Bresnan and Kanerva). When the OBJect/Theme is used as a SUBJect/Theme, however, coreference between the proper noun and the pro is possible, as shown by b.; the same applies to pro in the preposed adjunct clause and the proper noun as SUBJect of the main clause.

In order to cope with these facts, the algorithm must compute Obviation and from the obviative clausal structure see whether it can access another clausal structure at the same level or at a level below the one in which it is contained. This is done in our parser by a special procedure called "contains",

38a. contains (index1,index2) :-
    node(node1):index:index1,
    node(node1):path(Bo):index:index2,
    node(node2):index:index2.

b. contains(index1,index2) :-
    node(node1):index:index1,
    node(node1):path(Bo):index2,
    node(node2):index:index2.

Here below we list the program predicate which takes care of little pros and possible antecedents contained in another clause:

39a. refer(Net,Ind,[+ref,+pro,-ana,-me],Ante/N):-
    node(node):index:Ind,
    node(node):cat:features,
    node(node):num:number,
    find_gender(node,Gen),
    f_command(NAnte,F_ante,Ind,N),N > 0,
    f_structure(NAnte,F_ante,N_ante),
    not contains(NAnte,Ind),
    node(N_ante):F_sup:node(N2),
    node(N2):F/R:index:Ante,
8. Chains and Binding

At e-structure level, when a syntactic variable is bound to a TOPic or a FOCus, a chain is created, which essentially is a couple of f-structures carrying the same index. One of the two members of the chain - the foot, is the controlled or bound element: this is an argument function and carries a semantic-role; on the contrary, the head of the chain, the controller or binder is a non-argument function and has no semantic-role – no lexically and grammatically assigned theta-role in Chomsky’s theory. At f-structure level, the chain counts as a single element, in other words, the head of the chain plays no independent referential role from its foot, which is the argument function. Thus a short anaphor can be bound by the foot of a syntactic chain if contained in the same clause. On the contrary the head of the chain, which is contained in the higher domain cannot be the antecedent of anaphors or pronouns. The head of the chain, in turn, can contain a referring expression, a quantified expression, a pronoun or an anaphor: in the latter case, the foot cannot act as an antecedent, being coindexed with an element which must be itself bound in some domain. The domain is the one of the foot to which the anaphor contained in the head of the chain must be bound. We shall discuss some examples, now:


b. A se stesso Franco crede che Tom non pensi e mai. / Himself Frank believes that Tom never thinks to.

c. Parlando di se stesso, Nixon ha detto a Bush che ama la propria famiglia. / Talking about himself, Nixon told Bush that he loves his own family.

Consider a. and the status of suo/his: it is contained in an OBLique/subj_disc(subject of discourse) and as such it can either be bound to the local SUBJect, big PRO, which in turn being contained in an untensed adjunct is bound under f-command by the SUBJect of the matrix, or be free and be bound to the coargument of the matrix SUBJect, the OBJ2 "Bush". Now consider lo/him which is contained within the non-restrictive relative clause: being a pronoun, it is obviative within its minimal clause and must look in the higher f-structure, the matrix clause. At this level, two possible antecedents seem to be available: Nixon and Bush. However, Bush is already bound to the relative pronoun which is the SUBJect of the relative
clause that contains the pronoun lo. Thus, it must be eliminated from the list of the possible candidates. In example b. a short anaphor se stesso/himself has been "left dislocated" and is thus bound to its bindee in the embedded clause: since the anaphor requires a binder, and the interpretation of the anaphor is derived from the location of its bindee, the antecedent of the anaphor should be found in its minimal clause. Tom is thus the binder of the anaphor and not Frank.

Finally, in the c. example, the anaphor contained in the adjunct clause is bound only to big PRO and this in turn is anaphorically controlled by the SUBJect of the matrix, Nixon. Differently from the pronoun in the a. example, the anaphor cannot pick Bush as its possible antecedent. Now consider proprìa/his own: the reportive verb of the matrix dire/say requires the matrix SUBJect to bind the lower little pro and thus to act as antecedent for the possessive anaphor.

The main predicate which spots chain members contained in a separate f-structure from the one containing the variable and the reflexive or pronominal element is non_referred_in, which we list here below:

\[
\text{non\_referred\_in(index,Ante)}:
\begin{align*}
\text{pair\_level(index,LPair),} \\
\text{maplist(find\_ind,LPair,ListInd),} \\
\text{not referenced(Ante,\text{[]},ListInd).}
\end{align*}
\]

\[
\text{referenced(Npx,Path,LPair)}:
\begin{align*}
\text{((antecedent(\_,Npx,Np1);antecedent(\_,Np1,Npx))} \\
\text{;}
\text{(controlled(Npx,Np1);controlled(Np1,Npx))},}
\text{not member(Np1,Path),}
\text{referenced(Np1,[Npx|Path],LPair).}
\end{align*}
\]

\[
\text{referenced(N,Path,LPair)}:
\begin{align*}
\text{member(N,LPair), !.}
\text{find\_ind(node/\_,Ind):- node(node):index:Ind, !.}
\text{find\_ind(node/\_nil).}
\end{align*}
\]

This predicate deletes from the list of possible antecedents for lexical pronouns the NP head of the chain, and takes as local binder of a reflexive the controlled variable or foot of a chain.

Let's consider now more closely some residual problems with some English examples taken from Barrs(1988). First the English version of an Italian example, where we indicate the syntactic index, then after the slash the anaphoric index,

44. Himself\_j/k, Frank believes that Tom\_k never thinks to \_ej.

which has a lexical anaphor himself that can be bound both by Frank and by Tom. This is not allowed in Italian: in other words, Italian requires the anaphor to be "reconstructed" back into the place from which it has been extracted to produce the Topicalized structure. This is possible by considering the variable as the foot of a chain and the topicalized element as its head. Barrs's examples are very similar (his 7a, 42)

45a. Which pictures of himself did John say Bob liked e ?
b. Himself, he thinks Mary loves e.

In 45.a, the sentence is ambiguous - either John or Bob may be interpreted as the antecedent of the reflexive, in the b. example binding by he is grammatical, however in the corresponding Italian examples, no such ambiguity may arise and the b. version becomes ungrammatical.

46.a. Quali foto di se stesso Gino ha detto che Bruno ama e?
b. *Se stesso, egli pensa che Maria ama e.

Ungrammaticality is readily explained by the fact that se stesso must be locally bound and Maria is not an adequate SUBJect binder because of failure of agreement features. Two cases suspend ambiguity: the anaphor is contained in a predicative function, an ACOMP, or there is an accessible SUBJect, and are illustrated by the following examples, (his 7b, 17)

47.a. Whose pictures of himself did John say Bob liked e?
b. How proud of himself did John say Bob became e?

In 47.a. the possessive pronoun whose provides a POSSessor or a SUBJect for the binding of the anaphor in its minimal local domain; in b. the head predicate "proud" is a predicative function with a functionally controlled SUBJect which is lexically bound to the available SUBJect "Bob". This happens before f-structure is accessed, so that no more binding domains may be accessed. Barrs gives a version within Chomsky's (1986) "Barriers" framework and Higginbotham's (1983) Linking Theory which accounts for the same facts in a transformation model.

9. Computing Ambiguous Antecedents

In this section we shall discuss the procedure used by the binding module whenever an independent subject pronoun is contained within a subordinate or coordinate clause and the matrix clause contains two possible candidates as antecedents. As a general strategy, we also take plural independent subject pronouns to constitute a special case if related to other independent pronouns: the reason is the nature of plural reference, which can be assigned either to a single plural antecedent or to a set, a collective nominal, a single plural referent or a split antecedent constituted by a number of n singular and/or plural single referents.

Going back to singular subject pronouns, we take ambiguity to be simply a case of obviation with the SUBJect of the main clause: in other words, the default rule would apply any time obviation would fail and coreference would be forced with the SUBJect of the main clause. A number of procedures are activated in order to check whether grammatical, syntactic and semantic features may be used in order to produce obviation with the Subject of the higher clause.

Going back to singular ambiguous pronouns, at first, agreement of functional features is checked in order to ascertain whether the two or more possible candidates match the features of the pronoun. In case more than one candidate is filtered as possible antecedent, the algorithm performs contextual reasoning on the basis of semantic and aspectual features – more on this topic in Book 2. in a chapter dedicated to Causality.

The basic reasoning is in line with the findings published in a paper by Grober, Beardsley and Caramazza(1978), where however, there is no explicit procedure to explain the facts. Here are some of their examples:
50a. John telephoned Bill because he wanted some information
  b. John criticized Bill because he misplaced the file
  c. The director criticized the actor because he forgot his lines
  d. The actor was criticized by the director because he forgot his lines
  e. John sold the bike to Bill because he needed the money
  f. John sold the bike to Bill because he could pay cash
  g. The prisoner shot the warden because he knew there would be no amnesty
  h. The warden shot the prisoner because he was trying to escape
  i. The prisoner was shot by the warden because he was trying to escape

In all these examples we have an ambiguous singular independent pronoun contained in a *because* subordinate clause; but what actually matters, is the semantics of the verbal predicates and the semantic roles of the antecedent and the pronoun. Positively marked verbal predicate induce different behaviour in the reasoning performed inter sentimentally: *criticize, misplace, forget, need, shoot, escape, scold, dislike, apologize, disown, etc.* all point to bad opinion evaluation of the Affected Theme. Also consider *because* clauses when compared to *but*-subordinate clauses,

51a. Mary may scold Nancy because she dislikes people who crack their gum
  b. John must scold Bill because he disobeys orders continuously
  c. Mary must apologize to Nancy but she doesn't have to be sincere
  d. John may praise Bill but he doesn't have to like him

Here we see that the antecedent is picked out simply on a default Subject oriented rule. The procedure we follow is this one: the main predicate is accessed and its semantic category is checked: it must belong to a restricted set including "evaluative", "communicative", "extensional"; also the semantic marker of the subordinate must be a causal connective like "because", "since". Relevant aspectual categories include "achievement", "accomplishment". The same procedure is performed on the subordinate main verbal predicate. Thematic or semantic roles of the candidates for antecedents are checked and compared with the one of the pronoun if needed. Also temporal interpretation is taken into account. We show here below some examples with a full listing of both f-structure and binding module. However, more basic examples where ambiguity can be solved on a linguistic basis, i.e. without resorting to knowledge of the world, include at least the following:

52a. The teacher passed the student because he was prepared
  b. The teacher passed the student because he was tired

where we have an "achievement" with a verbal predicate classified as "social_role" and a RD which is "specific" or strongly referential, in the main clause, where the subordinate is a state with respectively: an evaluative adjectival predicate in 52a. and a state adjectival predicate in b. The reasoning performed by the algorithm is simple: an agent of a "social_role" may be in a given state and may evaluate but not be evaluated.

On the contrary, in the following examples, the exceptional predicate in the subordinate clause triggers the appropriate reasoning. The Italian version is different from the English one in that little pro does not carry information relatively to gender, and the clitic pronoun *gli* is not marked as to animacy vs. humanhood.

53a. Il contadino picchia il suo asino perché non gli ubbidisce/The farmer beats his donkey because pro does not obey him
  b. Il contadino picchia il suo asino perché ama spronarlo / The farmer beats his donkey because pro loves to spur him
Nothing special is invoked for example b. However with a., the verb obey is the trigger of the exceptional candidate choice, or the trigger of the obviation mechanism. The verb obey belongs to a very limited set of verbs which like psych verbs have a "theme" semantic role for SUBJect. In particular, "obey" has an "agent" as OBJ2 and a "theme_affect" as SUBJ and "social_role" as semantic category. Other such verbs are "grudge, envy, rebuke, etc.". Here are some complete f-structure,

[mario criticava luigi perché ha rovinato il file]/Mario criticized Luigi because he has damaged the file
index:f4
pred:criticare
lex_form:[np/subj/actor/[human], np/obj/theme_aff/[human]]
mood:ind
tense:imp
sem_cat:evaluative
subj/actor:index:sn6
cat:[human]
pred:mario:[gen:mas,num:sing,pers:3]
spec:def:'0'
tab_ref: [+ ref, - pro, - ana, - class]
obj/theme_aff:index:sn10
cat:[human]
pred:luigi:[gen:mas,num:sing,pers:3]
spec:def:'0'
tab_ref: [+ ref, - pro, - ana, - class]
adj:adj:sem_mark:perché
sub/prop:index:f8
pred:rovinare
lex_form:[np/subj/agent/[human], np/obj/theme_aff/[object]]
mood:ind
tense:pass_pross
cat:extensional
subj/agent:index:sn20
cat:[human]
pred:pro:[gen:_num:sing,pers:3,case:nom]
spec:def:
tab_ref: [+ ref, + pro, - ana, - me]
obj/theme_aff:index:sn28
cat:[object]
pred:file:[gen:mas,num:sing,pers:3]
spec:def:+
tab_ref: [+ ref, - pro, - ana, + class]

TO RESOLVE: sn20
THIS IS THE LIST OF PRONOMINALS: [sn20/subj/[ + ref, + pro, - ana, - me]]
I ORDERED THE LIST OF PRONOMINALS: [sn20/subj/[ + ref, + pro, - ana, - me]]
AMBIGUOUS ANTECEDENTS: [sn6/n8/subj/agent, sn10/n17/obj/theme_aff]
Possible Antecedents of sn20: [sn10/107]

[mario telefonò a luigi perché voleva delle informazioni]/Mario called Luigi because he wanted some
piece of information
index:f4
pred:telefonare
lex_form:[np/subj/actor/[human, animate], pp/obl1/address/a/[human]]
mood:ind
tense:pass_rem
sem_cat:communicative
subj/actor:index:sn6
Consider now two well-known Winograd's examples commented in Sidner (289-290):

54a. The city council refused to give the women a permit, because they advocated revolution

b. The city council refused to give the women a permit, because they feared violence

In her comment, Sidner builds up a very complex model of the reasoning underlying the inference chain from "refuse to give permit" to "fear violence". In case of example a., she assumes then that a contradiction must be reached somewhere in order to alter the decision of the focussing algorithm, which would otherwise choose the "city council" as antecedent or cospecifier for "they".

In our model, all the reasoning is done by means of semantic roles and semantic classes of predicates, which may be regarded as a decomposition of the meaning of predicates and their arguments in a specific context. Then the system uses a small number of semantic rules one of which is a default rule. The first thing to notice is that only one example needs obviation, i.e. example a.; on the contrary example b. is computed on the basis of default rules. In our system, a verb like "advocate" has independent semantic roles to assign, which coincide with the selectional restrictions of "city council". However, "advocating revolution", has a special idiomatic meaning which assigns an opposite set of restrictions to the SUBJec
Anaphoric Binding

This is sufficient to prevent a successful match with "city council", which bears the same selectional restrictions of the pronominal, with positive value, though.

9.1. Possessives and Ambiguity

In a language like Italian singular third person possessive pronouns are inherently ambiguous in that they only have one lexical form and agree with the possessed object, rather than with the possessor. However, from the analysis point of view, this corresponds to a situation in which we have a possessive pronoun and two possible antecedents in Germanic languages. So we can simply discuss this case which is commented by the following examples:

55. John told Mario that he sold his car yesterday

From what we said in the previous section, we are now able to compute the antecedent of the pronoun "he" in the complement clause of the verb "tell", as being the NP subject "John" of the matrix clause. What about the antecedent for the possessive pronoun "his"? This might be bound within its f-domain or look for an antecedent in the upper f-domain, or still be left free in order to look for an antecedent in the discourse domain, where a Main Topic should be chosen.

We may dub the first possibility as grammatical binding, the second possibility as semantic binding and the third as discourse binding. Now, grammatical binding should be taken as the default strategy, thus meaning that the other two should be activated first. In particular, the system applies discourse binding by recovering the referring expression which has been computed as Main, Expected or Secondary Topic in the previous portion of text, and then tries semantic binding with any of these, in case they are different from the already available individuals contained in the current utterance. Monitoring salient entities is not as expensive as an exhaustive search in the model for an individual with a distinguishing property, that of being John's or Mario's car.

In case there is no salient individual computed as Topic in the previous text span, semantic binding will be tried: the model is queried in order to ascertain whether there is a possession relation between any one of the agents explicitly mentioned in the context, John or Mario, and a car. In case there is one such relation, the pronoun is consequently coindexed with the actual possessor's index. This is the representation of facts that may be present in the model, that we may query,

\[
\begin{align*}
\text{fact}(\text{Infon}, \text{name}, [\text{john}, \text{Id1}], \text{Pol}, \text{Tloc}, \text{Sloc}) \\
\text{fact}(_, \text{poss}, [\text{john}, \text{Id1}, \text{Id2}], \text{Pol}, \text{Tloc}, \text{Sloc}) \\
\text{fact}(_, \text{is_a}, [\text{ind:Id2, class:car}], \text{Pol}, \text{Tloc}, \text{Sloc})
\end{align*}
\]

In case also this fails, we are left with grammatical binding. Here again we might assume that the search for the antecedent should be extended to discourse entities: however, the economy principle we discussed above establishes as a disambiguating criterion that in presence of possible local antecedents these should be considered first as preferential binding strategy. Thus, we are left with linguist cues like grammatical functions and semantic roles, as well as semantic classes of verbal predicates. We take the presence of a role like an
Addressee not to be the most adequate antecedent for a possessive of a Subject_of_Discourse argument, as in

56a. John talked to Mario about his journey

In other words we apply as preferential reading a disambiguating strategy based on semantic roles which gives us as first choice the Actor John as antecedent for "his". The same strategy can be used in the following pair of examples,

56b. John told Mario that he wanted to buy his car
c. John told Mario that he wanted to sell his car

where the local Subject variable has in case b. the semantic role of Agent, for the buying event, whereas in example c. it has the role of Owner, for the selling event. The pronominal chain established by functional control first, which binds the local Subject variable to the higher Subject "he" which in turn gets bound by John in the matrix clause, by anaphoric control, allows us to make the following deductions: John can be the Agent of the buying event and become the Owner of Mario's car, but surely not the Owner of John's (his own) car. In case c. we also can make up some inferential process on the basis of semantic roles: John can be the Agent of the selling event and be the Owner of John's (his own) Possession. In both cases the most favoured reading is the one which comes out as a default rule, since the most readily available antecedent is the local Subject. However there might be a case in which John is the agent of a car sale agency and has the social role of car seller (and buyer); in that case both binding decisions should be reverted.

More problems are addressed in the following chapter, where the presence of quantifiers is adequately taken into account.

10. PLURAL PRONOUNS

Plural pronouns are problematic essentially for two reasons: their antecedent may be a split antecedent, i.e. it is constituted by a number of NPs which are scattered in the f-structure even though they bear some relation with one another; their antecedent may be an indefinite NP which is computed as being in the scope of an universal quantifier with distributive reading - this is dealt with in the next chapter. Plural pronouns are problematic also at discourse level mainly for the same reasons and for others which we will be tackled in the following chapter.

10.1. Split Antecedents

Our system can cope with two basic configurations:

a. the NPs constituting the antecedents of the plural pronoun are coarguments of the same predicate as in the following example:

57a. John told Mary that they shall have to leave tomorrow.
b. John sent a letter to Mary because they had quarrelled before parting.

In 58 the NPs constituting the antecedents of the plural pronoun are in two separate clause and bear the same grammatical function:

58. While Mary was playing the piano, John was singing and together they were having a lot of fun.
Whenever a plural pronoun is encountered by the binding module, the f-structure net must be searched for exhaustively in order to find all possible f-commanding NPs. In case there is a plural NP which is evaluated as possible antecedent, then nothing else is done. However, in case no such NP exists, the algorithm looks for coargument NPs first. In case some are found, they are checked for semantic and functional matching.

11. Short Binding and Grammar Architecture - A Problem for Parsers

It is usually the case that anaphoric and pronominal binding take place after the structure building phase has been successfully completed. In this sense, c-structure and f-structure in the LFG framework - or s-structure in the chomskian one - are a prerequisite for the carrying out of binding processes. In addition, they only interact in a feeding relation since binding would not be possibly activated without a complete structure to search, and there is no possible reversal of interaction, from Binding back into s/c-structure level seen that they belong to two separate Modules of the Grammar. As such they contribute to each separate level of representation with separate rules, principles and constraints which need to be satisfied within each Module in order for the structure to be licensed for the following one.

11.1. Positive and Negative Constraints

Anaphoric and Pronominal Binding are usually treated as if they were one single grammatical phenomenon, even though the properties of the linguistic elements involved are quite different, as the subdivision of Binding Principles clearly shows. However, it is a fact, that the grammatical nature of a pronoun - be it an anaphor (short or long one), or a free pronoun - is never taken into account when searching for the antecedent. The anaphoric module of the grammar takes for granted the fact that both the structure associated to the anaphor/pronoun, the grammatical function - at f-structure level in LFG - and the functional features are all consistent, coherent and respondent to the Grammaticality constraints stipulated in each grammatical theory. It is the structural level that guarantees consistency, not the Anaphoric/Pronominal Binding Module, which has the only task to add antecedent-pronoun/anaphor indices in the structure, to be used by the semantic modules.

In fact, pronominals obey to two different sets of constraints: positive ones, those belonging to the anaphoric kind, and negative ones, for the free real pronominals. Positive constraints are usually activated within a local domain, where the same will not apply for free pronominals. In this paper we will be only interested in reflexive pronouns, usually called short anaphors, in OBJect position. We also restrict our interest to anaphors contained in relative clauses, i.e. open adjuncts which can be attached at different positions or levels. We thus eliminate anaphors in SUBJect position of sentential complement clauses and long anaphors ("proprio" and other reflexive pronouns).

We chose a couple of examples which represent the theoretical query to be solved, given a certain architecture of linguistic theories, which may differ in the way in which they reach a surface representation into syntactic constituents of the input string, but all converge into the
need to keep the anaphoric module separate from the structure building process. The examples are in English but may be easily replicated in other languages:

59a. The doctor called in the son of the pretty nurse who hurt herself
   b. The doctor called in the son of the pretty nurse who hurt himself

In the second example we have the extraposition of the relative clause, a phenomenon very common in English but also in Italian and other languages. The related structures theoretically produced, could be the following ones:

a1. s[np[The doctor],
   ibar[called in],
   vp[np[the son, pp[of, np[the pretty nurse, cp[who, s[pro, ibar[hurt],
   vp[sn[herself]][[[]]]]]]]]]]]

b1. s[np[The doctor],
   ibar[called in],
   vp[np[the son, pp[of, np[the pretty nurse]], cp[who, s[pro, ibar[hurt],
   vp[sn[himself]][[[]]]]]]]]]

If this is the correct input to the Binding Module, it is not the case that b1. will be generated by a parser of English without special provisions. The structure produced in both cases will be a1. seen that it is perfectly grammatical, at least before the binding module is applied to the structure and agreement takes place locally, as required by the nature of the short anaphor. It is only at that moment that a failure in the Binding Module warns the parser that something wrong has happened in the previous structure building process. However, the only output available is the one represented by a1., which wrongly attaches the relative clause to the closest NP linearly adjacent to the relative pronoun.

The reason why the structure is passed to the Binding Module with the wrong attachment is now clear: there is no grammatical constraint that prevents the attachment to take place. The arguments of the governing predicate HURT are correctly expressed and are both coherent and consistent with the information carried out by the lexical form. At the same time the Syntactic Binding has taken place again correctly by allowing the empty "pro" in SUBJect position of the relative adjunct to be "syntactically controlled" by the relative pronoun, which is the TOPic binder, in turn syntactically controlled by the governing head noun, the NURSE. There is no violation of agreement, nor of lexical information, nor any other constraint that can be made to apply at this level of analysis in order to tell the parser that a new structure has to be produced.

11.2. Short Anaphora

On a first run of our parser with these examples, it has shown the effect of "garden path", in that it has gone into a loop with the unwanted result of "freezing" the computer, due to data
overflow. In other words, as soon as the Binding Module tries to process the f-structure received as input, seen that short anaphora requires binding to take place within a local domain, f-command – as explained in previous sections - will impose the same level of containment for both the pronoun and the antecedent. And seen that the only antecedent available is the empty SUBJect which has functional features inherited by means of syntactic control from the governing relative pronoun, the agreement match is attempted, and a failure ensues systematically.

As a result of a failure at the Binding Level, a call to the structural level is issued which attempts to build the structure another time. But seen that no failure has taken place at this level of analysis, the result will be the same as the previous one. And this process will go on indefinitely, seen that the two modules obey different Principles and satisfy them separately.

To overcome this predicament we had to come up with a new theoretical proposal that we will now put forward, regarding exclusively short anaphors, thus disregarding long anaphors and reciprocals in particular or "proprio" in Italian, which call for a different treatment. The proposal we will make is very simple:

60. "short anaphora must be checked for agreement with their available binder already at the level of satisfaction of grammatical principles, before the structure is licensed"

This requirement is not introduced by the need to improve on the implementation side of the parser, but responds to theoretical principles inherent in the formulation of the Binding Principles. Short anaphora not only obey positive constraints, as opposed to the other pronominals, they also carry a locality requirement which is equivalent to the same domain in which Grammaticality Principles apply, such as the ones expressed in LFG - Uniqueness, Completeness, Consistency. At each "propositional" level, corresponding to a simple f-structure and roughly to a Complete Functional Complex in GB terms (see Chomksy 1986, 169), all arguments of the governing predicate must be checked for completeness - they must all be present at functional level, even if they may be lexically empty; they must be coherent, i.e. only those included in the corresponding lexical form must be present; each functional attribute must be assigned to a unique functional value. And in our case no violation is detectable seen that the attributes belonging to the empty "pro" SUBJect are unique even though they are not appropriate to bind the short anaphor OBJect of the same predicate HURT. However there is no indication in the grammar that they should be checked for agreement at that level of analysis.

By anticipating the working of the Binding Module, we assume that Short Anaphors belong partly to the Grammar level and partly to the Binding level: they belong to the grammar level since they require and can to be licensed at sentence or propositional level without their f-features being in agreement with their antecedent and binder. Besides, they belong to the binding level where agreement takes place and coindexation follows, in case of success.

As to cases in which the anaphor is contained within a NP in SUBJect position of a sentential complement, the search for the antecedent is suspended not being available locally and no agreement match can be performed. This will not apply to anaphors contained within the NP of the OBJect seen that the antecedent is available.
A failure in the Anaphoric Module will simply cause the Parser to backtrack but the structure produced will not change seen that the failure has taken place in a separate module. Of course, the alternative is using a single unification mechanism that takes context-free rules with all possible alternatives, builds a tentative structure than unifies functional features, and in case of failure tries another possible structure. However this perspective is not only computationally inefficient, it is basically psychologically unfeasible: there will be no principled reason to tell Garden Path sentences apart from the rest seen that all sentences can be adjusted within the parser, sooner or later. Also processing time is not controllable seen that the parser will produce all possible structures anyway and there is no way to control the unification mechanism in a principled manner. On the contrary, in a parser like ours, the order of the rules is controlled strictly, and also the way to produce backtracking is controlled, seen that the parser has both a lookahead mechanism that tells the parser which rule to access or not at a given choice point.

In this way we split Bound Anaphors and make them obey the same principles of Sentence Grammar to which they belong in all respect. The structures produced by the parser take only different processing time to allow for backtracking to take place within the main parser body: but then the right attachment is achieved and the complete structure is produced with the right binding, as shown below.

59a.index:f3
   pred:call_in
     lex_form:[np/subj/[human],np/obj/them_nonaff/[human],idioms/form/[in]]
     voice:active
     mood:ind
     tense:past
     cat:activity
   subj/actor:index:sn7
     cat:[human,social]
       pred:doctor
       gen:mas
       num:sing
       pers:3
       spec:def:+
       tab_ref:[+ref,-pro,-ana,+class]
   obj/them_nonaff:index:sn14
     cat:[human,animate]
       pred:son
       gen:mas
       num:sing
       pers:3
       spec:def:+
       mods:adj/specif:sem_mark:of
          index:sn21
          cat:[human,social]
          pred:nurse
          gen:fem
          num:sing
          pers:3
          spec:def:+
          mods:relative:topic:topic_type:relative
             index:sn22
             cat:[human,social]
             pred:who
Anaphoric Binding

pers:3
gen:fem
num:sing
case:nom
controller:sn21
tab_ref: [+ref,-pro,-ana,-me]
index:f2
pred:hurt
lex_form:[np/subj/experiencer/[human/animate],
np/subj/them_nonaff/[object,human,abstract]]
voice:active
mood:ind
tense:past
cat:perceptive
subj/experiencer:index:sn23
cat:[human]
pred:pro
pers:3
gen:fem
num:sing
case:nom
spec:def:+
controller:sn22
tab_ref: [+ref,+pro,-ana,-me]
obj/them_nonaff:index:sn24
cat:[human]
pred:herself
pers:3
gen:fem
num:sing
case:[acc,obl]
spec:def:+
tab_ref: [+ref,+pro,+ana,+me,-subj]
antecedent:sn23
interpretation: definite
aspect:activity
mod_a:index:sa3
cat:[subjective]
pred:pretty
tab_ref: [+ref,+pro,-ana,+class]
tab_ref: [+ref,+pro,-ana,+class]
form:in
aspect:activity
rel1:[tr(f3_amb12) < td(f3_amb12)]
rel2:[tr(f3_amb12) = tes(f3_amb12)]
definiteness:+
ref_int:[tr(f3_amb12)]

59b.index:f3
pred:call_in
lex_form:[np/subj/[human],np/obj/them_nonaff/[human],idioms/form/[in]]
voice:active
mood:ind
tense:past
cat:activity
subj/actor:index:sn1
cat:[human,social]
pred:doctor
12. CURRENT STATUS AND COMPARISON WITH RELATED WORK

In this section, we shall briefly address the problem of anaphoric binding by comparing our approach to the one called Functional Uncertainty and proposed by Dalrymple in her Ph.D. thesis (1990). In using f-structures rather than syntactic constituency, LFG makes it more natural and direct looking for information such as being the "subject of", a notion crucial for antecedenthood. Each referring expression receives a separate treatment by the binding rules according to feature matrix, grammatical function, and thematic or semantic role. A set of criteria for assigning priority scores to candidates for antecedenthood and binding is introduced in order to define what can be bound by what: candidates receive scores according to their grammatical function, SUBject scoring the highest; and to thematic role, agent scoring the highest, and so on. Exceptions are also individuated on the basis of the interplay of grammatical functions and thematic roles: for instance one such rule says that a possessive anaphor contained in a subject f-structure can be bound to a NP in its sentence unless it is a Theme. As appears, binding is crucially performed on a structural basis, rather than on a functional basis as the approach based on Functional Uncertainty - as Dalrymple proposes - would require. The structures involved are f-structures: the parser makes reference to the SUBJect, a primitive notion which is used primarily to set NP f-structure apart from clausal ones; untensed clauses may either appear as controlled complements, or as closed adjuncts or closed functions such as SUBJect: also in this case anaphoric binding applies as long as structural conditions allow it. In this sense, anaphoric binding together with syntactic binding are structurally determined and can be opposed to lexical binding which is entirely functionally determined. Scores are also very important and are based on the superiority hierarchy of theta-roles, and on the degree of referentiality a certain NP possesses.

In particular, the difference in binding domain existing between an anaphor like "himself" and a pronoun like "him" is obtained simply by reference to the level at which these two lexical items must start out looking for their antecedent: for the former it would be equal to 0, while for the latter the level would be equal to 1. Rather than formulating a "Coargument Disjointness Condition" it is sufficient to individuate a viable f-structure, which looks for the accessible SUBJect in the case of nominal ones and let the feature matrix do the rest.

As we saw, reference to the particular domain in which a certain element must be bound or be disjoint, and reference to the particular grammatical function the antecedent should carry in a particular environment is not sufficient to deal with the inventory of pronominals available in Italian and other languages: reference to semantic role is sometimes required,
whenever a psychic verb is used, as well as the type of quantified NP or quantifier that can become a candidate for antecedenthood in certain environments. The system does this directly by means of the feature matrix associated to the referential table and by directly investigating the content of the functional node, where roles are available together with functional labels. Possibly, the same result could be achieved by means of Functional Uncertainty, even though we have not tried to test this hypothesis.

However, let us consider why Functional Uncertainty has been introduced: basically because syntactic restrictions could be formulated in terms of grammatical functions, and could be expressed by the introduction of equations whose right-hand side member contained regular expressions like the following,

$$\neg \text{TOPIC} = (\neg \text{COMP} \ast \text{OBJ})$$

which refers to the analysis of Topicalization as discussed by Kaplan & Zaenen(1989). The equation specifies an infinite disjunction of paths within f-structures, paths involving zero or more COMPs: OBJ stands for the landing site or for the bindee for the binder. Using functional attributes makes things easier and does completely away with the need to keep in memory c-structure syntactic trees once they have been used to build the corresponding f-structures. I don't intend here to comment on Kaplan & Zaenen proposal, but simply to criticize Dalrymple's idea to use this procedure with some minor modification and adaptation in anaphoric binding.

It is clear to me that the regularity of syntactic phenomena has a different nature from the one belonging to anaphoric ones. An equation like the one reported in 58. states that no matter what happens within the COMP, and as long as the landing site is an OBJ, any number of COMP's may be traversed in order to adequately bind the TOPIC. This never happens with anaphoric binding: even though the difference existing between ADJunct clauses and COMPlement ones is relevant, the depth of embedding is also a crucial factor. Structural differences like the one existing between COMP and ADJ clauses are already taken care for by f-command: however, in order to let, say, a long-distance anaphor or a clitic pierce through, inside-out, more than one relevant domain, a number of conditions on antecedenthood and distance intervening between the anaphor and the antecedent must be also accounted for. Furthermore, it is worth while reminding that FU cannot possibly cope with that-trace effects or any other syntactic phenomena dependent on semantic roles rather than GFs. In our constituent-based analysis outlined in Chapter V, we also tackled the problem of functional features and inherent features to be associated with FOC and TOP elements in SpecCP. In particular, in languages like German, case must be made available to the landing site, and these are not easily accounted for in FU approach. Other problems may ensue if we consider the need to specify negation on the Path, caused by local constraints like the one discussed for the wh- in situ phenomenon in English.
Chapter 7

QUANTIFIERS AND ANAPHORA

1. QUANTIFIER RAISING

This chapter introduces a topic which is very borderline with Semantics and should partly be treated in Book 2. However, since a lot of its import can be related to Anaphora we decided to include it here. In our system, quantifiers are assigned scope at f-structure level, BEFORE entering the anaphoric binding module. As already noted previously, this is not what LFG assumes. In this way, NPs which fall under their scope may have received a quantified interpretation thus allowing for their number to be left unspecified. Consider as an example sentences like,

1a. The women who were carrying a baby each, dropped them.
b. Three women who were carrying a baby each, dropped them.
c. The women who were carrying a piano, dropped it.
d. A woman who wants to marry every man, doesn't have to disappoint them.

In these examples, as in donkey sentences, the antecedent of the pronoun them/it is contained within a restrictive relative clause, and is an indefinite NP a baby which is in the scope of the universal distributive quantifier each or itself a universally quantified NP, every man.

Two things should happen in a./b. in order for the indefinite NP to be taken as antecedent for the plural pronoun: the number of the indefinite NP should be neutralized to allow for anaphoric binding. Besides, whereas in b. cardinality of the babies is inherited from the quantifier which has scope over the indefinite NP, in case a. the cardinality of the indefinite NP should be entirely determined by contextual conditions in the discourse model. The same thing might have happened in case the antecedent were in a separate clause, subordinate or coordinate clause, or even a nonrestrictive relative clause. Remember that the net result of the binding module, is to prevent the pronoun to pop up at discourse level as a free external pronoun. In case the indefinite NP has distributive reading, it will be computed as a set; otherwise, a new individual will be asserted in the world. Finally in d., the quantified NP can bind the reference of the plural pronoun them.

Also definite NPs may be understood as being in the scope of some quantified expression: however, this might either be the result of external knowledge of the world in some case or the presence of a possessive:

2a. Every husband loves his wife
b. Every/each man left the bar at ten
c. Every man wants to buy the car of his dreams

As may be easily noted, both in a. and c. the definite NP may be understood as quantified owing to the fact that the universally quantified NP has scope over it and there is a possessive pronoun bound to it. On the contrary, in b. we might surmise that there is only one bar and the men left it all at the same time. At f-structure level, this fact can only be explained in case we take grammatical functions as carrier of structural information. In all examples, the dominance relation is determined by the presence of a SUBJect quantified NP and an OBJect definite NP. In addition, in a. we have a case of generic assertion; in b. on the contrary, a specific spatiotemporal location anchors the main relation and its arguments; finally, in c. an intensional predicate "want" makes the reading of the main relation "buy" opaque and deoids it of a specific anchoring, as was the case with 1d. above. Other cases of definite NPs with quantified reading in specific spatiotemporal locations may be given by the following example, taken from Story 3,

3. The two little pigs went back each to its own little house./I due porcellini ritornarono ciascuno alla propria casetta.

Given the fact that anaphoric binding should capitalize on the results of the module for quantifier scope assignment, we compute quantifier raising before anaphoric binding and after syntactic binding. This has already been discussed in Delmonte, 1990, where we also stated how we followed Halvorsen's proposal in his paper on semantics(1983). The algorithm for quantifier raising operates on f-structures and proceeds as follows:

- a. quantified NPs are individuated from the content of their Spec: the "quant" attribute should be present and filled with some value;
- b. a q-marker or quantified variable is added at f-structure level;
- c. the quantifier/s is/are raised by a recursive procedure which stops at the first q-barrier: a q-barrier is simply the first closed grammatical function f-commanding the f-structure in which the quantifier is construed. A q-operator is added at this level in a list which has q_ops as functional attribute;
- d. the same applies to indefinite NPs, which are locally disambiguated: not every indefinite NP can be treated as a quantifier;
- e. when we come to definite NPs, these are computed as indefinite quantified NPs in case a number of conditions are met, as specified above;
- f. finally, we might end up with a list of q-operators which has more than one member: at this point we only want to swap their linear order in case the quantifying force of the first q-operator is less than the force of the second, and so on;
- g. in the remaining cases we take the quantifier which is higher at f-structure or it has been assigned the hierarchically higher grammatical function - where SUBJect is the highest - to assume scope of the remaining q-operators. The result is marked at f-structure level with the attribute-value pair "interpretation:quantified".

The same procedure applies to wh- words in case they appear in the same proposition where a quantified NP is present: they behave like indefinite NPs. The well-known subject-object asymmetry applies to both types of elements (see Delmonte, 1990), as shown by the following examples:

4a. Who bought everything for the party?
4b. What did everybody buy for the party?
where the universal quantifier may assume higher scope than the wh-word in case they are both contained within the same proposition and the wh-word is not in a chain with a SUBJect. In this case we simply compute wh-binding and establish a syntactic chain by coindexing: in this way we do not need to go through reconstruction of the wh-element into the extraction place.

Here below we show the output of the parser on a typical example of donkey sentences, where a universal quantifier takes scope over an indefinite NP and arbitrary reading is produced: this can be read out at propositional level, where the attribute perf ormative is present:

8. [every farmer who owns a donkey beats it]
   perf:generic
   index:f6
   pred:beat
   lex_form:[np/subj/agent/[human], np/obj/theme_aff/[human, animate]]
   mood:ind
   tense:pres
   cat:activity
   subj/agent:index:sn2
       cat:[human]
       pred:farmer:[gen:mas,num:sing,pers:3]
       spec:def:'0':[part:-,quant:every]
       interpretation:arbitrary
   mods:mod/predic/topic/topic_type:relative
       index:sn5
       cat:[human]
       pred:who:[gen:mas,num:sing,pers:3,case:nom]
       controller:sn2
       tab_ref:[+ ref, + pro, + ana, - me]
   perf:generic
   index:f5
   pred:own
   lex_form:[np/subj/owner/[human], np/obj/actor/[human,animate,object]]
   mood:ind
   tense:pres
   cat:state
   subj/owner:index:sn24
       cat:[human]
       pred:pro:[gen:mas,num:sing,pers:3,case:nom]
       spec:def:+
       controller:sn5
       tab_ref:[+ ref, + pro, - ana, - me]
   obj/actor:index:sn26
       cat:[animate]
       pred:donkey:[gen:neut,num:sing,pers:3]  
       spec:def:-
       subj/nil:index:sn29
       pred:vbl
       controller:sn24
       tab_ref:[+ ref, - pro, - ana, - me]
       tab_ref:[+ ref, - pro, - ana, + class]
       qmark:q2
       interpretation:quantified
   aspect:state
   rel1:[td(f5_es06)=tr(f5_es06)]
   rel2:[included(tr(f5_es06), tex(f5_es06))]
As said above (but see also Pollack & Pereira, 1988:83-84), at first QR applies adding q-markers q1 and q2, locally for each quantified f-structure; then an operator is accordingly raised at propositional level; finally, quantifier scope is executed and a functional annotation is added where required. In our case, the f-structure where the indefinite NP "a donkey" is described receives the annotation, interpretation:quantified. Then, anaphoric binding is computed and pronouns are coindexed with their possible antecedents: in our case, the pronoun "it" is coindexed with the NP "a donkey" contained in the relative clause. Theoretically speaking, this relation is not an antecedent-pronoun relation – more on this below. F-command is suspended in this case and the relative portion of f-structure is made visible to the binding algorithm by a specific call. Finally, the algorithm for arbitrary interpretation is called and functional annotations are added both at NP level and at propositional level with the feature perf:generic. These annotations are used by the model as well as by the interpretation to compute adequate representations for the situations described by the utterance. In particular, as discussed at length here below, arbitrary readings arise owing to tense and aspect specifications and will produce - in case it is lacking - a quantification over events or states, so that we can understand that a given event/state will hold at any temporal location, assuming a specific anchoring for spatial locations – more on this in a specific chapter in Book 2. In turn, an arbitrary reading for an NP amounts to computing a quantified NP where we are dealing with a "class" rather than with sets of or singleton individuals - in other words, reference in the world is suspended and there is no specific set or group of people we are referring to. Though, since we describe classes as infons, we always intend them to be classes of individuals or entities related to some spatiotemporal coordinate, and not universally valid. In addition, a generic assertion is usually the subjective statement of some individual or other, and is computed from a given perspective.

2. QUANTIFIERS AND ANAPHORIC BINDING

2.1. Quantifiers and f-command
As a first approach to the problem of quantifiers, the algorithm for anaphoric binding takes care of precedence whenever a quantified NP is indicated as possible antecedent for a pronoun. Quantified antecedents are individuated by the presence of the q-marker introduced by the algorithm for QR.

However, when we want to deal with quantifiers and quantified NPs as possible antecedents of little pros, clitics or independent pronouns, a different procedure must be called in, which should prevent the pronoun from taking them as possible antecedents in case they are not f-commanding it.

In this way we can account for lack of coreference between a clitic pronoun contained in a fronted subordinate clause and a quantified NP contained in the main clause, as in the a. example

5a. When I insulted him, every student went out of the room.
   b. When I insulted him, John went out of the room.

as opposed to the b. example, where coreference is allowed as usual.

As already said, quantifiers can bind pronominals in case they precede them. Now, since f-command cannot account for precedence relations at clausal f-structure, we only have to allow the following Italian sentence on the basis of semantic roles and exclude the corresponding English version,

6. La propria saluta preoccupa ognuno.

where however, only a possessive reflexive is allowed. The same criteria can be extended to pronouns contained in a relative adjunct,

7a. I know every boy who loves his mother
   b. *I know every boy whom his mother loves

where the b. example is excluded being a crossover configuration. The corresponding Italian version is allowed as long as proprio, and ciascuno/each are used,

8. Io conosco ciascun ragazzo che ama la propria/*sua madre

We say that a quantified NP head of a relative adjunct is a weak antecedent in Italian and can only bind a reflexive. Generally speaking, a quantifier or quantified NP is a weak antecedent in Italian: it can only bind a possessive reflexive in the same clause and a clitic or a little pro in a separate clause.

9a. Everyone likes his friend.
   b. Ognuno ama il proprio/*suo amico.

2.2. Arbitrary or Generic Readings

Referentiality (or existential quantification) in human language, unlike what may appear in formal logic notation, is not a problem confined to nominals, but rather is predicted from properties of propositions. We shall draw from Talmy(1973) and Barwise(1985) to corroborate this view.

If a situation is presupposed to be true and characterizes an event that took place at some space time location, all the nominals in that sentence are obligatorily interpreted as having some referential identity. In particular, a nominal expression may receive either a referential or nonreferential interpretation according to the environment in which it is placed: in an opaque environment or context, an indefinite NP, or a quantified NP receives a nonreferential interpretation, as in
10a. John wants to read a book

Whereas in a nonopaque environment it is interpreted referentially, i.e. the speaker is committed to referring to a specific object, as in

b. John is reading a book

All else being equal, past and present-progressive tenses commit the speaker to the belief that the act did take place. These two modalities are thus factual, and the nominals associated with sentences in these modalities are therefore presupposed to be referential. Following again Talmy, we consider a factual modality one by which the speaker commits himself to the (past or present) truth of a certain proposition - and also commits himself to the referentiality of the participating nominals: as a result, entities and facts are asserted in the model. A nonfactual modality, on the other hand, is one in which the speaker does not make such a commitment. In other words, the nominal context in question is opaque, and only sits (nonfactual situations) and concepts are asserted in the model. This subject topic has been thoroughly exemplified and commented in Book 2.

As to the HABITUAL modality, rather than having uncertainty involved, generic propositions are not, to begin with, made about specific time-space bound processes, states or events. Therefore, they are not necessarily made about specific (referential) nominals. Rather, they are made about classes or kinds. Also conditional expressions are opaque and opacity is most likely traceable to the FUTURE and the uncertainty modality.

2.3. Generic vs Specific Quantifiers

Generally speaking, all quantifiers can enter into specific or generic readings according to temporal and aspectual interpretation at clause level. This interpretation may in turn be propagated or inherited by other clauses, in case the quantifier be the head of a chain. Consider well-known examples such as:

11a. Everybody loves the films he saw.

b. Everybody thinks he is a nice fellow.

In a. the embedded clause “he saw” can no longer be interpreted as specific in reading (or extensional) because the pronoun “he” is bound to a quantifier and this chain is assigned arbitrary reading due to the temporal aspectual interpretation of the matrix clause. The same would apply to the definite plural NP “the films” which cannot be used to cospecify with some entity existing in the discourse model, nor can it be used to establish a new set. The same would apply to the “he” in b. which at least in one interpretation must be taken to be bound to the quantifier and assigned arbitrary reading.

Another well known set of examples is constituted by indefinite NPs which can be assigned specific or generic reading according to domain restrictions. In particular, two cases can be found:

12a. The indefinite NP is in the scope of a distributive universal quantifier and is itself quantified;

b. The clause with the indefinite NP is governed by an opaque predicate like “want, intend, prefer, expect, etc.”; in both cases no specific reading for the indefinite NP is available, as shown by the following examples

13a. John dates every woman who love a fish.
b. Tom expects to catch a fish and wants to fry it for dinner.

In both cases, the indefinite NP will refer to generic sets or to classes of individuals with a given class restriction, "fish". In the model, both the relations and the participants in the relations are computed as sits, not as facts, thus indicating that there is no extensional reading available for them but only an intensional one. To check this, consider possible continuations:

13c. Everybody loves the films he saw. *They were by Fellini.
d. Tom expects to catch a fish and wants to fry it for dinner. *It was a sole.

Habitual or generic modality characterizes propositions about genera which do not pertain to specific individuals at specific time-place locations, or else pertain to referential individuals (subjects) with respect to which a proposition in the habitual modality holds true. No claim is made about any spatiotemporally anchored event. In both cases, universal quantification on events is involved: in the model, we introduce a quantification on events if lacking, in the form of a quantifier like "always" or "usually" – more on this in Book 2.

See the difference in interpretation ensuing from the presence of specific antecedents in the text,

14A. C'erano una volta tre porcellini che vivevano nella campagna./Once upon a time there were three little pigs who lived in the countryside
14B. I porcellini decisero di costruirsi una casetta ciascuno./The little pigs decided to build a little house each.

In this case, a set of houses is asserted in the discourse, its cardinality being determined by the cardinality of the owner's set, “the little pigs”.

2.4. Quantifiers as Pronominals

In our system, all [+ana] marked pronouns do not possess intrinsic reference, being also marked [-ref] and two consequences ensue: they must be bound in their sentence and cannot look for antecedents in the discourse, unless there are additional conditions intervening, i.e. tense must be specific and not generic, mood must be real. Else, they can be assigned ARBITRARY interpretation, when a controller is lacking, and a series of semantic conditions are met as to tense and mood specification. Since ARBITRARY interpretation is a generic quantification on events this can be produced with untensed propositions or tensed ones, but with no deictic or definite specific import. The notion of binding relevant for quantifiers, is shown by the pair

15a. A woman requires/demands that many/every men be in love with her, *and John knows her.
b. A woman believes that many men like her, and John knows her.

in a., in English as in Italian, the indefinite a woman is computed as generic in the main clause and the same happens to the pronoun her in the complement clause introduced by that.

However, the conjoined sentence contains a presuppositional deontic predicate "know" which requires its complement to be existing in the world, the pronoun "her". The only antecedent available for the pronoun is the NP "a woman": but in this case, the antecedent should be computed as referential and not as generic, so the result is semantic inconsistency. The opposite happens in b., where the indefinite is taken to refer to a specific woman in the discourse, and the two occurrences of her are to be bound to this individual. As shown above, the referential properties of pronouns are tightly linked to the ones of their antecedent. But
also the opposite may obtain, i.e. the referential properties of the antecedents are bound by those of the pronouns, and these in turn are conditioned by the referential nature of the RD-referential domain - in which they are contained: an [-BOUND] domain is one containing indicative mood and reference is free, whereas a [+BOUND] domain is one containing subjunctive mood and reference is not free but locally bound, for anaphors, or lacking in referential import for lexical pronouns.

With plural pronouns, the algorithm checks at first whether there is a single plural antecedent, if not, as a second move, indefinite expressions and quantifiers are searched for. This would apply for such simple cases which dub similar examples used by Webber (1983) as,

16. Ogni uomo in Italia vuole comprarsi la Ferrari, perché sono auto molto di moda / Every man in California wants to buy a Ferrari because they are very fashionable.

We find some of Webber's examples (hers D18-3/4) too contrived to constitute well-formed English cases, which we report here below:

17. D18 - 3 Three men who tried to lift a piano dropped them
   - 4 The three men who tried to lift a piano dropped them

The absence of the quantifier "each" which contributes to the distributed reading, makes these examples quite awkward - to say the least, considering also the fact that lifting a piano in the commonsense world requires more than one man, unless he is a gorilla.

As said above, pronouns may be free, controlled or bound. A pronominal is bound only when its antecedent is a quantifier, a quantified NP or in case no controller is available at sentence level, a number of semantic conditions are met at the level of tense and mood specification and arbitrary reading is assigned to the whole proposition. In the latter case, Italian provides both for empty pronominal expression like a little pro, or a big PRO; as well as for lexical pronouns, like the nominative clitic si roughly corresponding to English "one". Sentences with arbitrary or generic reading can in turn be assigned either universal quantification or existential quantification: only in the former case, when introduced in discourse, they may be iterated without producing incoherence.

In Carlson's approach to bare plurals in English, we see that the a. example receives a quasi-universal reading whereas the b. example receives a quasi-existential reading,

18a. Dogs run around in circles.
   b. Dogs are running around in circles.

The matter is discussed at length in Cinque (1988) who comes to the conclusions that the difference in meaning is due mainly to tense and aspect specification at propositional level. In 19a./b. below we have quasi-existential reading associated to the indefinite/impersonal "si" subject:

19a. Oggi a Beirut si è ucciso un innocente / Today in Beirut one killed an innocent
   b. Oggi a Beirut si è sparato tutta la mattina / Today in Beirut one shot the whole morning
   c. %Oggi a Beirut si è nati senza assistenza medica / Today in Beirut we were born without medical assistance

Example c. is bad because of tense specification and of the nature of the syntactic class of the main verb, an Ergative, as contrasted by the two previous cases where we have a transitive and an unergative verb. The sentence becomes good if we change tense, by introducing present indicative which assigns a non specific time reference to the sentence,

20. Oggi a Beirut si nasce senza assistenza medica
As a matter of fact, simple present may be assigned both a generic interpretation and a specific interpretation in case the spatio-temporal location coincides with discourse time. In this case, the discourse might continue by introducing a single individual satisfying the description denoted by the previous sentence:

21. John's child, for example.

However, this will not alter the meaning already computed in the model by the system, which is a description of a class of individuals - human beings. The previous example simply tells us that "John's child" is to be understood as an instance of that class of individuals, thus inheriting relations and properties: in particular, we now know that "John's child" was born in Beirut without medical care today. In conclusion, si/one are pronouns evoking generic entities which might become specific in the following discourse.

Language-specific conditions are called for Italian little pro: there is a 3rd pers. plural construction which enters arbitrary reading – at least in one interpretation and no discourse preceding it with a suitable antecedent, as long as the pronoun is the empty little pro, and the main relation is spatially located, as shown by his 47:545,

22a. Lì, odiano gli stranieri / There they hate foreigners
   b. Qui lavorano anche di sabato / Here they work even on Saturday

As with other generic readings, also in this case the undetermined subject cannot be bound to a specific individual in discourse. This must be treated as an exceptional case, where agreement is bound to a fixed set of values, as happens with empty expletives. The interpretation of little pro however becomes specific in case the spatial location is omitted, as in "Odiano gli stranieri" or "Lavorano anche di sabato". Arbitrary interpretation may also appear with 2nd person singular pronouns.

3. QUANTIFIERS AND DISCOURSE ANAPHORA

It is a well known fact that quantifiers and quantified NPs do not refer in the text or discourse, in the sense that they are unable to pick up a specific individual as antecedent to which they may corefer. However, when computing reference, quantifiers - either lexically expressed or unexpressed - may be used by speakers to continue the topic of discourse.

Are quantifiers and quantified NPs like other referential expressions? A first answer requires the two to be set apart:

a. Quantified NPs behave like ordinary NPs in that they may be treated as referential or not according to domain restrictions;

b. Quantifiers are different in that they do not possess any explicit set restriction and may only individuate broad classes or general concepts, definable in terms of selectional restrictions of their governing verbal predicate. Domain restrictions are crucial in the determination of what quantifiers may contribute to text or discourse.

We can identify two main cases:

CASE 1. Domain restrictions may trigger two different strategies:
   - A quantifier is computable as a referential entity, which however requires as antecedent a superset of the sets or individuals previously made available in the text or discourse;
   - A quantifier is itself used as such a superset and the following definite NPs may corefer with it.
CASE 2. Domain restrictions may prevent any of the two previous strategies to apply and impose a non referential or arbitrary reading.

3.1. Quantifiers and Quantified NPs as Coreferents

In her works, Webber (1977; 1983) extensively deals with the problem of the interpretation of quantified expressions and their role in binding coreferring pronouns. In order to decide whether a singular indefinite expression can be treated as a plural one, scope must be computed: in particular, its scope must be included in that of a universal quantifier with a distributive reading. We shall quote one of her examples (1983, 363-d25):

23.i Last week Wendy bought each boy a green T-shirt at Macy's.
   ii She prefers them in more subdued colors, but these were on sale.

24. Most people who own a gun
   a. never use it
   b. never use them

These are typical examples which require at first scope assignment to be computed for the two quantified NP's, "each boy" and a "green shirt". As Webber remarks, a definite plural anaphor may also specify a generic set entity and this is possible even with a singular definite noun phrase as antecedent. The only condition seems to be procedural and based on "recency": quoting from Webber, ". . . the listener can generate new generic-set entities whose IDs are based on generalizations of a recent description the listener has either heard or derived." the only restriction being constituted by the fact that these generalizations must somehow be shared by the speaker. We might add that such a generalization is reached through the interpretation process: a generic reference to a definite description is interpreted as such if it is not referential. In other words, there are strong restrictions to interpret an assertion as a generic statement, and they mainly concern the interpretation of tense and its modifiers. In order to compute the reference of a definite NP as generic, tense cannot be specific, and adverbial modifiers cannot be deictic. This is clear if we look at some of Webber's examples (hers d22 through d28), as for instance (the italic is mine):

25.1 Last week Wendy again bought each boy a green T-shirt at Macy's.
   2 She's always buying them.

26.1 I see seven Japanese cars in the parking lot.
   2 They're really selling like hot cakes.

27.1 Last week Wendy bought each boy a green T-shirt at Macy's.
   2 She gives them to everyone.

28.1 Wendy bought some T-shirt yesterday.
   2 Usually she charges them, but yesterday she paid cash.

29.1 Wendy wouldn't buy a green T-shirt, because they always run in the wash.

All the items italicized are either the main verb or the adverbial modifier: tense is simple present, or progressive, and adverbials are "always, usually, really". The intended meaning conveyed by the sentences in 2. is iteration on events, the events are those of buying, selling, charging, giving. In other words it is quantification on events introduced by tense and adverbial modifiers which acts on propositions, just like quantifiers and determiners act on NPs (see Hinrichs, 1988; Bianchi & Delmonte, 1989a) and also Book 2.

Consider now a short text in which quantifiers are introduced both as antecedents and as coreferential expressions: the story tells about habitual activities people used to carry out in a
 Quantifiers and Anaphora

region big enough to encompass a microcosm in which the participants of the story feel it as their only possible world in which they are trapped.

Text A.

"1. Everybody used to sow hemp, and then they laid it to rot in the river, and from that they got the hemp ready to be spinned. 2. Only on Sundays one (the people) did not work, one (the people) did not spin. 3. But Monday morning the women had to wake up at three - my father's sisters did so - to spin the cloth they hadn't spinned Sunday night. 4. Also when husking corn-cobs, they used to wake up at three on Monday, to make up for lost work on Sunday night. 5. The young used to go and check whether the girls worked, if they spinned: 6. the lazy girl did not get married."

In utterance 1. we see that we are dealing with a habitual event occurring in the past and the subject is a universal quantifier which binds a number of pronouns in the following conjoined sentences. In utterance 2. "one" or "the people" should be made to corefer with the generic set of individual characterized by a class membership to "human", the subject NP of the predicate "sow". However, when we get to utterance 3. we are in serious trouble because the subject NP is the plural "the women", which should be understood as a subset of "the people" or the impersonal "one". Notice that in this utterance tense specification as well as aspectual and semantic attributes of the main predicates indicate the passage from a habitual into a more specific description. Also the temporal adverbial "Monday morning" no longer indicates habituality, as happened with "on Sundays" in the previous utterance. The same applies to the parenthetical "my father's sister did so", where the same inference should be triggered: from a generic class specification "everybody", "one", "the people", we pass through inclusion into "women", and now into a specific instance of a woman, a "sister". Finally in utterance 5. we go back to the general entity "the people" in order to infer that "the young men" are in an inclusion relation, and that we are still analyzing the behaviour of a subset of the people indicated by the universal quantifier at the beginning of the text. The NP "the girls" however, should be understood as cospecifying "the women" again through an inclusion relation. The final generic assertion is expressed in the past and simply means that at that time it was habitual to have that kind of situation hold for a girl.

4. Quantifiers and Interpretation

As Cooper comments (1991, 39), the computation of quantifier scope is one of the central problems to be solved when building a general system: however, there is then the further problem of what to do with the correct scopings thus obtained. It is much better to have QR rather than not having it. However, scopings is not all there need to be as to contextual information necessary to compute quantified interpretation: in particular, in the case of indefinite NPs local information might not be sufficient. The general framework as to the relation intervening between quantifiers and interpretation which can be gathered from the previous sections, can be summarized as follows:

i. since we work from a linguistic perspective, we assume interpretation to be the by-product of the processes independently set up by a certain number of modules organized in a given sequence which might be paraphrased as follows: lexical properties of categories and predicates associated with the input sentence conspire to produce an adequate syntactic and
functional representation for the utterance which contains all information to be passed to discourse and interpretation modules;

ii. as a consequence, we take antecedent-pronoun relations to be represented by a coindexing which is computed by a separate module, the one of anaphoric binding;

iii. quantifiers as antecedents of pronouns are computed accordingly as other coindexing relations: interpretation of quantifiers relies heavily on propositional properties and other grammatical issues, such as grammatical function, predicative relations, etc.

iv. generic statements are treated as non-factive clauses in that they don't contribute direct facts to be added to the model: their main relation is quantified over by some temporal quantifier that suspends its extensionality in space and time, and quantifies over spatiotemporal locations (see Barwise, 1985:3);

v. definite noun NPs do not assert the existence of some individual, they only refer to it and in doing so they presuppose its existence (see Hess, 9). They do not even assert the uniqueness of the individual. The use of the definite article with singular NP for generic statements presuppose the existence of the concept referred to, an intensional rather than extensional assertion (see Book 2).

vi. indefinite NPs may be computed as quantified NPs in case they fall under the scope of some quantifier and the proposition they are in has been interpreted as a generic statement; or else they may be computed as quantified NP and receive cardinality from the quantifier that has scope over it, as discussed at the beginning of this section; finally their interpretation may be suspended: in this case, the system generates the description of an undefined entity which is converted in the following discourse either in an individual - in case some singular definite NP is used to corefer with it, or as a set - in case no such definite NP is used, or the NP is plural or quantified. Tense may contribute an existential reading if the main verb is in the past tense, in the progressive tense or in the perfective aspect; it will contribute a non factual or generic reading in case main verb is in the future tense or in the present tense. The same will apply with modality.

In turn, quantifiers and quantified NP may contribute either a collective or a distributive reading for the participants in the event denoted by the main relation. This will cause participants to be involved in a single event or in multiple events according to the semantic features of the main verbal predicate. We shall deal with the role of participants and their semantic nature, and then pass to the problem of interpreting the internal structure of the event in which participants are involved.

We also want to establish some criteria to account for the interaction existing between the interpretation of the event/participants relation as defined above and the internal structure of the event. How is the mapping from participants to event structure related? Should we use the previous interpretation to map it onto the event structure as it is computable locally? In other words, what is the procedure that leads us from the interpretation of the participants/event function to the interpretation of the event structure?

4.1. Uniqueness and Genericity

As to the number of remaining problems, we subscribe to what R. says about determiners and distributivity: in other words, it seems possible to classify NPs according to the nature of their determiners. In this way, compositionality should be preserved when computing
vi. we split the problem of characterizing indefinites into two parts: the description of the individual and the function applying over the relation in which that individual is involved. As to the description of the individual, in case it is an indefinite quantified over by some operator, be it an "if" modal operator, or another NP, quantified or quantifier itself, the system computes at first an "entity" description with a class restriction in the form of infon, i.e. a sit with a polarity and spatiotemporal location indices; this entity might become an individual or a set in the following text in case some pieces of information are made available about that individual.

Note that this treatment is not available for indefinite NPs contained in generic assertions: in this case the interpretation is straightforward. Let's take example 8. above, which we report here below,

30. Every farmer who owns a donkey beats it.
the meaning we want to get is approximately the following,
- there is an indefinite set of farmers and an indefinite set of donkeys which are however contextually bound by spatiotemporal indices - indefinite sets are described by the attribute "class" in our model which have as cardinality the value of the quantifier associated to the "card" attribute;
- there is a one-to-one function that applies on the owning relation and the beating relation requiring its arguments to be thus distributed;
- there is a quantification over events which is the unexpressed "always" that quantifies over the spatio-temporal locations of the event described by the main situation; the owing event is computed as being implied by the main situation. This might be an overgeneralization: in fact, the spatial location might be simply inherited from the previous text and thus equated with the one computed for the subject uttering the generic assertion, since we always want utterances to be associated to a perspective or other. In that case, the index corresponding to the spatial location would be filled rather than corresponding to nil. As to temporal locations, its event/state time is the location whose reference time which might be included in some more comprehensive main location inherited from the previous portion of text. Suppose now that we are in 1950, in Boston, and these are the actual main locations, this might be the description of the utterance in the model as produced by our system:

```plaintext
loc(infon1, id1, [arg:main_tloc, arg:1950])
loc(infon2, id2, [arg:main_sloc, arg:boston])
fun(id9, one_to_one, [arg:id3, arg:id4], 1, id1, id2)
quant(id9, always, [main_tloc:id1, main_sloc:id2], 1)
sit(id9, imply, [arg:id7, arg:id5], 1, id1, id2)
class(infon1, id3)
card(infon2, id3, every)
sit(infon3, isa, [ind:id3, class:farmer, 1, id1, id2])
sit(infon4, isa, [ind:id3, class:man], 1, id1, id2)
class(infon5, id4)
card(infon6, id4, some)
sit(infon7, isa, [ind:id4, class:donkey], 1, id1, id2)
sit(infon8, isa, [ind:id4, class:animal], 1, id1, id2)
sit(id5, own, [owner:id3, actor:id4], 1, tes(f5_aa1), id2)
```
We will comment on some examples which could be a problem for our approach. They have all been extensively commented in the literature: in particular we shall draw from J.M. Gawron et al. (1991). The first example is the following one, which is their (87)

31a. Every student revised a paper he wrote. ≠ It was accepted by L&P

b. Every student revised a paper John wrote. It was accepted by L&P

In example a. the anaphoric relation is impossible, while in b. it is. Example a. contains an indefinite NP which falls in the scope of the universal quantifier since it has a pronoun "he" bound to that quantifier; example b. contains an indefinite NP which is computed as specific, so it is able to escape scopal effects by the universal quantifier and be assigned wide scope.

The second example has a possessive pronoun bound by a quantified NP,

32a. Every boy washed his car.
≠ I inspected it.

b. Ogni ragazz ha lavato la propria/sua macchina. ≠ Io l'ho controllata.

The first thing to notice is the impossibility for the NP "his car" to become the antecedent of a singular pronoun "it" in the following utterance. As the authors comment (ibid., 336), this is due to the fact that the possessive is bound to/is in the scope of the universal quantifier and thus there is no single individual available in the world for the pronoun to corefer to. Before looking at another example, let us comment briefly on the Italian version of 32. The long distance reflexive possessive pronoun "proprio" can be used in this context, because it can be bound to quantified antecedents; the personal possessive pronoun "suo", however is banned. In case we use "proprio" there is only one reading available: it is the same one made available in the Norwegian examples discussed by Sem et al. (1991), where the pronoun "sin" appears. We can also notice that, if we use "sua" in this context, it cannot possibly be bound by the quantified subject, but it will receive a discourse antecedent. In this sense, Norwegian "sin" and "hans" reduplicate Italian "proprio" and "suo" but only in quantified contexts. In a non quantified context like the following, however, "proprio" still overlaps with "sin", but "suo" is ambiguous between a "discourse" bound reading and a local subject bound reading,

33a. John kissed his wife and Bill did too.

b. Gino ha baciato la propria moglie e lo stesso ha fatto Bruno
   lit. and Bruno has done the same

c. Gino ha baciato sua moglie e lo stesso ha fatto Bruno

d. Gino ha baciato sua moglie. Anche Bruno./Also Bruno

In the b. version we only get one meaning: Gino kissed Gino's wife and Bruno kissed Bruno's wife, the sloppy reading. However in c. we can have the strict reading: there is only
one wife, which is discourse bound and both Gino and Bruno kissed her. We can also have the sloppy reading, but the reading in which there is only Gino's wife and both Gino and Bruno kissed her is not available in Italian. The reason for this unprecedented state of affairs, may be ascribed to the pronominal inventories made available by the two different language systems: as we noted in a previous chapter, English is pronominally underdetermined if compared to other languages, in particular, Romance ones. Thus, the same lexical pronoun may serve different functions. Suppose now, that what is at stake here, as usually is the case with elliptical constructions, is the recovery of structural material to be restored in before entering semantically relevant modules. In this sense, this seems the correct approach only for the English version, where the auxiliary "did" could not possibly reach the semantics deprived of its main verbal predicate. In the Italian case, exemplified both by b. and c. versions, sentences are perfectly grammatical, and coreferential meanings would have to be captured at the semantic level, where the semantically "weak" or light verb "fare"/do-make, has a deictic object NP "lo stesso"/the same with pronominal import.

Thus we assume that at propositional level, there is a substitution procedure for semantically empty relations like those supported by predicate "fare" and deictic pronoun "stesso" which call for predicates occurring with same role in the previous proposition. In our case, "fare_lo_stesso" is computed as a special predicate which requires both the main relation and its internal arguments to be recovered from the previous proposition. At this point, however, we assume that only local binding is available, since semantically determined binding operations do not search for antecedent in a semantic structure, they only copy values for variables. In particular, since the possession relation in the previous proposition was entertained between the subject of the main relation, "kiss" and the possessed object, "wife", and seen that a new subject NP has to be inherited in the newly reconstructed proposition, Bruno, the index associated to this new individual will also be inherited by the possession relation.

Consider now the case represented by 33e. where the elliptical proposition is now a new separate utterance. We said that intrasententially strict reading was allowed only in case the pronoun had an external antecedent: in the d. version of the example, we see that all the readings in b. and c. are available, but one is lacking from the corresponding English version. We assume that the difference consists in the fact that at this level of computation, the newly generated abstract representation is directly carried out in the syntax, so that the new discourse would be:

33e./i. Gino ha baciato sua moglie. Bruno ha baciato sua moglie.

This structural substitution is then carried out before f-structure and pronominal binding takes place. We see that in the external antecedent case, neither Gino nor Bruno are married so that no wife is available in the preceding model; in the strict reading there is a wife salient in the adjacent text span and this is the wife that both Gino and Bruno kiss; in the sloppy reading both Gino and Bruno have a wife and each one kisses his own wife. However, the strict reading in which both Gino and Bruno are kissing the same woman, and she is Gino's wife is again, unavailable in Italian. The restriction applicable here, seems to be the one that forbids a possessive pronoun to be bound outside its proposition, in case a referential subject is locally available. This restrictions is overrun by the case in which, for reasons related to the domain defined belief system or knowledge base, another individual is the actual antecedent of the pronoun and is not grammatically determined. This is why, we always assume that knowledge related facts as well as discourse related facts should be covered first.
We shall now comment on some examples involving negation and coreference,

34a. John didn't plant any daisies. They need water.
b. John doesn't have a car. ≠ It's green./≠ They are green
c. John doesn't have a car. They are too expensive./It's too expensive.

The semantic description of the content of the first utterance, omitting irrelevant details, is as follows,

\[
\begin{align*}
\text{class(infon1, id2)} \\
\text{card(infon2, id2, any)} \\
\text{sit(infon3, isa, [ind:id2, class:daisy, 1, univ, univ])} \\
\text{sit(infon4, isa, [ind:id2, class:[object]], 1, univ, univ)} \\
\text{ind(infon5, id1)} \\
\text{fact(infon6, isa, [ind:id1, class:man], 1, univ, univ)} \\
\text{fact(infon8, name, [john, id1], 1, univ, univ)} \\
\text{fact(id3, plant, [agent:id1, aff_theme:id2], 0, tes(f5_free_aa1), univ)}
\end{align*}
\]

In the following sentence, "daisies" is a possible antecedent for the pronoun "they": we assume that the second utterance is a generic statement and coreference may be established between an entity which is a generic class, a sit and not a fact. However, in 34b. we see that coreference is blocked since we are trying to predicate a property which is an individual level predicate, or a permanent state inherently predicated of the subject: since we don't have a specific fact, or extensional entity in the world we cannot predicate extensional or objective properties with the aim of constituting a distinguishing property of that entity.

Coming now to 34c. we notice that the same entity "car" can be used as antecedent for a plural personal pronoun "they", and is also available for coreference by a singular pronoun. The reason is now clear: this kind of predication is no longer made by an adjective which represents an individual level predicate, but it is a stage level predicate, or an evaluative predication. Being "expensive" is not an intrinsic property of cars: people who cannot afford one judge it so, and the inference we are now allowed to make simply tells us that John belongs to this group of people.

Finally let us take into account cases of opaque or intensional context, with an intensional predicate like "want", in the following examples,

35a. John thinks that he will catch a fish, and he hopes I will grill it tonight.
b. Every boy wants to eat a fish. ≠ It is boiled./≠ It is a sole./≠ They are boiled./ They are soles
c. John wants to eat a fish. ≠ It is boiled./ It is a sole./≠ They are boiled./ ≠ They are soles
d. Every boy ate a fish. They were boiled./They were soles.

In all a.-c. cases the indefinite is present in world only as an intensional object, and not as a fact: as in previous examples, we can only predicate a general property which is a stage level predicate and not an individual level property. Singular pronouns at intersentential or intrasentential level, when they corefer to the indefinite NP, indicate that it has been interpreted as an individual in the world which however is a sit and not a fact. Example d. has an indefinite in the scope of a universal quantifier as before, but the context is now transparent and referential: thus both types of predications are possible, and the pronoun must be plural.
Chapter 8

**DISCOURSE ANAPHORA RESOLUTION**

1. INTRODUCTION

In this chapter we will deal with the problem of discourse anaphora: this is a text level problem, but we have decided to include it in this book. The reason why we decided so is related to the need to keep together Discourse Anaphora and Pronominal Binding - which in turn only applies at sentence or clause level and is preliminary to Discourse Anaphora. Besides, current systems of Anaphora Resolution don’t differentiate between the two levels of computation as we do. This will be discussed at length in another Section of this chapter. The first five sections are devoted to present the theoretical background; section 6 presents the output of the Finite State Discourse Anaphora (hence FSDA) algorithm. Then section 7 will deal with the comparison with other systems and the shallow or partial version of the anaphora resolution algorithm.

Computing discourse anaphora is a local process which is carried out on the basis of the information made available by two adjacent utterances and the discourse structure. In particular, access to system modules is guided by the linguistic basis of the computation, rather than by the knowledge base, the Discourse Model (DM), which is however used in case of anaphora resolution involving inferencing with bridges. The algorithm works on the output of the previous modules: the parser and its associated module for intrasentential pronominal binding; the module for the individuation of subjective contexts and subjects of consciousness (see Bianchi, Delmonte, Pianta, Sartori, 1993); the interpretation set up in the Discourse Model for the previous portion of text by the Semantic Interpreter discussed in Book 2. The last two modules are triggered by specific items, as for instance: the presence of a Subject of Consciousness in the previous discourse segment. The Model search is triggered by failure of the two most likely candidate referring expressions to be picked up as Main Topic, thus conflicting with evaluation procedures independently set up by the algorithm. Inferential processes are only activated at a given state whenever the scoring procedures independently set up by the algorithm require a definite nominal expression to corefer in the text. They can also be triggered in a given state whenever the system is in an ambiguous context: one or two pronouns to be coreferred and no MAIN or SECONDARY TOPIC. All inferential processes are presented and discussed in detail in Book 2.

When understanding or producing discourse or text, human beings must strictly follow some general constraints underlying the distribution of information which include not only
coherence but also what might be regarded as biologically set mental constraints. These constraints determine the amount of information which may be computed at a given time when reading or listening to texts. In particular, any given text introduces one or more TOPICS (see Grosz, 1978, 1981; Webber, 1981, 1983, 1988) of discourse or themes and builds up on it a structurally and semantically coherent argumentation which consists both of generic and specific descriptions, personal or generic evaluations, expansions and so on. It is a fact, that a human being cannot possibly follow these textual processes for more than one or two TOPICS at a time, even though he may well temporarily store the information related to a previously discussed TOPIC in a memory storage. To establish coreference for a pro/nominal expression, the Main topic of discourse must be established. As for terminology, we use topic of discourse (see Bullwinkle 1977) rather than focus, an attribute which we keep for non-argument grammatical functions, which in line with LFG are derived by grammar rules at sentence level and are FOCUS TOPIC ADJunct MODifier.

2. ACCESSIBILITY AND REFERRING

Following M.Ariel(1988) we believe that referring expressions are processed in a certain way according to their inherent definition or classification in terms of referential features: these in turn determine when a referring expression must be processed, if an anaphor as soon as possible - and we compute anaphors at sentence level only, whether it can act as antecedent or not; and whether it is dependent for its reference on other expressions or it is free. A pronoun is free in a certain domain, but a noun or a proper noun is free anywhere. The context taken into account is viewed in terms of accessibility of the referent to the addressee. In case a referring expression requires Knowledge of the World, this is less accessible than previous linguistic material, which alone can provide the higher degree of accessibility for a given referent. In turn, a referring expression must be a noun or a proper noun in case of ambiguity: in a stretch of discourse, whenever a pronoun is not usable because it can cause incoherence, being ambiguously referrable to one or the other antecedent, then a noun or proper noun must be introduced.

Broadly speaking, one could say that Italian is a language structurally underdetermined but referentially overdetermined in the sense that the syntactic structure of Italian is highly ambiguous whereas the referring processes set up both at sentence and at text level are very well determined. The contrary may apply to English, which is structurally overdetermined and referentially underdetermined – more on this topic below. Italian and Romance languages allow the SUBJECT to be left lexically unexpressed or to be inverted in postverbal position according to pragmatic or discourse principles: a pronoun may be left morphologically unexpressed in case it is the Main Topic of discourse. Consider the following example, which is taken from the first version of the story of the three little pigs and has both the wolf and one little house as Topics of Discourse:

18 i. pro Notó che pro non era solida
   ii. He noted that it was not solid.

The Italian version in i. has two little pro's where the English version uses two different pronominal forms, a personal form 'he' pointing to a third person human antecedent and 'it' indicating a non human antecedent. No coreference would result between the two pronouns
when the binding module is activated. However, the Italian sentence only makes available an empty category with third person specification, more features should be provided by the grammatical module. In particular, verb subcategorization would assign a human semantic feature to the first pro in force of the fact that a SUBJect for the verb NOTE has to be 'human'. However, the second pro is associated to the SUBJect of a copulative verb, BE, which only indirectly governs this function. According to LFG, an indirectly governed function is deprived of semantic features and is interpreted in the predicate, which in our case is the adjectival SOLID. Our system would provide the little pro of semantic features percolating from the SUBJect of the predicate adjectival SOLID, and transmitting via Lexical Functional Control its features, Person Number and Gender to the controller, the non thematic SUBJect, which happens to be an empty category, deprived of a lexically expressed element - a variable in prolog. Only in this way, binding is prevented to apply and no coreference would result between the two empty categories. This is clearly shown in the f-structure representation for example 13 taken from Story1, "he began to watch the house very carefully and he noted that it was not very solid indeed".

[esso si mise ad osservare attentamente la casetta e notò che non era davvero molto solida]

main/prop/index:f41
coord:index:f8
pred:mettersi
lex_form:[np/subj/actor/[hum, anim], vcomp/prop/a/[subj=subj/actor]]
mood:ind
tense:pass_rem
cat:process
subj/actor:index:sn294
cat:[anim]
pred:esso:[pers:3,gen:mas ,num:s,case:nom]
spec:def:+
tab_ref:[+ ref, + pro, - ana, + me]
antecedent:external
interpretation:specific
vcomp/prop/index:finf73
pred:osservare
lex_form:[np/subj/exper/[hum],np/obj/th_unaff/[hum, anim, obj]]
mood:inf
tense:pres
cat:perceptive
subj/experiencer:index:sn354
cat:[hum]
pred:pPro
controller:sn294
tab_ref:[+ ref, + pro, + ana, - me]
ob/th_unaffected:index:sn353
cat:[obj, place]
pred:casa:[pers:3,gen:fem ,num: sing]
spec:def:+
mods:mod a:sa:index:sa40
cat:[evaluative]
pred:piccolo:[gen:fem ,num:sing]
tab_ref:[+ ref, - pro, - ana, + class]
adj:adj/modal:pred:attentamente
aspect:activity
aspect:achieve_tr
coord:index:f12
2.1. Topic Shift and Focus

In our system, we activate only two TOPICS at a time which may either be an EXPECTED and a POTENTIAL TOPIC in case the text just starts being analysed; or, a MAIN and a SECONDARY TOPIC, in case text analysis has already started. Pronominal and nominal expressions introduced at a given point are computed accordingly. They may either CONTINUE a Main or Secondary Topic or they may contribute to turn an Expected and/or a Potential Topic into a Main Topic. Secondary Topics may only arise whenever a given nominal expression has already been used as Main Topic and is then temporarily stored into memory.

After a Main Topic has been established new topics can be added or there can be a topic SHIFT. To add new topics it is sufficient to connect the subject of a sentence to a previously mentioned POTENTIAL TOPIC. Potential topics are all noun phrases used non-predicatively: these are included in the list called Potential Topics which works as a Stack, with a Lifo policy. However, connections can be created in various ways according to context and
situation. So inferences must be used in order to understand the conceptual link existing between the Potential Topic and the Expected Topic.

A SHIFT or Topic Movement can take place only by means of overt syntactic structures, as discussed by Sidner at length. In our framework, there is no need to mention the kind of structure being instantiated since LFG uses labels such as TOPic and FOCus for fronted constituents - i.e. for constituents which have been clefted, dislocated, extraposed, topi
calized and so on. Non-argument labels such as TOPic and FOCus in fact are discourse markers which can be used directly in our algorithm without any further elaboration.

A FOCus constituent is the trigger for TOPIC SHIFT and causes two things to happen: i. the previously established Main Topic is demoted to SECONDARY TOPIC, a role which will be explained below; ii. the FOCus constituent is automatically raised to MAIN TOPIC without the need to wait for it to be reinforced.

The SECONDARY TOPIC is a repository for constituents which have been previously used as Main Topics and may be reintroduced in the following discourse. In case the shifted topic is not 'reinforced' in the following sentence and reference to the previous Main Topic is activated by means of SUBJECT noun phrase, the Secondary Topic may be restored to its previous role. The other important function that Secondary Topic fulfils in our system is the possibility of making available more than one referring expression to corefer to, and bind, pronouns and anaphors. When this happens the Main and the Secondary Topic are the more plausible candidates by grammatical procedures, rather than by inferential processes.

The stack containing Potential Topics is renewed with each new sentence analysed, following Sidner again, who comments: "Potential foci(topics) have a short lifetime. If a potential focus(topic) does not become the focus after the interpretation of the sentence following the one in which the potential is seen, it is dropped as a potential focus(topic)".

We can try to establish some relations between the output of grammatical representation and the working of the FSDA algorithm. Whenever there is a nonargument referring expression (a Foc or a Top) we correspondingly have a Topic Shift; otherwise, the Subject usually gets the highest score, or else the Experiencer with psych verbs. In aContinuing State the Object may be used to introduce a new participant in the story, in case both have to appear together in the following text. We use agreement and selectional restrictions to check for coreference with pronouns; else, if a common or a proper noun is used we activate inferential processes. Lexical entries do not in themselves mark TOPic/FOCus, but lexical redundancy rules do. In fact there is more than one case in which TOPic information is automatically associated with functional assignment: subject inversion in locative contexts (see Bresnan & Kanerva, 1989) is one such case. FOCus/TOPic labels are attached to NPs according to grammatical functions and other lexical information at sentence level and they play an important role in the global analysis of a text. Features associated with definiteness include the following: [+def] for definite NPs, [-def] for indefinite NPs, [0def] for naked nouns, [-class] for proper nouns, [+class] for common nouns, [+pro] for pronouns or pronominalized NPs, [+ana] for short/long anaphors, [+pro,+class] for deictic pronouns and [+pro,+part,+class] for nominal substitutes.

2.2. Comparing Getaruns with other Systems
In the following, we shall discuss differences between our algorithm and Sidner's (1983) and next Brennan et al. (1987) algorithm.

Every time a topic has become such, the Model builds up an Initial Description (ID) for it in a situational semantics notation, explained in Book 2. The ID associated to a given referring expression may be coredrefered by another nominal expression indicating a generic property or a specific role fulfilled in the text world, or cospecified by a proper name. In both cases, inferential processes are called in by the discourse module. In this sense, inferences are only drawn contextually and not as in Sidner model, to confirm choice of an expected topic. In our system the knowledge representation of the world is represented by WordNet ontology in addition to the dynamically built database of situations and facts contained in each utterance. In order to produce inferences, inherent semantic features and semantic roles associated to arguments of predicates are used at discourse level. This information is resident also in lexical forms or they can be derived from WordNet in the form of generic concepts.

Another important difference is constituted by the internal structure of the algorithm: on the basis of the argument thematic or semantic roles, Sidner (1983) distinguishes Actor Focus from Discourse Focus, which can become expected, secondary or main focus according to the structure of discourse. In our system, information about semantic roles, grammatical features, inherent features and other functional features are used by the algorithm creating the weighted list, which in turn is the only input to the discourse module: every time an utterance is processed, only the first two arguments in the Weighted List are accessible, in case no pronoun exist.

As in Sidner's algorithm, we work at first on the basis of syntactic structural cues which are directly encoded in the grammatical function label associated to a given referring expression: on the basis of that information a Foc element is computed as Main Topic without the need to wait for further reinforcement in the following utterance. However, the state of the algorithm will be Shifting, meaning that the text might continue by introducing yet another participant in the story with the same structural cues used in the first utterance. In this case, the Main Topic will be shifted to Secondary and the newly presented character of the story will become the updated actual Main Topic of discourse.

If we look at some of her examples, we can see that selectional restrictions are sufficient to prevent the choice of a wrong focus:

D17-1 Cathy wants to have a big graduation party at her house
   2 She cleaned it up
   3 so that there would be room for everybody

In our system, Cathy would be assigned to Expected Topic, "party" and "house" to the stack of Potential Topics. When the pronoun "it" is accessed, since the verb "clean" would have assigned the features [object, place] to its OBJect, and "party" would be classified as [social, event], no match would take place between the latter two and only "house" be selected as possible coreferent. No inferential devices seem to be needed for these cases. In general, we assume that whenever a pronoun is used, the author should be collaborative enough not to induce the reader into a mistaken coreference which would cause misunderstanding. We want to keep access to the knowledge base as limited as possible. The same would apply to two examples of hers:

D21-1 I want to schedule a meeting with Harry, Willie and Edwina
   2 We can use my office.
3 It's kind of small,
4 but the meeting won't last long anyway
D22-1 I want to schedule a meeting with Harry, Willie and Edwina
  2 We can use my office.
  3 It won't take very long,
  4 so we could have it in the conference room.

In both texts, the second sentence makes it clear that the Main Topic is constituted by the participants in the meeting. The meeting itself and the office are introduced simply as Expected Topic: in the first text – D21-, no real problem arises due to the fact that the pronoun "it" is used to corefer to "the office" which was introduced as expected topic in the previous utterance. In order to prevent the pronoun "it" to pick up the wrong coreferent in the second discourse – D22 -, we simply let the semantic inherent features filter out the wrong coreferent: the office would be assigned [place, social], whereas the meeting [event, social]. However, the pronoun "it", being the subject of the verb "take" will be assigned [event] which would automatically make the right selection in the Potential Topics Stack.

In their paper on the Centering Approach Brennan et al. (1987; as well as Di Eugenio, 1990) present a formalization to modeling attentional structure in discourse as a means for capturing coherence at text level. Their approach embodies a set of rules and constraints that should reflect the relationships existing between what the discourse is about and linguistic choices made by discourse participants as to their local relevance in a given discourse segment. They also establish a typology of transitions from one utterance to the next which should describe the way in which utterances are linked together in a coherent local segment of discourse. As the authors note, if a speaker has a number of propositions to express, one very simple way to do this coherently is to express all the propositions about a given entity before introducing a related entity and then perhaps shifting the center to this new entity.

Even though their paper is an excellent presentation of the problem at hand, and the algorithm seems perfectly wrought out, we assume - as the authors themselves admit - that it is only in a preliminary stage in the coverage of real texts. The algorithm is deficitary in two ways: it is too strict and too limited in scope. This might depend on the fact that the authors did not try it on an extended number of texts, perhaps because they believed that the same approach could be simply adopted as it stands to cover other more complex cases.

Our approach to the general problem of reference resolution is guided by the following economy principle:

"Try a match in the Local context on the basis of grammatically encoded information; then explore the Model and/or the Discourse Domain in search of a suitable entity; in case there is none explicitly mentioned try an inference”

2.3. Coherence as a Finite State Machine

The Finite State Machine we use at intersentential level, is not so strict as the one proposed by Brennan et al. In particular, we also build up a Weighted List of Topics (WLT) which relies on Grammatical Functions, Semantic Roles and Selectional Restrictions information associated with each referring expression of the current utterance. The first
Ref_ex in the rank list is always assigned a preferential status, however we also keep an eye to the second ranked Ref_ex. The rest of the stack might also be used to search for the presence of some Ref_ex which corefers with one of the previous Topics: this is done in order to ascertain whether some cohesion is present.

The representation we assign to each state of discourse is then more articulated: we use four specific Topic labels, Expected, Main, Secondary, Potential. The latter is in fact a stack, and the previous ones are individual slots. A Main Topic usually occurs for more than one single clause or utterance, and covers a stretch of discourse, until a new entity appears, and is computed as Expected Topic. Whenever a Ref_ex becomes Main Topic it might reappear as Secondary Topic in the following text even though it is not included in the list of Ref_ex for that utterance. This is done to establish the persistence of a Main Topic in the discourse and for possible recovery by grammatical rather than inferential means. Matching procedures are much quicker and much easier than inferential processes on the Model Knowledge Base.

In case a pronoun has been used to corefer, a description of the head it has been bound to should be present in the Model. In case a nominal head has been used, this might or might not be present in the Model.

In the first sentence of a text the current Expected and Potential Topics are chosen from referring expressions available according to their rank, the Main and Secondary Topics are asserted, usually by reinforcing the Expected or the Potential Topic. From now on, however, the Model of discourse is available as well as the Domain of Consciousness: these may be accessed to guide the choice of the Main Topic from the currently available Weighted List of Topics or from the Model or else by picking the Subject of Consciousness.

Another important criterion for the setting up of a Main Topic is the state assigned to the previous segment of discourse: we use the following ones,

- CHANGE, CONTINUE, RESUME, RETAINING, SHIFTING, CONTINUE_ANALYSE

Whereas a Change state indicates that a new entity is being referred to by some nominal or pronominal expression and is the current Expected Topic, a Continue state indicates that the same entity has been asserted as Main Topic. The remaining states are Resume, Retaining, Shifting and Continue_Analyse. Shifting is used whenever there is a grammatically marked focalized constituent: in other words, a presentational construction has been used in the text, either by Locative Inversion or a There-sentence or any other available means offered by the specific language. This causes the focalized constituent to be set as Main Topic. As to Continue_Analyse, we use it to indicate the fact that a previously described entity which was a set, is now being reintroduced as a subset.

Resume is a state that is invoked any time the first referring expression on the Weighted List of Topics is not available in the adjacent portion of text and must be recovered from the Model. The entity is set as Expected Topic. Finally, a Retaining state indicates that the current utterance is promoting one of the Topics present in the adjacent portion of text as possible new topic of discourse: in this case, the first referring expression on the WLT does not coincide with the previous Main Topic. It will be set as Expected Topic.
2.4. Topics Hierarchy Manipulation and Discourse States

Before entering this module, the algorithm substitutes the head of the pronoun with the head of its antecedent. In this way, all free pronouns are eliminated from the current list of referring expressions. In fact there are cases in which some entity is introduced in the discourse as an indefinite pronoun and no antecedent may be found in the adjacent stretch of discourse. These cases are dealt with in the Model, by asserting the indefinite entity as an entity belonging to the class defined by semantic selectional restrictions associated to that NP by the verb.

In all other cases, we always deal with nominal heads of some kind, be they proper or common nouns. The module has two main tasks: one of coupling the first or the second ref_ex in the weighted rank list with any of the previously asserted Topics; the other of asserting the new state of discourse.

The state of discourse guides the selection of matching procedures between current referring expressions and previous Topics. In case the coupling or matching does not succeed because there is no relation between the adjacent discourse segment and the current sentence, the state of discourse should be a shifting. Or else the Discourse Domain should provide a Subject of Consciousness to introduce as Main Topic to mark the fact that the current sentence is a subjective sentence. In this case the first referring expression in the list will be asserted as Expected Topic, in case the following text might be shifting to this newly introduced entity.

This module accesses the World Model of the previous text whenever needed. In particular there are two possible situation in which this is compulsory: whenever the current ref_ex is a proper name which however corefers with a property already associated to that name in the previous text. In this case the algorithm should compute this as a Continue and a reassertion of the same Main Topic.

A subcase is represented by the situation in which an entity is simply reintroduced with its proper name but it was not present in the adjacent portion of text. The entity is simply recovered from the model and asserted as Main Topic. Another subcase is constituted by the case when the current ref_ex is a proper name but the previous Main or Expected Topic was a plural common noun, a property already associated with the current name.

2.5. Deictic Pronouns

A number of interesting phenomena can be covered by an adequate linguistic representation and they concern discourse bound referring elements like deictic pronouns, quantified expressions and nominal expressions with an indirectly governed function. Discourse bound pronominals can be divided up into different types (partially following Bosch 1981): anaphoric ones, 'which continue or sustain a previously established focus towards a specific item which he had oriented his attention to before'; deictic ones, 'which are a means for achieving the focussing... of the attention towards a specific item which is part of the respective deictic space'(ibid.,68).

The use of deictic pronouns is quite common in texts and ranges from first and second person personal subject pronouns, to deictic demonstrative pronouns like 'this' and 'that', which in Italian can be used to corefer both with a human referring antecedent and with a non
referring sentential antecedent - differently from what happens in English (see Halliday and Hasan 1975, 63). Consider some of the examples included in our texts: the most interesting case is represented by utterance no.9 of Story 3. "Like that of straw (it) was not surely very resistant". In this utterance we see that the entity referred to by the deictic is not locally available, and must be recovered from the previous discourse segment. In fact the Main Topic of discourse in the previous utterance is "casa/house", and this will be linked both to the subject little_pro of the copulative construction and to "quella". However the entity referred in the model will be different in each case: for the little_pro the actual nominal reference will be assumed and its identifier recovered directly from the model without any further computation; in the case of the deictic, the system will look for a house which has the property specified in the adjunct of matter, "di paglia/of straw". As a result of the overall computation, both houses, the current one and the one recovered from the model will be ascribed a property, the one of not being resistant.

Another interesting case is constituted by the deictic "questo/this one" used to pick up the Secondary or Expected Topic in the utterance, and is from Story 1., "These, then, to protect themselves from the wolf,...". At this point of the story both the little pigs and the big bad wolf have been introduced by presentational utterances: however, the current Main Topic in the previous portion of text is the wolf, and this has been achieved by a shifting which has turned the previous Main Topic into a Secondary and the newly introduced entity, the wolf, into the current Main. In order to attach reference of a pronoun to the Secondary, a deictic must be used, since a normal personal pronoun, or an empty pronoun will simply be attached to the Main Topic. The same applies to utterance no.9 of Story 1. where we see that there is an alternation between the wolf and the little pigs in the previous three utterances, but in the last previous one the wolf is Secondary Topic while the little pigs are Main Topics. So the utterance refers back to the wolf by a deictic, "Questo intanto si leccava i baffi.../This was licking his lips meanwhile..."

Another linguistically interesting use of a deictic can be found in copulative constructions like the ones used to start a story, "This is the story..." or the one appearing in Story 2., utterance no.5, "This was a task". From a grammatical point of view we might regard the deictic as the predicative nonreferring portion of the sentence in case the example is "This is the story...", whereas in the second case, "This was a task..." we might regard it as the referring non predicative portion of the utterance. In the first case, "this" is used to ostensively point to the story in what will follow: in other words, it is used for identificational aims. On the contrary, in the second case, it is used for referring aims: this point to a portion of text or discourse, or to some property already made available in previous text.

There are two more interesting instances of deictic pronouns in the text we studied, which are reported here below in the section dealing with Domain of Point of View. The first one, is in Text 2., utterance no. 4 "He felt strongly about all this". The reference of the demonstrative "this" reinforced by the presence of the quantifier "all" is a section of text or a discourse segment in Webber's(1991, 114). In our case, the segment being referred to is constituted by all the previous utterances with the exception of utterance no.1. This is computed by the semantic interpreter at discourse structure, which is shown in Chapter VII: all clauses at the "same_level" in the previous portion of text are taken to constitute or to be included in the same Discourse Segment, that is the set [clauses: 2-2, 2-3, 2-4, 3-5] where the first number individuates the utterance, and the second the clause.
The other interesting use of a demonstrative is in Text 3., where we find the phrase "such a move" in utterance no.2. The indefinite NP is taken to be deictic by the presence of "such" and is computed as a textual deixis: the entity being referred to in this case is a relation, that is the calling event. This is due to the semantical restriction associated with the noun "move", which is classified as an "event". Event nouns require to be coreferred to relations in case they are used as anaphora. We might as well have decided to associate a discourse segment to the noun "move": all the preceding clauses would have constituted a suitable discourse segment in that case. That is the move would be understood as both the pick_up and the calling events.

2.6. Nominal Anaphora or Bridges

As said previously, we let the algorithm activate inferential mechanisms whenever a nominal expression appears as first or second in the rank weighted list and it does not match any of the Topics available in the previous discourse segment, in their current linguistic description: the resulting state will be a RESUME. No such triggering takes place for SHIFTING from a given Main Topic into a New Main Topic. In a state of CONTINUE inferencing may be activated also accessing grammatical representation. The result of scoring is used to set up adequate conditions for triggering inferential processes both in presence of a pronominal and a nominal expression. However, the behaviour of pronominals is only determined after grammatical constraints are satisfied at sentence level. As to what triggers an inference to be drawn Ehrlich(1981) clearly states the point: at first a relation between expressions must somehow be perceived before an inference is drawn, and this relation is syntactic and semantic in nature. As she comments 'people do not draw inferences randomly to relate linguistic expressions', showing how in two examples people related 'bus' and 'vehicle' only when certain conditions obtain; the examples she uses are the following ones:

33i. A bus came roaring round the corner.
   ii. The vehicle nearly flattened a pedestrian.
34i. A bus came roaring round the corner.
   ii. It nearly smashed one vehicle.

In the first example an entity 'bus' is introduced in the discourse as the Topic and then reinforced in the following sentence by using a class noun 'vehicle' which subsumes the reference of 'bus'. Criteria for relating the two referring expressions are the use of SUBJ function in the second sentence, together with definiteness. In example 34, once the pronoun 'it' is processed as anaphoric to the Expected Topic, it must be obviative with any other referring expression contained within the same f-structure, the sentence. In our system triggering material may derive both from sentence level analysis and from discourse level one. In case the Expected Topic is rejected as possible coreferent of the pronoun and another phrase is chosen, the rejected phrase is retained for possible re-introduction later in discourse. Rejection is possible when compatibility requirements and/or agreement tests are not met.

Once the algorithm has independently ranked a nominal expression rather than a pronominal as best candidate to corefer with the previously established Main Topic, a number of subcases may be detected and are all present in the texts analyzed:
CASE 1:
A. A proper noun is used to corefer with a common noun in the previous sentence: the proper noun has already been associated to the property specified by the common noun in a previous segment of text;
B. Two or more proper nouns are used to corefer with a plural common noun: the same procedure as before, but in addition, cardinality of the common noun must be checked in order to match it with the two or more proper nouns under analysis;

CASE 2.
A. A common noun is used as a property to corefer with a proper noun in the previous discourse segment: it can either be already assigned as a property to the individual with that proper noun or be inferred from world knowledge.

In our framework we can easily distinguish SUBJect from OBJect nominals, and furthermore SUBJect nominals in copulative constructions from the rest. In predicative contexts, the nominal expression has already been computed as an indirectly governed function by the parser: these are SUBJ or OBJ functions which are not directly governed by the main predicate of a sentence, but by a predicate function, such as the ACOMP or NCOMP contained in the lexical forms of verbs such as 'BELIEVE<SUBJ, ACOMP>OBJ', or 'BE<NCOMP>SUBJ'. We compute the difference in government by associating a special semantic role, THEME_BOUND to the governed or non-thematic function.

3. DISCOURSE DOMAINS AND SUBJECT OF CONSCIOUSNESS

J.Wiebe and W.Rapaport in their paper(1988) and J.Wiebe(1993) present a computational theory for recognizing discourse passages which are told from the perspective of a character. In what follows, I shall only comment on Wiebe's latest paper which enhances the framework she presented in the previous paper. The final aim is as before producing an algorithm for computing Ann Banfield’s (1982) - which we also assume to be a landmark for literary text analysis from a linguistic perspective - categorization of sentences of a narrative text into subjective and objective sentences. In some cases, she criticizes Banfield’s categorization of sentences since it is clear from W’s work that it is not sufficiently extended for establishing the current perspective.

When a subjective context is independently established by the presence of some subjective element, we are dealing with the character’s beliefs, seen that what is being told in the narrative reflects his perspective.

Let’s consider more closely Banfield’s categorization: a subjective context is established by subjective sentences which may either portray the character’s thoughts (represented thought) or present a scene as a character perceives it (represented perception). In particular, such verbs as “hear, see, realize, know, think, wonder, remember, want” are regarded as markers of subjectivity.

However, in our system, these verbs are classified into separate categories:

a. Mental activity verbs
   think, wonder
b. Stative, Presuppositional and Factive verbs
see, hear, remember, realise, know

c. Intensional verbs
want, desire, wish, expect

d. Subjective
consider, believe

It is our opinion that only a. c. and d. verbs may attribute a sentence to a subjective domain; on the contrary, stative and factive verbs only depict objective facts: as a consequence, the object of any such verbs is always regarded as a state or a fact in the world, and should be so understood by the reader.

More information about this problem can be found in W's paper, where we see that she rightly differentiates potential subjective elements from explicit lexical items that indicate a subjective context. As she comments, most of the elements are lexical, however the meaning instantiated in the particular context must always be checked. Among these elements, we may count all linguistic elements which express emotions, uncertainty, evaluations: they usually report the private state of the Subject of Consciousness, such as believing, wondering; emotive states such as hating, loving or being afraid. As W.(ibid.,3) comments, these states are not open to objective observation or verification. Further on in the paper, we see a number of tables(ibid., 36-38) where these elements are listed: they contain, exclamations and direct questions; elements that express evaluation or judgement, such as appropriate adjectives or epithets, adverbs and auxiliaries like modals; finally elements that express a lack of knowledge, such as interrogative pronouns and adjectival phrases such as "some kind of".

Other lexical elements are evidentials, conjuncts, comparatives with "like", and two tenses when used in main clauses: past perfect and progressive.

Also in this second paper, we noticed a lack of consideration for structural contribution to the categorization of sentences in a subjective or objective domain. Though, we found that sometimes W. refers to "sentence fragments" as a whole, or to phrases: the overall idea she wants to convey is that in order to tell a subjective from an objective context all one needs is access to meaning through context. Consider a verb like "say" or "tell" which are classified as Reportive verbs in our system: in case the addressee is a reflexive pronoun as in one of our examples, the context is presubjective. “John said to himself that...” is understood as the indication that something objective is happening, the saying relation, which however preludes to a subjective context. In other words, in this situation, the system sets up an initial boundary for subjective domain, as discussed in some detail here below. Another important element which we use throughout our algorithm is "aspect", which is paramount both for adjective and verb's meaning contribution: in W.'s paper there is only a footnote where she postpones the analysis of aspect to future work.

### 3.1. Subject-of-Consciousness and Domain-of-Point-of-View

Subject-of-Consciousness is a semantic property assigned to a referent whose thoughts and feelings are represented by a portion of discourse.

Contemporary authors like Virginia Woolf and James Joyce are masters in the use of this subjective fiction, where subjective means that facts are depicted as filtered by a conscience,
that of the Subject-of-Consciousness, contrary to the objective narrative where no consciousness intervenes.

The Subject-of-Consciousness is a concept already pertaining to literature that has to be formalized because it is a factor that plays a fundamental role in discourse and in particular it helps the binding of Long-Distance-Bound anaphoric expressions: the Subject-of-Consciousness results as the antecedent of such LDB anaphoric expressions. As a result, to be able to identify it, is paramount for the interpretation of anaphoric processes.

Our aim is to create a Domain where entities introduced in the previous discourse structure do not count as possible Topics and thus they cannot be computed as coreferents of pronouns occurring inside it. The Domain is a referential island, very much like a digression - as commented in Allen(1995, Chapter 16) - were pronouns occurring inside it may only pick up local possible antecedents and at the end of the digression pronouns occurring may not enter the digression domain.

In addition, new entities introduced in the Domain may or may not become referential entities according to their intrinsic semantic properties.

3.1.1. Long-Distance-Bound Reflexive Pronouns

Anne Zribi-Herz calls the first generative linguist to assume that the binding of Long-Distance-Bound (LDB) Reflexive Pronouns in English, that is, those pronouns that search for their antecedents outside the sentence domain, must be viewed from the level of Discourse Grammar. She introduces concepts such as Subject-of-Consciousness and Domain-of-Point-of-View to demonstrate that it is necessary to rely on semantic discourse concepts in order to explain the behaviour of LDB anaphoric expressions.

It is a fact that c-command or f-command do not rule over such occurrences of pronominal expressions and work only at sentence level.

Her analysis represents a step forward to the understanding and conceptualizing of discourse grammar and, at the same time, this kind of approach confirms that an exclusively syntactical sentence-internal approach is insufficient and inefficient.

Sells(1989) reaches almost the same conclusion as Zribi-Herz’s but inside a different theory: he introduces concepts such as SELF, PIVOT, SOURCE, that is, few discourse roles which affect the distribution of LDB pronouns.

He further shows that LDB pronouns should be read as logophoric: their antecedent is always an entity whose thoughts and feelings are represented in the clause of the pronoun.

3.2. Domain-of-Point-of-View

Before giving our formal definition of SC, we need to define the concept of Domain-of-Point-of-View (DPV), since these two are interrelated and interdependent concepts.

A DPV may be formally defined as a portion of discourse which is the grammatical expression of one and only one narrative point-of-view; the point-of-view being the perspective used by the author to describe the facts, real or hypothetical, in a text: it may be the author himself - or narrator -, the main character, other characters, or there may be no point-of-view at all.
We have identified at least four different kinds of Domain-of-Point-of-View not just two as it was suggested by Zribi-Herzt's analysis; each has its own particular structure, its features and functions.

We also built some tests that help identifying the different kinds of DPVs.

The Objective DPV (OD) is taken to be the default domain, that is, the domain in which reality is what a written text focusses on. In an objective DPV, reality is depicted as not being expression of any point-of-view, or rather, an objective DPV does not fall under the scope of any Subject-of-Consciousness.

In English, in an objective DPV usually the aspectual (or semantic) category of the verb in the main clause is a process, that is, in an OD the Temporal Focus moves.

We even noted that normally (not always) whenever we pass from a Subjective DPV to an Objective one, the old SC is re-established through the use of a proper name.

A Presubjective DPV (PSD) is still an Objective DPV in the sense that there is still no SC, but the presence of a particular formal mark acts like a door which introduces necessarily as its object an Explicit Subjective Domain.

The formal mark that characterizes a a Presubjective DPV is represented by the category of the verb: in a PSD the verb may be a psych verb, a verb of mental activity, or an emotional verb.

In a Presubjective Domain the Subject-of-Consciousness of the depending Explicit Subjective Domain is instantiated through the use of a pronominal expression (i.e. it is the thinker, perceiver, or senser present in the PSD) and it is identified through a selection of semantic roles as well as on basis of syntactic information:

- \text{presubjective\_cat(subjective, [experiencer, actor, theme\_unaff])}.
- \text{presubjective\_cat(presuppositional, [actor])}.
- \text{presubjective\_cat(emotional, [experiencer, actor, theme\_emot])}.
- \text{presubjective\_cat(reportive, [actor])}.

Let us look at one example to clarify this point:

\text{Ex 1 [psd Mary felt that] [esd she was unable to say "no"]/}

Here, the main clause is a PSD due to the presence of 'felt' which is an emotional verb; therefore, it necessarily introduces an Explicit Subjective Domain whose characteristics will be discussed below.

Note that in the PSD the SC is established "Mary", that is the "Senser", and that in the ESD it is present in the form of a personal pronoun.

The following utterance is part of Text 1,"She knew that she had been given a present, something precious".

\text{DISC-DOM: presubjective(3-n1, sn3, she) SOC: none}
\text{DISC-DOM: explicit\_subjective SOC: she/sn3 from 3-n1}

\textbf{3.2.1. Explicit and Implicit Subjective Domain-of-Point-of-View}

The preceding example shows what happens as a rule: there is a strict correlation between the Presubjective DPV and the Explicit Subjective DPV (ESD), that is, the ESD always follows a presubjective DPV and the SC of the ESD is always instantiated in the PSD.

Direct speech is always treated as an objective domain, rather than as an Explicit Subjective DPV: when we are in presence of a direct report it is evaluated objectively by the
reader not as something reported by the SC but as something viewed from an external position.

The Implicit Subjective DPV (ISD) is more complex, but more interesting for our purposes: it expresses the thoughts and feelings of the current SC which is not syntactically present and remains implicit. The important fact is that all the pronouns (especially LDB reflexive pronouns) found in an ISD all refer back to the current SC.

First, we have realized that the Subject-of-Consciousness of an ISD is always the SC of the last preceding ESD. Thus, we always require the preceding DPV to be an Explicit Subjective one.

Furthermore, in order to individuate ISDs we have formulated some tests that allow us to single out this kind of DPVs, that is, we have found some formal marks which are sufficient conditions for Implicit Subjectivity.

If a verb is a state or indicates existence, this confirms Subjectivity. Secondly, we have seen that exclamations, questions in indirect report and the presence of modals (i.e., would, could, should, must, may) or intensional verbs are all manifestations of the current Subject-of-Consciousness and then always indicate Implicit Subjective DPVs.

As also reported in W & R’s paper, few adverbs seem to be 'subjective', as they express the will of a character and his personal judgement: for instance, 'of course', 'perhaps', 'literally', 'obviously' and many others.

In conclusion, we have individuated four different DPVs which establish the SCs, but there may be other factors at work, semantic or syntactic, which will allow us to deepen our knowledge of discourse segmentations into Domains-of-Point-of-View.

3.3. Building Informational Structure

In the FSDA algorithm, we take into account both semantic and syntactic structural considerations: in the tables here below the information recovery to be used for further computation is shown.

In order to produce the adequate representation we access f-structure dags and recover information related to Speech Act if present, Semantic Category of Main Verb, presence of Modals, Mood, Tense, Aspect as computed at f-structure level, Polarity, Voice, Function and Role if not Main Clause. These items are used to compute View, Intensionality, Change in the World, Relevance, and they will be used to build up the following intermediate representations: i.e. Temporal Relation, Discourse Relation, Discourse Domain, Subject_Of_Consciousness

We discuss here below the main four informational items in more detail, by showing their calls, the linguistic items they need as input to produce their output:

1. view(NoFr,FunRole,Speech,SemCat,Voice,Verb,Support,View)
2. factuality(Verb, Tense, Mood, FunRole, Support, Intensionality)
3. change(Speech,Pol,Support,View,Mood,Tense,Aspect,Change)
4. relevance(Change, Relevance)
3.4.1. **Point Of View**

The PointOf View takes as input Function and Role, Speech Act Type, Verb Semantic Category, State, Discourse Domain, SupportVerb if present, and outputs PointOfView which may be either internal, external or social_engagement as shown in the Table here below:

**Table 28. Point Of View**

<table>
<thead>
<tr>
<th>Function/Role</th>
<th>Speech Act</th>
<th>Semantic Category</th>
<th>State</th>
<th>Discourse Domain</th>
<th>PointOfView</th>
</tr>
</thead>
<tbody>
<tr>
<td>direct</td>
<td></td>
<td></td>
<td></td>
<td>internal</td>
<td></td>
</tr>
<tr>
<td>adj/hypothesis</td>
<td>indirect</td>
<td></td>
<td></td>
<td>internal</td>
<td></td>
</tr>
<tr>
<td>main/hypothesis</td>
<td>indirect</td>
<td></td>
<td></td>
<td>external</td>
<td></td>
</tr>
<tr>
<td>indirect</td>
<td>indirect</td>
<td>perlocutory</td>
<td></td>
<td>social_engage</td>
<td></td>
</tr>
<tr>
<td>indirect</td>
<td>indirect</td>
<td>intensional</td>
<td></td>
<td>internal</td>
<td></td>
</tr>
<tr>
<td>indirect</td>
<td>indirect</td>
<td>existence</td>
<td>continue</td>
<td>implicit_subjective</td>
<td>internal</td>
</tr>
<tr>
<td>indirect</td>
<td></td>
<td></td>
<td></td>
<td>external</td>
<td></td>
</tr>
</tbody>
</table>

3.4.2. **Factuality**

Factuality takes as input Verb, Tense, Mood, Function and Role of current proposition, Support verb if present, and outputs Intensionality: it may be either factual or nonfactual, intensional or extensional:

**Table 29. Factuality**

<table>
<thead>
<tr>
<th>Main Verb</th>
<th>Tense</th>
<th>Mood</th>
<th>Function/Role</th>
<th>Support</th>
<th>Factuality</th>
</tr>
</thead>
<tbody>
<tr>
<td>vcomp/ _</td>
<td>nonvar</td>
<td></td>
<td></td>
<td></td>
<td>factual</td>
</tr>
<tr>
<td>vcomp/ _</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>nonfactual</td>
</tr>
<tr>
<td>_/hypothesis</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>nonfactual</td>
</tr>
<tr>
<td>fcomp/{ propq,propint}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>nonfactual</td>
</tr>
<tr>
<td>adj/condition</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>nonfactual</td>
</tr>
<tr>
<td>irrealis</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>nonfactual</td>
</tr>
<tr>
<td>Future; present</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>nonfactual</td>
</tr>
<tr>
<td>Any modal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>factual</td>
</tr>
</tbody>
</table>

3.4.3. **Change in the World**

Change takes as input Speech Act, Polarity, View, Tense, Aspect, and outputs a Change marker, which may take one of the following values: negated, null, culminated, gradual, earlier, as can be seen from the following Table 30.
3.4.4. Relevance

Finally, relevance may be either background or foreground:

\[ \text{relevance(null, background)} : \neg !. \]
\[ \text{relevance(_, foreground).} \]

Eventually all these intermediate informational representations are used to produce the Domain of Point of View and Subject of Consciousness.

Table 30. Change in the World

<table>
<thead>
<tr>
<th>Speech</th>
<th>Polarity</th>
<th>Support</th>
<th>View</th>
<th>Mood</th>
<th>Tense</th>
<th>Aspect</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Conditional</td>
<td>null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>no</td>
<td>nonvar</td>
<td></td>
<td></td>
<td>state</td>
<td>negated</td>
<td></td>
<td></td>
</tr>
<tr>
<td>no</td>
<td></td>
<td></td>
<td></td>
<td>state</td>
<td>null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>no</td>
<td></td>
<td></td>
<td></td>
<td>negated</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>internal_ intension-al</td>
<td></td>
<td></td>
<td></td>
<td>state</td>
<td>null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>internal_ extension-al</td>
<td></td>
<td></td>
<td></td>
<td>state</td>
<td>null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>indirect</td>
<td></td>
<td></td>
<td>present</td>
<td>null</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>indirect</td>
<td></td>
<td></td>
<td>imperfetto</td>
<td>state</td>
<td>null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>indirect</td>
<td></td>
<td></td>
<td>past</td>
<td>Culminated</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>indirect</td>
<td></td>
<td></td>
<td>imperfetto</td>
<td>≠ state</td>
<td>gradual</td>
<td></td>
<td></td>
</tr>
<tr>
<td>indirect</td>
<td></td>
<td></td>
<td>pluperfect</td>
<td>earlier</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>direct</td>
<td></td>
<td></td>
<td>Imperative</td>
<td>null</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>direct</td>
<td></td>
<td></td>
<td>future</td>
<td>null</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>direct</td>
<td></td>
<td></td>
<td>present</td>
<td>state</td>
<td>null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>direct</td>
<td></td>
<td></td>
<td>perfect</td>
<td>Culminated</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>direct</td>
<td></td>
<td></td>
<td>pluperfect</td>
<td>earlier</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>direct</td>
<td></td>
<td></td>
<td>present</td>
<td>Activity</td>
<td>gradual</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>null</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3.4. Domain Boundaries

Recognizing the structure and characteristics of each Domain-of-Point-of-View is paramount for individuating the Domain Boundaries: since a piece of literature is a sequence alternating Subjective and Objective DPVs, we have to establish where they begin and where they end.

For our purposes, what is relevant is the capability to detect the Subjective Boundaries: we have assumed that a Subjective Boundary possesses a formal subjective mark such as the presence of an Explicit Subjective DPV, and may potentially open an Implicit Subjective
DPV without formal marks. In this last case, the presence of a modal, or exclamative sentence acts as a Subjective Boundary.

In sum, we have established that Implicit-Subjective, Explicit-Subjective and Presubjective+ Explicit-Subjective DPVs may be Subjective Boundaries as will be shown below:

subjective_boundary(implicit_subjective).
subjective_boundary(explicit_subjective).
subjective_boundary(pre_subjective(_, _, _) + explicit_subjective).
subjective_boundary(subjective(_, _, _)).

Even if it is clear that SOC arises from the interaction of Discourse Domain as determined by previous informational computation and local discourse situation as computed from the Discourse anaphora module, other interesting elements of interaction may be derived by looking at the actual code. In particular, input information includes factuality, function and Role, semantic Category, mood, performative if present, preceding Discourse Domain, and output will be current discourse Domain, and SOC.

explicit_subjective_function_role(fcomp/prop).
explicit_subjective_function_role(fcomp/propq).
explicit_subjective_function_role(vcomp/prop).
presubjective_cat(subjective, [exper, actor, agent, th_nonaff]).
presubjective_cat(presuppositive, [actor]).
presubjective_cat(emotive, [exper, agent, th_emotive, actor]).

This is represented in a tabular fashion in Table 31. below.

<table>
<thead>
<tr>
<th>Factuality</th>
<th>Function/Role</th>
<th>Semantic Category</th>
<th>Mood</th>
<th>Performative</th>
<th>Preceding Discourse Domain</th>
<th>Discourse Domain</th>
<th>SOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>non factual</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>subjective</td>
<td>SOC</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>exclamative</td>
<td></td>
<td>implicit_subjective</td>
<td>SOC</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>modal</td>
<td></td>
<td>implicit_subjective</td>
<td>SOC</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>conditional</td>
<td></td>
<td>implicit_subjective</td>
<td>SOC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>stative</td>
<td>subject boundary</td>
<td></td>
<td></td>
<td>implicit_subjective</td>
<td>SOC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Object/</td>
<td>presubjective Categ.</td>
<td></td>
<td>pre_subjective</td>
<td>pre_subjective + explicit_subjective</td>
<td>SOC</td>
<td></td>
</tr>
</tbody>
</table>
3.5. Some Examples

Here below we report the texts we listed in the Introduction as woolfian subject-of-consciousness texts starting from Text 1 and 2: for each utterance we give the Weighted List of Topics and the Topics Hierarchy, then the type of Discourse Domain and the presence of Subject of Consciousness, specified at clause level. In Text 1 there is both a LDB reflexive pronoun "herself", and a couple of implicit utterances which require complete control of subject-of-consciousness.

Text. 1
1. [john gave mary a rose]
   DISC-DOM: objective      SOC: none
   In the following utterance, the presence of a verb like "know" induces the creation of a presubjective domain; in addition the complement clause contains a pluperfect which causes the creation of an explicit subjective domain and the appearance of the SOC.
2. [she knew that she had been given a present, something precious]
   DISC-DOM: pre_subjective(3-n1, sn2, she)   SOC: none
   DISC-DOM: explicit_subjective     SOC: she/sn2 from 3-n1
   In the following utterance there is a new character, Steve, who "says" something which related to a subjective domain, as the predicate "enjoy" clearly indicates:
3. [when steve faced them saying: "are you enjoying yourselves"]
   DISC-DOM: subjective      SOC: she/sn2 from 3-n1
   The following utterances are both implicit exclamations ascribed to one character, the SOC, i.e. "she" and Mary indirectly.
4. [it was horrible! it was shocking!]
   DISC-DOM: implicit_subjective     SOC: she/sn2 from 3-n1
   Also the following utterance is computed as belonging to an implicit subjective domain.
6. [not for herself.]
   DISC-DOM: implicit_subjective   SOC: she/sn2 from 3-n1
   The following utterance turns to an explicit subjective domain, mainly due to the presence of the predicate "feel".
   7. [she felt only hostility and his determination to ruin that wonderful moment.]
   DISC-DOM: explicit_subjective   SOC: she/sn2 from 3-n1
   8. [john smiled and went away embarrassed.]
   In the second text the need for a subject-of-consciousness is represented by the presence of an utterance which is an internal thought of John where however a new character is introduced: but in the following text John is coreferred by a number of pronouns one of which a LDB reflexive.
   Text 2
   1. [the three friends went all outdoors]
   2. [as they were walking in the garden, john said to himself “ sara will marry that man”, without any resentment.]
   DISC-DOM: pre_subjective(2-n3, sn42, john)+expl_subjective SOC: john/sn42 from 2-n3
   DISC-DOM: subjective   SOC: john/sn42 from 2-n3
   In the following utterance, in case the SOC was not established clearly, John would have been computed as a Secondary and not as a Main Topic; this would have caused the pronoun in the following utterance to be bound to Richard, the Expected Topic, and not to John with a complete change of perspective.
   3. [richard would marry sara.]
   DISC-DOM: subjective   SOC: john/sn42 from 2-n3
   4. [he felt strongly about that]
   DISC-DOM: subjective(4-n1, sn2, he)SOC: he/sn2 from 4-n1
   5. [she was the right person for a man like richard.]
   DISC-DOM: implicit_subjective   SOC: he/sn2 from 4-n1
   Notice that from utterance number 4 the SOC has been recomputed, so that the lexical pronoun "he" is associated to it rather than the actual coreferent in the DM, John. This fact is to show that the SOC is computed solely on a grammatical and linguistic basis and not on the basis of knowledge of the world. Also the following LDB reflexive external pronoun would have not received the appropriate antecedent in case the SOC was not available: in particular, Richard could have been used as possible antecedent as well.
   6. [for himself he was absurd.]
   DISC-DOM: implicit_subjective   SOC: he/sn2 from 4-n1
   7. [his demands upon sara were absurd.]
   DISC-DOM: subjective   SOC: he/sn2 from 4-n1
   8. [she would have accepted him still if he had been less absurd.]
   DISC-DOM: subjective   SOC: he/sn2 from 4-n1
   9. [richard began to sing.]
   The following is Text 3 where the LDB reflexive pronoun "herself", being in an Implicit Subjective DPV, is computed as coreferent with the Subject-of-Consciousness previously established with the procedure we have described in this paper.
   1. [mary picked up the phone and called jason]
This first sentence consists of two coordinate clauses; the DPV is objective, there being no subjective markers, this is further confirmed by the presence of two non-stative verbs.

2. [her husband, she thought, would have considered such a move as untruthful and utterly base]

   DISC-DOM: presubjective(2-n2, sn23, she)  SOC: none
   DISC-DOM: explicit_subjective  SOC: she/sn23 from 2-n2

   This construction is an indirect report where the main clause is a Presubjective DPV due to the presence of a mental activity verb: the reportive clause is an Explicit Subjective DPV whose SC is 'she', that is, the 'senser/thinker' of the verb in the main clause. The ESD is a Subjective Boundary.

3. [perhaps there was something in herself that could not help but do the wrong thing at the wrong time]

   DISC-DOM: implicit_subjective  SOC: she/sn23 from 2-n2

   Being the preceding DPV a Subjective Boundary, the presence of a stative verb in this sentence confirms that this is an Implicit Subjective DPV. Other indicators of subjectivity are the possibility adverb “perhaps”, the presence of a predicative complement which is an “assertive indefinite pronoun” leaving the reference unspecified. The reflexive pronoun "herself" is successfully bound to the Subject-of-Consciousness established in the last ESD.

   Consider again Text 1.: when utterance 3 is processed, a subject of consciousness is established with the pronoun “she”, which is resolved in the Model as coreferring to Mary. Utterance 4. has a plural pronoun and no antecedent available in the adjacent text segment. The Model is invoked in order to search for two fully specified individuals, John and Mary, which are assigned as antecedent of “them”: this in turn binds “you” in the following complement clause. Utterance 5. is a case of Implicit Subjective Domain whose content is ascribable to the Subject of Consciousness(S-C), and is an evaluation of a previous assertion: “it” is bound to a previous relation. Also Utterance 6. is a case of Implicit Subjective Domain where the LDBR is bound to the S-C. Also consider the binding of “his” in utterance 7. which has no local antecedent, but is computed on the basis of “recency” criteria: Steve is the most recent proper name, or fully specified individual available as a singular entity.

   Now consider Text 2. where a S-C is established in utterance 2, and is a proper name, “John”. In the following utterance, the system computes an Implicit Subjective Domain, a nonfactive assertion, a Continue with John as Main Topic, and the same S-C. In this way, when utterance 4. is reached, the pronoun “he” is bound to John rather than to Richard. Utterance 5. is again an Implicit-Subjective Domain so that when the following utterance is processed, the LDBR “himself” is adequately bound to John. More about the other texts in the following chapter.

4. The Algorithm

We shall now go into some technical notions contained in an implemented algorithm to compute anaphora resolution and coreference, relatively to pronominal and nominal expressions in discourse.
The algorithm for discourse analysis takes an utterance at a time and looks for antecedents to pronominals which have been assigned to an EXTERNAL referent by the binding algorithm at sentence level.

We proceed then, on the basis of the following principles:

- there is only one possible TOPIC of DISCOURSE which may be assigned as the MAIN TOPIC for each sentence;
- there may be another topic which is assigned as SECONDARY TOPIC and may be computed together with the MAIN for each sentence;
- TOPICS may be changed according to states of the discourse model: in this case we follow Brennan’s (1986) approach to centering, even though we use a different set of states, and apply some changes in the overall strategy.

We analyse any sequence of utterances as if it were a text and the analysis is carried out in a number of steps. We shall illustrate the work of the algorithm in detail by commenting each call separately. This is the higher call of the FSDA:

$$discourse\_anaphora(Net, \text{Utt}_\text{No}) : -$$
$$extract\_arguments(Net, \text{RefList}),$$
$$corefer(\text{Utt}_\text{No}, \text{RefList}), !.$$

The module for discourse anaphora receives the current utterance f-structure dag as realized by the parser. The first call is to extract_arguments, i.e. the first thing to be done is to organize the list of referring expressions contained in the utterance under analysis, by the creation of a rank list where most likely candidates to become topics of discourse are ranked higher than other referring expressions.

$$extract\_arguments(Net, \text{RefList}) : -$$
$$bagof(\text{Ind}, N1^N2^N3^\text{arc}(N1,N2,\text{tab}\_\text{ref},Net),$$
$$\text{arc}(N1,N3,\text{index},Net),$$
$$\text{leaf}(N3,\text{Ind},Net)), \text{List}),$$
$$\text{maplist}(\text{find}\_\text{pred}, \text{List}, \text{NList}),$$
$$\text{apply}(\text{retractall}, [\text{ref}\_\text{ex}(\ldots,\ldots,\ldots,\ldots)]) ,$$
$$\text{modify}\_\text{head}(\text{NList}, \text{ListArgs}),$$
$$\text{apply}(\text{assertz}, \text{ListArgs}),$$
$$\text{maplist}(\text{find}\_\text{path}\_\text{I}, \text{List}, \text{PList}),$$
$$\text{weight}\_\text{list}(\text{PList}, \text{WeightList}),$$
$$\text{create}\_\text{list}(\text{WeightList}, \text{RefList}), !.$$

In our representation, referring expressions are all kinds of NPs, and in order to become visible to the procedure, any NP must have a referring table "tab_ref" associated to its f-structure representation. Then we use the index which identifies the ref_ex univocally and on the basis of these identifiers we proceed by building up the weighted list of all candidates to topichood. At first we build up the information vector which we call ref_ex for short, where all syntactic, semantic and grammatical information relevant for creating the weighted list is assembled. The next call collects all preds in the f-structure with their referentially relevant functional attributes, mostly contained in SPEC.
The second step is what is usually called Reconstruction in linguistic theory: i.e. we reconstruct syntactic, lexical and anaphoric controllers into their controllee constituents. This is achieved by a call to modify_head, which copies the head of a controller into the head of a controlled element. The idea is simply that a functionally, lexically or syntactically controlled element has the same reference as the one of its controller: in other words, we only need the lexical head of the controller and we may do away with all controllee. In order to do this, we must separate externally controlled pronouns, which will be made visible to the main call of the discourse module, from the rest. Also, arbitrary pronouns will be set apart, and their referential feature turned from [+ref] to [+arb]. Here below are the calls. As can be noticed, the call to modify_head has two arguments: the first one contains the actual ref_ex computed by the parser, the second one is the output of the call. The input ref_ex contains Pred and Ind to search for the Antecedent. This is then used to substitute the Pred when possible.

\[
\text{modify\_head}([\text{ref\_ex}(\text{Ind}, \text{Pred}, \text{Tab}, \text{Pers}, \text{Gen}, \text{Num}, \text{Cat}, F/R) \mid \text{Others}]),
\]
\[
[\text{ref\_ex}(\text{Ind}, \text{Ante}, \text{Contr}, \text{Tab}, \text{Pers}, \text{Gen}, \text{Num}, \text{Cat}, F/R) \mid \text{Rest})] :-
\]
\[
\text{node}(N)\::\text{index}\::\text{Ind},
\]
\[
\text{node}(N)\::\text{controller}\::\text{Contr},
\]
\[
\text{new\_head}(\text{Contr}, \text{Ante}),
\]
\[
\text{modify\_head}((\text{Others}, \text{Rest})).
\]

\[
\text{modify\_head}([\text{ref\_ex}(\text{Ind}, \text{Pred}, \text{Tab}, \text{Pers}, \text{Gen}, \text{Num}, \text{Cat}, F/R) \mid \text{Others}]),
\]
\[
[\text{ref\_ex}(\text{Ind}, \text{Ante}, \text{Tab1}, \text{Pers}, \text{Gen}, \text{Num}, \text{Cat}, F/R) \mid \text{Rest})] :-
\]
\[
\text{node}(N)\::\text{index}\::\text{Ind},
\]
\[
\text{node}(N)\::\text{antecedent}\::\text{external},
\]
\[
\text{node}(N)\::\text{interpretation}\::\text{arbitrary},
\]
\[
\text{Tab} = \{\text{Ref}, \text{Def}, \text{Part}, \text{Card}, \text{Pro}, \text{Ana}, \text{Me}\};
\]
\[
\text{Tab} = \{\text{Ref}, \text{Def}, \text{Part}, \text{Card}, \text{Pro}, \text{Ana}, \text{Me}, \text{Sub}\},
\]
\[
\text{Tab1} = \{\text{+arb}, \text{Def}, \text{Part}, \text{Card}, \text{Pro}, \text{Ana}, \text{Me}, \text{Sub}\},
\]
\[
\text{modify\_head}((\text{Others}, \text{Rest})).
\]

\[
\text{modify\_head}([\text{ref\_ex}(\text{Ind}, \text{Pred}, \text{Tab}, \text{Pers}, \text{Gen}, \text{Num}, \text{Cat}, F/R) \mid \text{Others}]),
\]
\[
[\text{ref\_ex}(\text{Ind}, \text{Ante}, \text{Contr}, \text{Tab}, \text{Pers}, \text{Gen}, \text{Num}, \text{Cat}, F/R) \mid \text{Rest})] :-
\]
\[
\text{node}(N)\::\text{index}\::\text{Ind},
\]
\[
\text{node}(N)\::\text{antecedent}\::\text{Contr},
\]
\[
\text{new\_head}(\text{Contr}, \text{Ante}),
\]
\[
\text{modify\_head}((\text{Others}, \text{Rest})).
\]

\[
\text{modify\_head}([\text{ref\_ex}(\text{Ind}, \text{Pred}, \text{Tab}, \text{Pers}, \text{Gen}, \text{Num}, \text{Cat}, F/R) \mid \text{Others}]),
\]
\[
[\text{ref\_ex}(\text{Ind}, \text{Pred}, \text{Tab}, \text{Pers}, \text{Gen}, \text{Num}, \text{Cat}, F/R) \mid \text{Rest})] :-
\]
\[
\text{modify\_head}((\text{Others}, \text{Rest})).
\]

The call to new_head actually copies the head of a control chain into that of the controllee.

\[
\text{new\_head}(\text{Index}, \text{Nhead}) :-
\]
\[
\text{node}(N)\::\text{index}\::\text{Index},
\]
Now we are ready to create the Weighted List of referring expressions with their associated information vector and weight them, by means of the call to weight_list:

\[
\text{weight_list(PathList,WeightedL)} :\- \\
\text{sn_ref(PathList,List),} \\
\text{maplist(scoring1,List,L4),} \\
\text{bubblesort(L4,WeightedL),!}.
\]

Weight_list verifies whether the referring expression possesses the feature [+ref] and also erases expressions like implicit arguments, identified by the presence of the dummy existential quantifiers "exist" as head, or the impersonal indefinite quantified pronoun SI/ONE, which do not constitute possible referring expressions in a discourse.

\[
\text{sn_ref([Ind/Pa|Rest],[Ind/Pa|Others]) :\-} \\
\text{node(N)::index::Ind,} \\
\text{node(N)::pred::Pred,} \\
\text{node(N)::tab_ref::Tab,} \\
(\text{member(+ref,Tab)},) \\
\text{Tab \[= [+ref,-pro,-ana,+me];} \\
\text{Tab = [+ref,-pro,-ana,+me],Pred\[=exist;} \\
\text{Tab = [+ref,-pro,-ana,+me],Pred\[=si].}
\]

We now proceed as follows:

\begin{itemize}
\item[a.] \textbf{we compute the distance of a given NP from the root of the graph of the sentence because we intend to privilege as possible TOPICs those NPs which are positioned at the higher levels;}
\item[b.] \textbf{we assign scores according to thematic or semantic roles and grammatical functions according to the following hierarchy:}
\end{itemize}

\begin{itemize}
\item[A] \text{AGENT}<\text{CAUSer}<\text{EXPeriencer}<\text{PERCeiver}<\text{ACTor}<\text{CAUSer_emotional}<\text{Goal}<\text{Patient}<\text{LOCative}<\text{THeme_Affected}<\text{THeme_Effected}<\text{THeme_Unaffected}
\end{itemize}

As to ADJuncts, we also established a hierarchy since we take INSTRumental to be more relevant than MODal or TEMPoral ADJuncts. Note that AGent subsumes other semantic roles like POSSessor, SOURce Info, INSTigator, etc.; GOAL subsumes ADDressee; THeme_Unaffected subsumes POSSEssion, INFormation, LOCAtion etc. This subsumption rule is extended to all semantic roles since it would be unrealistic to try establish a graded ranking function for all and each of the roles of, say, FrameNet, that we also use to classify arguments of predicates.

As to grammatical functions, the nuclear functions come first, i.e. SUBJect OBJect and OBJect2, then we have OBLique and ADJuncts. Only referring arguments are imported at this
level since non-refering ones cannot become the object of coreference (or co-specification, in Sidner's terms) in subsequent discourse. This subdivision is readily made available in LFG at the functional level, in particular because the theory distinguishes between open and closed functions (see Bresnan 1982).

After NPs have been weighted, we proceed by ordering them and then we sort NPs which have the same head erasing from the list occurrences of the same NP which have received a lower score. At this point, a semantic filter is applied, by pruning those referring expressions which have been assigned a score lower than a given threshold. The pruned list becomes the Weighted List, in which all referring expressions have been ordered according to their scores.

To summarize, the weighted list is created by eliminating all controlled elements and leaving controllers as long as their head is not already present in the list, and by filtering out those expressions which have received a very low score - or very high in our system, higher than 1200.

\[
\text{create\ list}(L1,L3) : - \\
\quad \text{create\ list1}(L1,[],_L2), \\
\quad \text{filter}(L2,L3,1200).
\]

Now we are ready to call the actual modules for anaphora resolution, and this is done by activating the call "corefer".

\[
\text{corefer}(Uttt\_No,RefList) : - \\
\quad \text{externals}(Uttt\_No,RefList,List), \\
\quad \text{discourse}(Uttt\_No,List).
\]

The call to externals is activated in case an external pronoun is present in the actual utterance. In case no external pronoun is present, the list of referring expressions is simply passed down to "discourse".

\[
\text{externals}(Uttt\_No,RefList,List) : - \\
\quad \text{findall}(\text{externals}(\text{Snx}), (\text{node}(N)::\text{index::Snx},\text{node}(N)::\text{antecedent::external}),L), \\
\quad ((L [= ][]), \text{apply}(\text{assertz},L), \\
\quad \quad \text{resolve}(Uttt\_No,RefList,L)) \\
\quad ; (L = [], \text{List}=\text{RefList}).
\]

Whenever an external is present, the call passes to "resolve" which has the task to find an adequate coreferring expression in the previous portion of text. As will be made clear in the examples below, there is a wide difference in the behaviour of independent pronouns which have no deictic import, from those which are computed as deictics. In turn, nominal substitutes are also computed as pronominals with a quantified specifier. Pronouns may then be focalized in order to make them contrastive. Generally speaking, external pronouns are bound to their antecedents in discourse mainly on the basis of their grammatical features and the state in which the discourse algorithm is set at a certain point of the analysis.

The result of the discourse binding module is the modification of the pronominal head by that of the antecedent, a procedure similar to the one we performed previously for controlled chains. The ref_ex with the new head is copied in the output list and passed on to "discourse".
At this point we are ready to compute a new state of the discourse according to what happened in the previous portion of text. The call to "discourse" takes care of bound externals but also and foremost of coreferring nominals. Every time the call to "externals" fails, the text internal cohesion mechanisms must be accounted for on the basis of the presence of some nominal expression, either a common noun or a proper noun. Different procedures are activated in case one of these two basic nominal types are identified by the system. However, remember that we have already resolved external pronouns and that no other pronominal expressions is now present in the argument list.

5. RESOLUTION STRATEGIES AT DISCOURSE LEVEL

As discussed above, pronominal binding at discourse level may by achieved in three different ways which will be reviewed in the following sections: the first mode is by simply matching functional features between the pronoun/s and the possible antecedent/s available in the adjacent discourse segment. This is what we dubbed as grammatical mode. In this case, the system activates a strategy for choosing the most adequate antecedent which is based on a number of cues automatically set up by the general algorithm. Discourse states provide a first cue: in case a Continue is present, a Main Topic should be available; in case a Change is present in the previous stretch of discourse, an Expected Topic is available and so on. In case there is only one pronoun, a first match is tried with the adjacent most relevant possible antecedents. However functional features may direct the choice to other previously asserted Topics, like Secondary or Potential Topic. When two pronouns are present, the Weighted List of current referring expressions is used as the main cue for finding possible antecedents. Actually all other cues, i.e. discourse state and previous Topics relevance should conspire in directing the algorithm to the best choice.

5.1. Local Procedures

The higher procedure calls local procedures according to the number of external pronouns to be solved - also definite NPs might be included at this level of analysis.

\[
\text{resolve}(\text{StN}, \text{RefList}, \{\text{extern(Ext1)}\}, \text{ListOut}) : -
\]

\[
\text{resolve\_one}(\text{StN}, \text{RefList}, \text{Ext1}, \text{ListOut}), !.
\]

\[
\text{resolve}(\text{StN}, \text{RefList}, \{\text{extern(Ext1)}, \text{extern(Ext2)}\}, \text{ListOut}) : -
\]

\[
\text{resolve\_two}(\text{StN}, \text{RefList}, \text{Ext1}, \text{Ext2}, \text{ListOut}), !.
\]

In this case there are two pronominals to be solved and the discourse is in the state of shifting.

\[
\text{resolve\_two}(\text{StN}, \text{RefList}, \text{Ext1}, \text{Ext2}, \text{ListOut}) : -
\]

\[
\text{back\_state}(\text{StN}, \text{PrecSN}, \text{shifting}),
\]

\[
\text{find\_antecedents}(\text{StN}, \text{Ext1}, \text{Ext2}, \text{RefList}, \text{main-sec, sec-main}, \text{ListOut}), !.
\]
Find_Antecedents

The procedure find_Antecedents has two pronominals (Ext1, Ext2), to be matched which are recovered from the List of Referential expressions and are usually the highest candidates in the Weighted list of Potential Topics: antecedents are searched for in the previous portion of text according to their topicality: in this case, the two pronominals try a match, respectively, the first one with the previous Main Topic and the second one with the Secondary Topic and then they try the opposite.

\[
\text{find_antecedents}(\text{SentN}, \text{Ext1}, \text{Ext2}, \text{RefList}, \text{Types}, \text{ListOut}) : -
\]

\[
\begin{align*}
on(&\text{ref_ex}(\text{Ext1}, T, Ta, P, G, N, C, F/R, Px, \text{RefList}), \\
&\text{ref_ex}(\text{Ext2}, T1, Ta1, P1, G1, N1, C1, F1/R1, Py, \text{RefList}), \\
&\text{try_agreements}(\text{SentN}, \text{ref_ex}(\text{Ext1}, T, Ta, P, G, N, C, F/R, Px), \\
&\text{ref_ex}(\text{Ext2}, T1, Ta1, P1, G1, N1, C1, F1/R1, Py, \text{Types}, \text{Head1}, \text{Head2}), \\
&\text{append}(\text{est}(\text{Ext1}, \text{Head1}), \text{est}(\text{Ext2}, \text{Head2}), \text{RefList}, \text{L}), \\
&\text{modify_heads}(\text{L}, \text{ListOut}).
\end{align*}
\]

As shown in the previous section, arguments to ref_ex are the following:

\[\text{ref_ex}(\text{SyntacticIndex}, \text{LexicalHead}, \text{ReferentialTable}, \text{Person}, \text{Gender}, \text{Number}, \text{SemanticCategory}, \text{Function/Role}].\]

Try Agreement

\[
\text{try_agreements}(\text{SentN}, \text{Ref1}/\text{Px}, \text{Ref2}/\text{Py}, [\text{Type} | \text{OtherType}], \text{Head1}, \text{Head2}) : -
\]

\[
\begin{align*}
&\text{try_agrees}(\text{SentN}, \text{Ref1}/\text{Px}, \text{Ref2}/\text{Py}, [\text{Type}], \text{Head1}, \text{Head2}) ; \\
&\text{try_agrees}(\text{SentN}, \text{Ref1}/\text{Px}, \text{Ref2}/\text{Py}, \text{OtherType}, \text{Head1}, \text{Head2}).
\end{align*}
\]

\[
\text{try_agrees}(\text{SentN}, \text{Ref1}/\text{Px}, \text{Ref2}/\text{Py}, [\text{main-sec}], \text{Head1}, \text{Head2}) : -
\]

\[
\begin{align*}
&\text{main}(X), X!=\text{nil}, \\
&Px =< Py, \\
&\text{matcha}(\text{SentN}, \text{Ref1}, \text{main}, \text{Head1}), \\
&\text{matcha}(\text{SentN}, \text{Ref2}, \text{secondary}, \text{FrasePrec}, \text{Head2}).
\end{align*}
\]

where Px, Py are the score computed for that ref_ex as to its intrinsic referring weight.

5.2. Accomodation and Global Context: Coping with Bridging Phenomena

However, grammatical binding is not always possible, either because of some inconsistency due to inferential processes or because of domain restrictions. The second mode is then the one related to the presence of a pronoun and/or a definite NP which could modify the expectations built up by the general algorithm. In particular, in case there are two important characters, it is necessary to keep under control definite NPs in order to infer some general property related to one of the characters and then leave the pronoun for the usual
matching devices with the other character. In case no such inference takes place, inconsistencies may arise because it might be the case that the current Main Topic is not the adequate antecedent for pronominal reference. This information can only be captured once a definite NP is checked by inferential means to be a property belonging to the Main Topic. In this case, binding the pronoun and coreferring the definite NP to the same entity would result in a clash, which might be dubbed as a case of Obviation or of Disjoint Reference.

**Special Cases**

A special case is a search in the external knowledge of the world for a coreferring or cospecifying property which is then used to match with the main or the secondary topic of the previous portion of text. The relevant term is is_a(A,B) where A is a superset of B, the current property of the ref_ex.

```prolog
resolve_two(SentN,RefList,Ext1,Ext2,ListOut) :-
  back_state(SentN,PrecSentN,[continue, cont_analyze]),
  on(ref_ex(Ext1,Head,Tab,Pers,Gen,Num,Cat,F/R)/Px,RefList),
  on(ref_ex(Ext2,Head2,TabY,PersY,GenY,NumY,CatY,FY/RY)/Py,RefList),
  infer_match_global(main-sec, Head, RY, Ref1, SentN, PrecSentN, Head2, Head1),
  assert(external(Ext1)),
  append([est(Ext1,Head1),est(Ext2,Head2)],RefList,L),
  modify_heads(L,ListOut), !.
infer_match_global(main-sec, Pred, Ref1, N, N1, Head2, Head1) :-
  (main(ref_ex(_,Head2,_,_,_,_,_)),
   is_a(Head2,Pred),
   matcha(N,Ref1,main,N1,Head1), Head1 \= Head2 ;
  secondary(ref_ex(_,Head2,_,_,_,_,_)),
   is_a(Head2,Pred),
   matcha(N,Ref1,secondary,Head1), Head1 \= Head2).
```

This is another special case: the referential expression ranked higher in the Weighted List is a definite NP - a proper name or just a common noun - which does not match with any of the previous topics; its cospecifier is searched for in the Discourse Model - it might be a name or a role and its basic property is checked against the feature matrix of the RefExpr.

```prolog
resolve_two(SentN,RefList,Ext1,Ext2,ListOut) :-
  back_state(SentN,PrecSentN,[continue, cont_analyze]),
  on(ref_ex(Ext1,Head,Tab,Pers,Gen,Num,Cat,F/R)/Px,RefList),
  on(ref_ex(Ext2,Head2,TabY,PersY,GenY,NumY,CatY,FY/RY)/Py,RefList),
  fact(_,role,[Head,Id,Idx],1,Temp1,Loc1),
  fact(_,name, [Head1, Id], 1, Temp, Loc),
  fact(_,name, [Head2, Id2], 1, Temp, Loc),
  Head2 \= Head1, topic(_,secondary,Id2),
  (fact(_,isa,[ind:Id2,class:Cat2],1,Temp2,Loc2),
   match_gender(Cat2, Gen) ;
  fact(_,inst_of,[ind:Id2,class:Cat2],1,Temp3,Loc3),
   match_gender(Cat2, Gen)),
```
assert(external(Ext1)),
append([est(Ext1,Head1),est(Ext2,Head2)],RefList,L),
modify_heads(L,ListOut), !.

where Cat2 may assume default properties like man/woman.

5.3. Subject_of_Consciousness and Discourse Domains

Finally the last mode takes advantage of Discourse Domain: this notion is related to the
need of segmenting the text into Objective and Subjective Domains. This subdivision is taken
advantage of whenever a Subjective Domain is established and some character is assigned as
Subject of Consciousness(SOC). This usually happens whenever the character is already
asserted as the Main Topic of a given discourse segment. In case a SOC is present, the
weighted list of referring expressions for the current sentence is only used to introduce
Expected or Potential Topics: the Main Topic on the contrary is always fixed to the current
SOC. This is also applicable to cases of “psychological atmosphere statements” in which no
new entity is being introduced, but there is the need to maintain the SOC as Main Topic, so
that it would be available to any pronoun or definite NP intervening in the subsequent
stretches of text.

resolve(SentN,RefList,[external(Ext1),external(Ext2)],ListOut) :-
fetch_sc(SentN, Factive, Change, Rel, VerbCat, DiscDom, Head),
check_definit(Pronoun, RefList, F, R, Tab_ref, Gen, Ext1, Head1),
(main(ref_ex(_,Head,_,P,G,N,C,F1/R1))
->
Gen=G ;
fact(_,inst_of,find:Id,class:Cat],1,Temp1,Loc1),
match_gender(Cat, Gen)),
on(ref_ex(Ext2,Head2,Tab_ref1,Pers1,Gen1,Num1,Cat1,F1/R1)/_,RefList),
append([est(Ext1,Head-sc),est(Ext2,Head3-sc)],RefList,L3),
modify_heads(L3,ListOut), !.

where fetch_sc is a call that should recover the SOC if present, Gender is recovered by
the call check_definit in which the variable Pronoun may assume the possible values
short_anaphor, clitic and independent personal pronoun. Then a match is attempted with Main
Topic.

5.4. Discourse Level Definite NPs Resolution and Topics Hierarchy

Finally the topic schema is created where referential expressions which are possible
topics of discourse are assigned to a hierarchy according to their topichood, and a state is
associated to each sentence or utterance thus analyzed.

discourse(1,Args) :-
Arg\(s=\{\text{ref\_ex(Ext1,Head,\_,Pers,Gen,Num,Cat,F/R)/|}_\text{ListOthers}\),
assert(expected(ref\_ex(Ex1,Head,\_,Pers,Gen,Num,Cat,F/R))),
assert(main(nil)),
assert(secondary(nil)),
assert\_pot(ListOthers,1),
assert(state(1,change)).

General cases are dealt with by procedures like the following one where a match is searched for between the first two refexprs in the Weighted List and the previous main and potential topics which are then asserted respectively as main and secondary topic.

discourse(N,\[\text{Ref}1/_\_|\text{Others}\]) :-
create\_disc\_grid(N,Ref1, Others, r\_main, r\_pot, a\_main, a\_sec, continue),!.

Special Cases
At his level of analysis, in case no pronominal had been spotted and only NPs are present a hierarchy is created among them in order to ascertain which one of them is the most semantically relevant, from its grammatical features. This NP is then searched for in the previous topic schema, and if not present, it is searched for as a subject of consciousness, then as a property in the Discourse Model associated to an entity already present in the previous portion of text; finally the external world knowledge is searched for.

discourse(N,Args) :-
Args=\{\text{ref\_ex(SnX,HeadX,\_,PersX,GenX,NumX,CcX,FX/RX)/|}_\text{Others}\),
(expect(ref\_ex(SnP,HeadP,\_,Pers,Gen,Num,Cc,F/R));
main(ref\_ex(SnP,HeadP,\_,Pers,Gen,Num,Cc,F/R))),
HeadP=HeadX,
match\_sem\_num(TabX, NumX),
( secondary(ref\_ex(SnY,HeadY,TabY,PersY,GenY,NumY,CcY,FY/RY)),
control\_cat(CcY,CcX),
retract(secondary(_)), assert(secondary(ref\_ex(SnP,HeadP,\_,Pers,Gen,Num,Cc,/_))),
retract(main(_)), assert(main(ref\_ex(SnX,HeadX,\_,PersX,GenX,NumX,CcX,FX/RX))),
retract(expect(_)), assert(expect(nil))),
assert(state(N,resume)),
assert\_pot(Others,N), !.

where the call match\_semantic\_number verifies whether the current ref\_ex is a definite NP and is plural or singular or it is a proper name, by calling a fact in the discourse model with the lexical head of the ref\_ex and checking its cardinality or its semantic status of a set or of an individual.

6. The Stories and Discourse Anaphora
The following representations are the output of the system and are organized as follows: they contain a list of discourse topics candidates ranked according to their grammatical relevance, in reverse order - the smaller the value of the score the higher the rank, the score being attached at the end of each information vector. At the end of the list, we find the TOPIC structure, or the association between each Topic type and candidate ref(erring)_ex(pression), except for Potential Topics which receives a list. Then, the present state of the algorithm. This representation is used to compute relevance scores associated to each discourse topic. Here is the output from the story of 'The Three Little Pigs'.

6.1. Story 1

1. [ci erano una volta tre fratelli porcellini che vivevano felici nella campagna]/There were one time three piglets who lived happy_plur/mas in the countryside
WEIGHTED LIST OF TOPICS :
ref_ex(sn9, porcellino, [+ ref, 0def, + part, 3, - pro, - ana, + class], 3, m, p, [edible, anim], subj_foc/th_unaff)/3
ref_ex(sn26, campagna, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, f, s, [place], obl/locat)/58
TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn9, porcellino, _, 3, m, p, [edible, anim], subj_foc/th_unaff)
POTENTIAL TOPICS : ref_ex(sn26, campagna, _, 3, f, s, [place], obl/locatrelat)
state(1, shifting)

Sentence one sets the algorithm in the state of SHIFTING establishing the Main Topic owing to the Presentative structural configuration marked out by the presence of a Focussed Subject. However, the second sentence also contains a Focussed Subject in a configuration where locative inversion has been used: thus, the algorithm goes again to shifting establishing a new Main Topic, and turning the previous Main Topic into a Secondary Topic.

2.[nello stesso luogo però viveva anche un terribile lupo che si nutriva proprio di porcellini grassi]
WEIGHTED LIST OF TOPICS :
ref_ex(sn7, lupo, [+ ref, - def, nil, nil, - pro, - ana, + class], 3, m, s, [feroc, anim], subj_top/actor)/1
ref_ex(sn1, luogo, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [place], obl/locat)/28
TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn7, lupo, _, 3, m, s, [feroc, anim], _/)
SECONDARY TOPIC : ref_ex(sn9, porcellino, _, 3, m, p, [edible, anim], _/)
POTENTIAL TOPICS : ref_ex(sn1, luogo, _, 3, m, s, [place], obl/locat)
state(2, shifting)

Sentence three reintroduces both participants in the story with different roles and in different ways: the little pigs are reintroduced as SUBJect, thus receiving a high score, by means of a demonstrative pronoun; on the contrary, the wolf is mentioned in an adjunct clause and receives a low score. As all deictic pronouns, the demonstrative is computed both as a pronoun and as a nominal head, thus receiving the feature +class. In this case all functional features match. The state of discourse is now retaining: notice that we have both an expected and a main topic.

3.[questi allora, per proteggersi dal lupo, decisero di costruirsi ciascuno una casetta]
WEIGHTED LIST OF TOPICS :
ref_ex(sn2, questi, [+ ref, + def, - part, nil, + pro, - ana, + class], 3, m, p, [anim, hum], subj/actor)/11
In this and the following sentence, there is a number mismatch between the quantifier “ciascuno”–each, the nominal substitute ‘the oldest one’ and the antecedent, the set of three little pigs, only pointing to a subset made up of a singleton. In order to compute this difference, we simply neutralized the feature Number and let all other features match as before. In addition, since the individual being referred is strictly speaking contained as a member in the set constituted by the antecedent and is the Main Topic, we use continue_analyse. Neutralizing is made possible by information made available by the algorithm of Quantifier Raising that has raised the distributional quantifier “ciascuno” – see chapter on Quantifiers.

4. [Il maggiore, Jimmi, che era saggio, lavorava di buona lena e costruì la sua casetta con solidi mattoni e cemento]

WEIGHTED LIST OF TOPICS:

ref_ex(sn1, maggiore, [+ ref, + def, - part, nil, + pro, - ana, + class], 3, m, s, [obj, anim, hum], subj/agent)/30
ref_ex(sn26, casa, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, f, s, [obj, place], obj/th_eff)/35

TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(_ , porcellino, _ , 3, m, p, [hum, obj, anim], subj/agent)
POTENTIAL TOPICS : ref_ex(sn26, casa, _ , 3, f, s, [obj, place], obj/th_eff)

state(4, continue_analyse)

In this case, the nominal substitute 'other ones' has a plural Number but the same set is coreferred, extracting though a different subset from the previous one. We used the same strategy of Neutralization in order to let the Main Topic continue, and the state is continue_analyse as before:

5. [gli altri, Timmy e Tommy, pigri e oziosi se la sbrigarono in fretta, costruendo le loro casette con la paglia e con pezzetti di legno]

WEIGHTED LIST OF TOPICS:

ref_ex(sn1, altri, [+ ref, + def, - part, nil, + pro, - ana, + class], 3, m, p, [hum, obj, anim], subj/agent)/10
ref_ex(sn42, casa, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, f, p, [obj, place], obj/th_eff)/35

TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn1, porcellino, _ , 3, m, p, [hum, obj, anim], subj/agent)

state(5, cont_analyse)

In the following utterance, the discourse state is Continue: the set of two little pigs has now steadily become the Main Topic. The big bad wolf is always in the background: in this case it is the subject topic of a song.

6. [i due porcellini pigri passavano le loro giornate suonando e cantando una canzone che diceva chi ha paura del lupo cattivo]

WEIGHTED LIST OF TOPICS:
In the following utterance the wolf reappears as Main Topic: we see here that linguistic form is paramount in deciding the role participants in a discourse or text should assume: presentational structure usually has an inverted subject NP which is so computed by grammatical rules. In our case, the presentational nature of the utterance is computed by the presence of "ecco che improvvisamente".

7.

WEIGHTED LIST OF TOPICS:

TOPICS HIERARCHY:

MAIN TOPIC : ref_ex(sn1, lupo, _ , 3, m, s, [feroc, anim], obl/malef)

SECONDARY TOPIC : ref_ex(sn7, loro, _ , 3, n, [feroc, anim], subj/poss)

state(7, retaining)

In the following utterance the scenario changes again and the little pigs are the focus of discourse, while the wolf is asserted as secondary topic,

8.

WEIGHTED LIST OF TOPICS:

TOPICS HIERARCHY:

EXPECTED TOPIC : ref_ex(sn7, porcellino, _ , 3, m, p, [edible, anim], subj/agent)

SECONDARY TOPIC : ref_ex(sn10, pasto, _ , 3, m, s, [edible, event, obj], obl/theme)

state(8, continue)

In the following utterance attention is again shifted to the wolf which becomes the expected topic; it is interesting to note that in order to do that a deictic demonstrative pronoun is used, so that the secondary topic might be easily addressed,

9.

WEIGHTED LIST OF TOPICS:

TOPICS HIERARCHY:

EXPECTED TOPIC : ref_ex(sn2, questo, _ , 3, m, s, [anim, hum], subj/agent)

MAIN TOPIC : ref_ex(sn10, porcellino, _ , 3, m, p, [edible, anim], subj/agent)

POTENTIAL TOPICS : ref_ex(sn10, pasto, _ , 3, m, s, [edible, event, obj], obl/theme)

state(9, retaining)
However, in the following utterance the wolf momentarily disappears from the text and the attention is completely devoted to the two little pigs and their house, which will become a single individual entity. Notice that the same textual strategy we saw before is at stake here: there is a Main Topic and an Expected Topic in utterance 7, which are turned into a Main Topic and a Secondary Topic in the following utterance. The central character around which the story is now revolving, are the two lazy little pigs, and the wolf appears and disappears.

10. [finalmente i porcellini riuscirono a raggiungere la loro casetta e vi si chiusero dentro sbarrando la porta]

CONTINUE is also the state in the following utterance which has the same organization of the previous one, However in the next utterance the wolf takes on a new role: from Expected Topic it will become Main Topic. Also notice what happens to the little house: it appears as Potential Topic, then it is reinforced in the following utterance, no.13, and again in the following utterance, no.14. The result of this text strategy, is that the main character, the wolf, is the Main Topic and the little house has a secondary role, thus becoming Secondary Topic. The two little pigs, however, disappear from the current scenario.

12. [il lupo stava intanto pensando al modo di penetrare nella casa]

Another interesting case of Neutralization is constituted by the use of Direct Speech and first and second person personal forms. This happens suddenly in the text: the algorithm should behave in such a way as to let the Main Topic and or Secondary Topic to continue, and this is what it does, by neutralizing the feature Person and letting the remaining features match.

15. [spaventatissimi i due porcellini corsero a perdifiato verso la casetta del fratello]
EXPECTED TOPIC : ref_ex(sn1, porcellino, _, 3, m, p, [edible, anim], subj/agent)
SECONDARY TOPIC : ref_ex(sn3, lupo, _, 3, s, [anim, hum], subj/agent)
POTENTIAL TOPICS : ref_ex(sn69, fratello, _, 3, m, s, [hum, anim, relat], subj/poss)

state(15, resume)

16. ['presto, fratellino, aprici! Abbiamo il lupo alle calcagna'] / Quick, brother + little, open_imper/ 2nd_pers+ us_plur! (We) pro have_present_plur/1st_pers t the wolf at our heels.
WEIGHTED LIST OF TOPICS :
ref_ex(sn17, pro, [+ ref, + def, nil, nil, + pro, - ana, - me], 1, _, p, [hum, anim], subj/exper)/ -7
ref_ex(sn18, lupo, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [feroc, anim], obj/th_bound)/26
MAIN TOPIC : ref_ex(sn17, porcellino, _, _, _, p, [hum, anim], subj/exper)
SECONDARY TOPIC : ref_ex(sn18, lupo, _, 3, m, s, [feroc, anim], obj/th_bound)

state(16, continue)

As can be noticed, personal forms should be made visible to the sentence binding module so that the clitic 'ci/us' could be adequately bound by the little pro 1st person pronoun of the following sentence before reaching the discourse module. Also, the lexically bound subject of the imperative is anaphorically bound to the vocative 'fratellino' at sentence level. In the following sentence, 3rd person plural verbal agreement is used to continue the story.

17. [fecero appena in tempo ad entrare e a tirare il chiavistello]
WEIGHTED LIST OF TOPICS :
ref_ex(sn3, pro, [+ ref, + def, nil, nil, + pro, - ana, - me], 3, _, p, [hum, anim], subj/agent)/ -20
ref_ex(sn31, chiavistello, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [obj, instr], obj/th_aff)/135
TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn3, porcellino, _, 3, _, p, [hum, anim], subj/agent)
SECONDARY TOPIC : ref_ex(sn18, lupo, _, 3, m, s, [feroc, anim], obj/th_bound)
POTENTIAL TOPICS : ref_ex(sn31, chiavistello, _, 3, m, s, [obj, instr], obj/th_aff)

state(17, continue)

An interesting interchange of topicality between the participants of the story can be found in the remaining part of the text which we report below. Utterance 18 reintroduces the wolf as Expected Topic, and the state becomes RETAINING with the little pigs still as main topic:

18. [il lupo stava già arrivando deciso a non rinunciare al suo pranzetto]
WEIGHTED LIST OF TOPICS :
ref_ex(sn1, lupo, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [feroc, anim], subj/th_aff)/13
ref_ex(sn10, pranzetto, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [edible, event, obj], obl/th_aff)/53
TOPICS HIERARCHY:
EXPECTED TOPIC : ref_ex(sn1, lupo, _, _, m, s, _/_) 
MAIN TOPIC : ref_ex(sn3, porcellino, _, 3, _, p, [hum, anim], subj/agent)
POTENTIAL TOPICS : ref_ex(sn10, pranzetto, _, 3, m, s, [edible, event, obj], obl/th_aff)

state(18, retaining)

The state is now turned into Continue due to the fact that the utterance refers to the wolf again, while the little pigs become secondary topic:

19. [sicuro di abbattere anche la casetta di mattoni il lupo si riempì i polmoni di aria e cominciò a soffiare con forza alcune volte]
WEIGHTED LIST OF TOPICS :
At this point however, an impersonal statement is made "there was nothing to be done", and even though no particular antecedent is actually meant, the discourse is taken to continue with the same structure of the previous state, and the big PRO subject of "to do" is bound to the wolf, which is computed as Main Topic and also the Subject of Consciousness,

20. [non ci era niente da fare]
WEIGHTED LIST OF TOPICS:
ref_ex(sn38, pPro, [+ ref, nil, nil, + pro, + ana, - me], _, _, _, [hum, anim], subj/agent)/10
ref_ex(sn5, niente, [+ ref, + def, - part, nil, - pro, - ana, + me], 3, _, s, [obj, neg], subj/th_unaff)/1013
TOPICS HIERARCHY:
MAIN TOPIC: ref_ex(sn38, lupo, _, _, _, [hum, anim], subj/agent)
SECONDARY TOPIC: ref_ex(sn5, niente, _, 3, _, s, [obj, neg], subj/th_unaff)
state(20, continue)

A new expected topic is now reintroduced, the "house" the wolf was trying to blow down a few utterances before: this topic is no longer available in the topic stacks of the previous sentence, so it is resumed from the discourse model and will have to be reinforced in the following portion of text in case it has to become a main topic; the wolf, which was main topic in the Continue state of the previous utterance, is now turned into secondary - seen that there are more expectations about the newly resumed topic:

21. [la casa non si mosse di un solo palmo]
WEIGHTED LIST OF TOPICS:
ref_ex(sn1, casa, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, f, s, [obj, place], subj/th_aff)/13
ref_ex(sn5, palmo, [+ ref, - def, nil, nil, - pro, - ana, + class], 3, m, s, [meas, place, obj], obj/meas)/62
TOPICS HIERARCHY:
EXPECTED TOPIC: ref_ex(sn1, casa, _, 3, f, s, [obj, place], subj/th_aff)
SECONDARY TOPIC: ref_ex(sn38, lupo, _, _, _, [hum, anim], subj/agent)
POTENTIAL TOPICS: ref_ex(sn5, niente, _, 3, _, s, [obj, neg], subj/th_unaff), ref_ex(sn5, palmo, _, 3, m, s, [meas, place, obj], obj/meas)/62
state(21, resume)

However, no such thing happens and the text goes back again talking about the wolf, previously turned into a secondary topic, and now a main topic again,

22. [alla fine esausto il lupo si accasciò a terra]
WEIGHTED LIST OF TOPICS:
ref_ex(sn4, lupo, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [feroc, anim], subj/exper)/13
ref_ex(sn11, terra, [+ ref, 0def, nil, nil, - pro, - ana, + class], 3, f, s, [place], obj/locat)/30
TOPICS HIERARCHY:
EXPECTED TOPIC: ref_ex(sn4, lupo, _, 3, m, s, [feroc, anim], _/_)
SECONDARY TOPIC: ref_ex(sn1, casa, _, 3, f, s, [obj, place], _/_)
POTENTIAL TOPICS: ref_ex(sn11, terra, _, 3, f, s, [place], obj/locat)
At this point of the story, the three little pigs have to be reintroduced in the discourse as expected topics and are resumed from the discourse model. They couldn't possibly be coreferred by a pronoun, because they are now too far away in the previous portion of the text. The little house is present again and so it becomes main topic.

Finally, the last utterance of the text, talks about little pigs as the previous utterance, but only about two of them: the discourse module takes notice of that fact by detecting a definite NP which has cardinality spelt out and corefers to a previous definite NP which also had cardinality spelt out in its specifier. Every time a Main Topic is asserted by means of a NP which has an explicit cardinality indication different from the one expressed in the previous utterance associated to the same head noun, the state the algorithm will choose is continue_analyse.

6.2. Story 2

We shall now discuss the second version of the story of the three little pigs, which has a somewhat different – but very intriguing – structure from the first abridged version. After the little pigs have been introduced as a set and names have been assigned to them, we have the following descriptions:

4. [a timmy non piaceva per niente lavorare così pensò di costruirsi rapidamente una capanna di paglia] WIGHTED LIST OF TOPICS :
ref_ex(sn2, timmy, [+ ref, 0def, nil, nil, - pro, - ana, - class], 3, m, s, [hum, anim], obj2/exper) -10
ref_ex(sn45, capanna, [+ ref, - def, nil, nil, - pro, - ana, + class], 3, f, s, [obj], obj/th_eff) 155
TOPICS HIERARCHY:
EXPECTED TOPIC : ref_ex(sn2, timmy, _ , 3, m, s, [hum, anim], obj2/exper)
At this point of the story we have the set of three little pigs, and now computed as secondary topic, then Timmy, one of the three little pigs, as expected topic and 'capanna'/hut, as potential topic.

As can be noticed from this computations, a number of different things have happened: first of all, Timmy has been turned from expected to main topic; then, the "capanna"/hut has now been coreferred as "casetta"/little house, by an inference that allows huts to be similar enough to little houses. Now consider, the NP "i suoi fratellini": the system understands this NP to be coreferred to the little pigs "i porcellini" of the previous portion of text. Now that this relation has been established by the Semantic Module, the subsequent move is to find a set where a simple property - i.e. not a role or a name, for instance - is asserted of Timmy, and this is then checked against the current topic stack in order to see whether "fratellini" could be taken to corefer to it. This is what happens, and "fratellini" is taken to be coreferential with "porcellini", previously computed as secondary topic. The story continues by introducing a new topic as proper name Object.

The discourse state is now Continue and a new possible topic has been introduced as secondary topic; however, the main topic is continued and in order to shift the topic of discourse from the main to the secondary a special construction is needed. This is what happens in the following utterance,
TOPICS HIERARCHY:

EXPECTED TOPIC : ref_ex(sn3, tommy, __, 3, m, s, [hum, anim], __/__)
MAIN TOPIC : ref_ex(sn3, timmy, __, 3, __, s, [hum], __/__)
POTENTIAL TOPICS : ref_ex(sn28, casa, __, 3, f, s, [obj, place], obj/th_eff)

state(7, retaining)

The Subject of the main clause is the independent pronoun "lui" which can be used to contrast or emphasize a new topic: in that case, a focalizer should be present in the structure. And we see that "anche"/also, too, a typical focalizer is used in order to set the pronoun in focus. In case no such focalizer were present, or another independent pronoun were used, the reference to the secondary topic would have been impossible to attain.

8. [ben presto anche la casa di legno fu pronta]
WEIGHTED LIST OF TOPICS :
ref_ex(sn2, casa, [+ ref, + def, nil, nil, - pro, - ana, + clas], 3, f, s, [obj, place], subj/th_bound)/14
TOPICS HIERARCHY:
EXPECTED TOPIC : ref_ex(sn2, casa, __, 3, f, s, [obj, place], subj/th_bound)
SECONDARY TOPIC : ref_ex(sn3, tommy, __, 3, m, s, [hum, anim], __/__)
state(8, retaining)

In the previous utterance, the little house has become the new Expected Topic of discourse and the little pig Tommy has been demoted to Secondary Topic. In this utterance it has become Main Topic, but in the following text it will disappear. Notice that in fact there are two houses here being coreferred to: the current one being built by Tommy, and the one Timmy already built: i.e. the deictic demonstrative pronoun is one house and the little pro is another.

9. [come quella di paglia, non era certo molto resistente]
WEIGHTED LIST OF TOPICS :
ref_ex(sn2, quella, [+ ref, + def, - part, nil, + pro, - ana, + clas], 3, f, s, [obj], subj/compar)/80
ref_ex(sn15, pro, [+ ref, + def, nil, nil, + pro, - ana, - me], 3, __, s, [obj], subj/th_bound)/114
TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn2, casa, __, 3, f, s, [obj, place], sub/compar)
POTENTIAL TOPICS : ref_ex(sn3, tommy, __, 3, m, s, [hum, anim], __/__)
ref_ex(sn15, casa, __, 3, s, [obj], subj/th_bound)
state(9, cont_analyse)

The little house now being referred to is main topic, and the little house previously referred to by the comparative is recovered here as secondary topic through the reference to "paglia"/straw. In the following text, the two lazy little pigs reappear as SUBJect of the main clause and are introduced as Expected Topic, the little house being demoted to Secondary Topic.

10. [ma i due porcellini scansafatiche se la erano sbrigata in poco tempo ed ora potevano tranquillamente divertirsi]
WEIGHTED LIST OF TOPICS :
ref_ex(sn1, porcellino, [+ ref, + def, nil, due, - pro, - ana, + clas], 3, m, p, [edible, anim], subj/agent)/40
TOPICS HIERARCHY:
EXPECTED TOPIC : ref_ex(sn1, porcellino, __, 3, m, p, [edible, anim], subj/agent)
As can be noticed, the system now has a new set which however at the level of grammatical binding is absolutely identical to the previous one, being denoted by the same property. Not so in the DM where the two lazy little pigs, are understood as a subset of the previous set of three little pigs. The secondary topic is the most recent little house being talked about in the previous text.

This utterance contains a pronominal subject which takes splits antecedents as its binder in the anaphoric binding module. Being bound at the lower utterance level, this pronoun does not pop up in the discourse module, directly. But the text is built in such a way that its presence must be recovered indirectly by the semantic module. In the previous text, we had the set of two little pigs as expected topic: however, in the current utterance we only have two proper names, which in turn have been collectively taken as split antecedents of a plural pronominal subject. The discourse module tries to match the proper nouns with the previous main or expected topic: but the two higher ref_ex candidates do not match in number with the expected topic, nor in semantic features with the main topic of the previous discourse. Thus, in this discourse configuration, i.e. whenever a proper noun is the best candidate for topichood, and the possible coreferent NP is a common noun, the model is searched for. The Discourse module looks for facts relating the name to properties of the individual in order to see whether the previous main or expected topic is a property associated to the current more likely candidates, i.e. the ref_ex ranking higher in the argument of discourse entry list. And it finds that the names are associated to little pigs: however, the model contains two sets, one with three members and another one, more recent, with only two members. In order to pick up the expected topic as coreferent of the current set of names, then, we check whether the names constitute and adequate and semantically compatible set of two members.
At this point of the story we know that there is a set of two little pigs which is Main Topic of discourse, and we know their names. Now consider the reference to "fratellino"/little brother which becomes Secondary Topic. The following text cospecifies it with its name, Jimmy, still a Secondary Topic. The state of the algorithm is however, Retaining, thus indicating that there can be the possibility for the story to turn to Continue by assuming the secondary topic as new main topic of discourse.

13. [si misero in cammino e ben presto raggiunsero jimmy]

WEIGHTED LIST OF TOPICS:
ref_ex(sn3, pro, [+ ref, + def, nil, nil, + pro, - ana, - me], 3, _, p, [anim, hum], subj/agent)/0
ref_ex(sn17, jimmy, [+ ref, 0def, nil, nil, - pro, - ana, - class], 3, m, s, [hum, anim], obj/goal)/13

TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn3, porcellino, _ , 3, _, p, [anim, hum], subj/agent)
SECONDARY TOPIC : ref_ex(sn17, jimmy, _ , 3, m, s, [hum, anim], obj/goal)
state(13, retaining)

Thus the story continues by introducing a coreferential common noun as SUBJECT of the main clause. In order to capture the referential link of the singular definite NP, we proceed as before by searching the model in order to find a property associated to Jimmy that matches the one being put forward by the best candidate in the rank list.

14. [il bravo porcellino stava costruendo anche lui la sua casetta]

WEIGHTED LIST OF TOPICS:
ref_ex(sn1, porcellino, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [edible, anim], subj/agent)/10
ref_ex(sn6, casa, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, f, s, [obj, place], obj/th_eff)/15
MAIN TOPIC : ref_ex(sn1, porcellino, _ , 3, m, s, [edible, anim], subj/agent)
SECONDARY TOPIC : ref_ex(sn6, casa, _ , 3, f, s, [obj, place], obj/th_eff)
state(14, continue)

In this sentence both the little house and the little are coreferred by pronouns and require an appropriate matching which is done by agreement checks.

15. [ma poiché era previdente e non aveva paura di lavorare sodo, la costruiva con mattoni e cemento]

WEIGHTED LIST OF TOPICS:
ref_ex(sn255, pro, [+ ref, + def, nil, nil, + pro, - ana, - me], 3, _, s, [anim], subj/agent)/-10
ref_ex(sn256, la, [+ ref, + def, nil, nil, + pro, + ana, + me], 3, f, s, [obj], obj/th_eff)/-5
MAIN TOPIC : ref_ex(sn255, porcellino, _ , 3, _, s, [anim, hum], subj/agent)
SECONDARY TOPIC : ref_ex(sn256, casa, _ , 3, f, s, [obj, place], obj/th_eff)
state(15, continue)

6.3. Story 3

We shall now discuss a text we have analysed, which, differently from the examples found in the literature is taken directly from a newspaper. It deals with politics and there are three main topics: Avveduti who has been appointed secretary general by Antonio Alberti, his father-in-law who is a senator, and Trabucchi, a minister of trade. In the first section of the text, Franco Avveduti is presented on the scene and his properties asserted. So we learn that
he was an "immigrato"/immigrant, and a beaurocrat; that he enrolled in military academy and that he was a good cadet. Since the text reinforces the topic by means of morphologically unexpressed pronouns, there are no variations in the topics hierarchy: Avveduti persists as Main Topic. This takes us to utterance No.5 where he is asserted as becoming a good officer. This property is important in the follow up, where Avveduti is no longer the only topic on the scene.

[fino a tre anni fa franco avveduti non si era mai immischiato col mondo della pubblica amministrazione.]

WEIGHTED LIST OF TOPICS:
ref_ex(sn6, avveduti, [+ ref, 0def, nil, nil, - pro, - ana, - class], 3, m, s, [hum], subj/ag)/10
ref_ex(sn12, mondo, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [place, instit, activ], obl/comit)/30

TOPICS HIERARCHY:
EXPECTED TOPIC : ref_ex(sn6, avveduti, __, 3, m, s, [hum], subj/agent)
POTENTIAL TOPICS : ref_ex(sn12, mondo, __, 3, m, s, [place, instit, activ], obl/comit)

stato(1, change)

[pìù tardi fu un ufficiale di,successo]

WEIGHTED LIST OF TOPICS:
ref_ex(sn1, pro, [+ ref, + def, nil, nil, + pro, - ana, - me], 3, __, s, [ind, real, mil, role], subj/th_bound)/16
ref_ex(sn4, ufficiale, [+ ref, - def, nil, nil, - pro, - ana, + class], 3, m, s, [ind, real, mil, role], ncomp/prop)/70

TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn1, avveduti, __, 3, m, s, [hum], subj/agent)
SECONDARY TOPIC : ref_ex(sn4, ufficiale, __, 3, m, s, [ind, real, mil, role], ncomp/prop)
state(5, continue)

[poi nel 1945 avveduti si dimise dallo esercito.]

WEIGHTED LIST OF TOPICS:
ref_ex(sn6, avveduti, [+ ref, 0def, nil, nil, - pro, - ana, - class], 3, m, s, [hum], subj/actor)/11
ref_ex(sn11, esercito, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [coll, real, mil, hum_set], obl/locat)/48

TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn6, avveduti, __, 3, m, s, [hum], __)
SECONDARY TOPIC : ref_ex(sn11, esercito, __, 3, m, s, [coll, real, mil, hum_set], obl/locat)
state(6, continue)

[i militari lo avevano deluso.]

WEIGHTED LIST OF TOPICS:
ref_ex(sn6, lo, [+ ref, + def, nil, nil, + pro, + ana, + me], 3, m, s, [hum], obj/exper)/15
ref_ex(sn2, militare, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, p, [ind, real, mil, role], subj/causer_emot)/110

TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn6, avveduti, __, 3, m, s, [hum], __)
SECONDARY TOPIC : ref_ex(sn2, militare, __, 3, m, p, [ind, real, mil, role], subj/causer_emot)
state(7, continue)

Then we learn that he graduated and became a lawyer and a solicitor. We have now reached utterance No. 10 where a new character is being introduced and will persist on the
scene for a while. Avveduti gets married and his father-in-law is presented on the scene, even though in a rather complex way. In the following three utterances the FSDA passes from a configuration in which Avveduti is the only Topic into a new configuration where Avveduti is Secondary and Alberti Main Topic, in utterance No. 12. To achieve this result, the text does no longer mention Avveduti and the object clitic pronoun used in utterance 14 reinforces the new Main Topic.

[intanto a verona aveva conosciuto paola, figlia di antonio alberti, potente senatore democristiano, e la aveva sposata.]

WEIGHTED LIST OF TOPICS:
ref_ex(sn5, pro, [+ ref, + def, nil, nil, + pro, - ana, - me], 3, s, [hum], subj/exper)/3
ref_ex(sn7, paola, [+ ref, 0def, nil, nil, - pro, - ana, - class], 3, f, s, [ind, real, nat, hum], obj/th_emot)/15

TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn5, avveduti, _, 3, s, [hum], _)
SECONDARY TOPIC : ref_ex(sn7, paola, _, 3, f, s, [ind, real, nat, hum], obj/th_emot)

state(10, continue)

[il senatore poteva considerarsi il più influente uomo politico veronese.]

WEIGHTED LIST OF TOPICS:
ref_ex(sn1, senatore, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [hum, social], subj/exper)/13
ref_ex(sn7, uomo, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [ind, real, nat, hum], ncomp/prop)/70

TOPICS HIERARCHY:
EXPECTED TOPIC : ref_ex(sn1, senatore, _, 3, m, s, [hum, social], subj/exper)
MAIN TOPIC : ref_ex(sn5, avveduti, _, 3, s, _, _)
POTENTIAL TOPICS : ref_ex(sn7, uomo, _, 3, m, s, [ind, real, nat, hum], ncomp/prop), ref_ex(sn7, paola, _, 3, f, s, [ind, real, nat, hum], obj/th_emot)

state(11, retaining)

[gli elettori lo mandavano al parlamento coprendolo di voti preferenziali.]

WEIGHTED LIST OF TOPICS:
ref_ex(sn9, lo, [+ ref, + def, nil, nil, + pro, + ana, + me], 3, m, s, [hum], obj/patient)/-16
ref_ex(sn5, elettore, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, p, [hum, abstr, social, role], subj/agent)/10

TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn9, senatore, _, 3, m, s, [hum], obj/patient)
SECONDARY TOPIC : ref_ex(_, avveduti, _, 3, s, _, _)
POTENTIAL TOPICS : ref_ex(sn5, elettore, _, 3, m, p, [hum, abstr, social, role], subj/agent)
ref_ex(sn11, parlamento, _, 3, m, s, [coll, real, soc, insti], obl/goal)

state(12, continue)

The text now reintroduces Avveduti and in order to do this both name and surname are used. A pronoun would not recover the appropriate antecedent. Also notice that Alberti, who has been referred to as "senatore"/senator also requires the name to be reproposed as secondary topic, though. Utterance 14 introduces a new property "suocero"/father-in-law which has never been expressed explicitly and must inferred from the DM. However, the two topics

[al seguito di alberti, che era diventato vicepresidente del senato, franco avveduti nello immediato dopoguerra si trasferi a roma.]

WEIGHTED LIST OF TOPICS:
Another interesting segment of text follows in which the activities that Avveduti carries out in his new position are described. This starts from utterance 16 onward: notice that "posto"/position is bound in the DM to "delegato"/delegate and to "compito"/task in force of the fact that all three predicates have been classified as "activity" in their inherent features list. However this semantic inferential process does not appear at Discourse level and has no effect to the description of Topics hierarchy. In fact, we are here left with Avveduti: and when the suocero reappears in utterance No.19 it is simply reasserted as secondary topic.

[solo verso il 1950 decise di accettare un posto nella organizzazione della fiera di verona.] WEIGHTED LIST OF TOPICS :
  ref_ex(sn38, pro, [+ ref, + def, nil, nil, + pro, - ana, - me], 3, _, p, [hum], subj/agent)/20
  ref_ex(sn55, posto, [+ ref, - def, nil, nil, - pro, - ana, + class], 3, m, s, [activ, place, state], obj/th_nonaff)/5
TOPICS HIERARCHY:
  MAIN TOPIC : ref_ex(sn38, avveduti, _, 3, m, s, [hum], subj/actor)
  SECONDARY TOPIC : ref_ex(sn55, posto, _, 3, m, s, [activ, place, state], obj/th_nonaff)
state(16, continue)

[lo nominarono delegato cioè una specie di funzionario viaggiante con incarichi diplomatici di tenere i rapporti con le delegazioni commerciali, curare i produttori stranieri, le grandi ditte, la stampa.] WEIGHTED LIST OF TOPICS :
  ref_ex(sn1, pro, [+ ref, + def, nil, nil, + pro, - ana, - me], 3, _, p, [hum], subj/agent)/20
  ref_ex(sn2, lo, [+ ref, + def, nil, nil, + pro, + ana, + me], 3, m, s, [hum, social], obj/th_bound)/20
TOPICS HIERARCHY:
  EXPECTED TOPIC : ref_ex(sn1, organizzazione, _, 3, _, p, [hum], subj/agent)
  MAIN TOPIC : ref_ex(sn2, avveduti, _, 3, m, s, [hum, social], obj/th_bound)
state(17, retaining)

[questo era un compito che corrispondeva bene alla sua vocazione e nel quale avveduti sapeva giostrare con notevole agilità.] WEIGHTED LIST OF TOPICS :
  ref_ex(sn19, avveduti, [+ ref, 0def, nil, nil, - pro, - ana, - class], 3, m, s, [hum], subj/actor)/21
  ref_ex(sn2, questo, [+ ref, + def, nil, nil, + pro, - ana, + class], 3, m, s, [activ], obj/th_bound)/114
TOPICS HIERARCHY:
  MAIN TOPIC : ref_ex(sn19, avveduti, _, 3, m, s, [hum], _)
state(18, continue)
[quando il suocero morì, egli non perse il posto.]

Utterance No.20 delimits another main segment of text since from here onward a new character appears on the scene: Trabucchi, the new political figure who inherits Alberti's, Avveduti's father-in-law, political heritage. The remaining part of the text sees the interchange of these two characters in their hierarchical roles as topics and the disappearance of Alberti due to his death.

[a verona il collegio di alberti lo aveva ereditato trabucchi e col collegio aveva ereditato la presidenza della fiera.]

At this point the text has established a hierarchy in which Trabucchi is Main and Avveduti is secondary Topic. The following utterances respect this hierarchy: however, in order to allow the reader a clear understanding of what are the correct antecedents of definite expressions and pronominals, reference is made to properties which help to clarify the hierarchy. So even though the following utterance has only one pronoun, utterance No.23 has two and there are two antecedents to be coreferred to. We assume then that in order to reduce access to the DM and to its knowledge representation the referring expressions should made clear what they are actually coreferring to.

[lo ex ufficiale del novara_cavalleria gli era simpatico.]
Here abruptly we have a change of perspective and Avveduti becomes the new focus of attention. In order to cause such a change, a pronoun is insufficient and the name must be used. As to the other main character, Trabucchi, even though he is no longer coreferred in the current text, he persists in the background as secondary topic: this is not compulsory and we have seen above that in order to become secondary topic a referring expression must be very important, both semantically and pragmatically.

The last utterance is remarkable because it reintroduces reference to Trabucchi by means of a pronoun: this is done both grammatically and semantically. There is a disjoint NP in the
same utterance, "segretario", which is a property just been associated to Avveduti, the current Main Topic. Thus, the system understands that the pronoun cannot possibly be bound by the same entity. The secondary Topic is checked and this appears to agree with "gli" a singular masculine dative clitic. In fact, one might compute the two previous utterance which determined the alternation of Secondary and Main as Trabucchi's afterthought. The question is that it is not apparent while reading them, even though they might be so interpreted when we get to the last utterance.

[tutti, gli, invidiavano, il, suo, segretario, particolare, .]

WEIGHTED LIST OF TOPICS:
ref_ex(sn5, gli, [+ ref, + def, nil, nil, + pro, + ana, + me], 3, m, s, [hum], obj2/act)/ -18
ref_ex(sn6, segretario, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [hum, social], obj/th_nonaff)/15

TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn5, trabucchi, _, 3, m, s, [hum, anim], obj2/actor)
SECONDARY TOPIC : ref_ex(sn6, avveduti, _, 3, m, s, [hum, social], obj/th_nonaff)

state(27, retaining)

6.4. Psychological Texts

We will now turn to another set of examples, the seven texts called Psychological Statements texts in the Introduction. As can be easily noticed, we made permutations of the central utterances while keeping fixed the first and last two utterances. In this way we can control the behaviour of our Finite State Discourse Automaton(FSDA).

In the first text, the one taken directly from Sankoff and Garrod's we notice that the moves of our discourse algorithm are as follows:

i. after utterance 1. where "john" is computed as Expected Topic, in utterance 2. even though "a table" is presented on the scene by a "there-sentence", John is reinforced as Main Topic;

ii. when the "waiter" appears as definite NP, the current state of the FSDA builds up some expectations on this new character and there is a local movement in the discourse structure, in case the waiter might become a new possible character. However, since the waiter is understood as a Role in the Restaurant location or scenario, the Main Topic is not demoted to Secondary Topic;

iii. in fact, when the discourse continues no such thing takes place, the waiter is not reinforced and we are in presence of the so-called "psychological statement" which is computed as belonging to the Main Topic, which has not changed and is always John.

In the second text the perspective changes with the presentation of a "man in the corner", who, as in the previous text, is not superseded by the introduction of the definite NP "the waiter" for the same reasons explained above. Two possibilities arise now: we could either have John become the Main Topic or "a man". From tests executed by us the first seems to be the case: this might be explained by the presence of John, a proper name, which is more individuating than an indefinite general noun like "a man". Suppose now that the second
sentence were "In the corner was sitting his brother". In this case the definite NP individuated by a possessive pronoun would revert the perspective.

When we get to text three, the waiter is presented on the scene as an indefinite NP and then reinforced him with a definite NP. However the system does not interchange the waiter with John simply because he is understood as a role in the scenario. Besides, the waiter becomes a secondary topic who in other words could become a main topic in case the text shifts attention to him and leaves John in the background.

In text four, we have added another utterance with an additional pronoun, "him" and reintroduced John or resumed John on the scene. But the man is Main Topic and maintains its prominence until the end.

In text five things are more complicated: as in text three the presentation utterance introduces the waiter on the scene. However, as happened before, John is reinforced as Main Topic and the waiter is assigned as expected Topic. The two pronouns in the following utterance are bound in this way: the subject to John - Main Topic - and the object to the waiter, thus becoming a secondary topic, keeping John in focus. In the remaining utterances the waiter is always kept in the background and in the final utterances he disappears.

Text six is again like text three: in utterance three, the two pronouns are bound on the basis of the current status of the FSDA, the subject "he" is bound to the main topic, John, and the "man" becomes secondary topic. Then the waiter arrives on the scene and we understand that John is being served and that the man is simply watching.

Finally text seven, reproposes the presentation of a man as happened before in text five. However the next utterance introduces the waiter on the scene: now all three characters have been introduced on the scene, and the following utterance has two pronouns to bind. The FSDA decides as usual according to prominence as established independenly in the Topics Hierarchy: the waiter disappears and the man and John respectively are assigned as antecedents of the two pronouns. As happened in text five, John remains the Main Topic until the end.

Text 1

r01.obj - [john, went, into, a, restaurant]
WEIGHTED LIST OF TOPICS :
ref_ex(sn2, john, [+ ref, 0def, nil, nil, - pro, - ana, - class], 3, m, s, [hum], subj/agent)/ -10
ref_ex(sn5, restaurant, [+ ref, - def, nil, nil, - pro, - ana, + class], 3, s, [place], obl/locat)/1020
TOPICS HIERARCHY:
EXPECTED TOPIC : ref_ex(sn2, john, _, 3, m, s, [hum], subj/agent)
POTENTIAL TOPICS : ref_ex(sn5, restaurant, _, 3, s, [place], obl/locat)
state(1, change)

r02.obj - [there, was, a, table, in, the, corner]
WEIGHTED LIST OF TOPICS :
ref_ex(sn11, table, [+ ref, - def, nil, nil, - pro, - ana, + class], 3, _, s, [place, subj_foc/th_nonaff])/3
ref_ex(sn14, corner, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, _, s, [place, obj], subj/locat)/58
TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn2, john, _, 3, m, s, [hum], _)
SECONDARY TOPIC : ref_ex(_, table, _, 3, _, s, [place, obj],_/)
state(2, continue)
DISC_DOMAIN: subjective SOC: john/_ from 2-n1
r03.obj - [the, waiter, took, the, order]
WEIGHTED LIST OF TOPICS:
ref_ex(sn1, waiter, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [hum, social], subj/agent)/10
ref_ex(sn4, order, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, _, s, [activ], obj/th_aff)/115

TOPICS HIERARCHY:
EXPECTED TOPIC: ref_ex(sn1, waiter, _, 3, m, s, [hum, social], subj/agent)
MAIN TOPIC: ref_ex(sn2, john, _, 3, m, s, [hum], _)
POTENTIAL TOPICS: ref_ex(sn4, order, _, 3, _, s, [activ], obj/th_aff)

state(3, retaining)

r04.obj - [the, atmosphere, was, warm, and, friendly]
WEIGHTED LIST OF TOPICS:
ref_ex(sn1, atmosphere, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, _, s, [subst], subj/th_bound)/114

TOPICS HIERARCHY:
MAIN TOPIC: ref_ex(sn2, john, _, 3, m, s, [hum], _)

state(4, retaining)

DISC_DOMAIN: subjective SOC: john/_ from 4-n1

r05.obj - [he, began, to, read, his, book]
WEIGHTED LIST OF TOPICS:
ref_ex(sn2, he, [+ ref, + def, nil, nil, + pro, + ana, + me], 3, m, s, [hum], subj/agent)/-20
ref_ex(sn13, book, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, _, s, [obj], obj/th_aff)/125

TOPICS HIERARCHY:
MAIN TOPIC: ref_ex(sn2, john, _, 3, m, s, [hum], _)
SECONDARY TOPIC: ref_ex(sn13, book, _, 3, _, s, [obj], obj/th_aff)

state(5, continue)

Text 2

r02.obj - [there, was, a, man, in, the, corner]
WEIGHTED LIST OF TOPICS:
ref_ex(sn8, man, [+ ref, - def, nil, nil, + pro, + ana, + class], 3, m, s, [hum], subj/th_nonaff)/13
ref_ex(sn37, corner, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, _, s, [place, obj], obj/nil)/130

TOPICS HIERARCHY:
EXPECTED TOPIC: ref_ex(sn8, man, _, 3, m, s, [hum], subj/th_nonaff)
MAIN TOPIC: ref_ex(sn3, john, _, 3, m, s, _/)

POTENTIAL TOPICS: ref_ex(sn37, corner, _, 3, _, s, [place, obj], obj/nil)

state(2, retaining)

r03.obj - [the, waiter, took, the, order, .]
WEIGHTED LIST OF TOPICS:
ref_ex(sn1, waiter, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [hum, social], subj/agent)/10
ref_ex(sn33, order, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, _, s, [activ], obj/th_aff)/115

TOPICS HIERARCHY:
MAIN TOPIC: ref_ex(sn3, john, _, 3, m, s, _/)
SECONDARY TOPIC: ref_ex(sn8, man, _, 3, m, s, [hum], _/)

POTENTIAL TOPICS: ref_ex(sn1, waiter, _, 3, m, s, hum, social], subj/agent)

state(3, continue)

r04.obj - [the, atmosphere, was, warm, and, friendly, .]
WEIGHTED LIST OF TOPICS:
ref_ex(sn1, atmosphere, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, _, s, [subst], subj/th_bound)/114

TOPICS HIERARCHY:
MAIN TOPIC: ref_ex(sn3, john, _, 3, m, s, [hum], _)}
SECONDARY TOPIC : ref_ex(sn8, man, _, 3, m, s, [hum],_/)
POTENTIAL TOPICS : ref_ex(sn33, order, _, 3, _, s, [activ], obj/th_aff)

Text 3
r02.obj - [there, was, a, waiter, in, the, corner]
WEIGHTED LIST OF TOPICS :
ref_ex(sn8, waiter, [+ ref, - def, nil, nil, - pro, - ana, + class], 3, m, s, [hum, social], subj/th_nonaff)/13
ref_ex(sn37, corner, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, _, s, [place, obj], obj/nil)/130

TOPICS HIERARCHY:
EXPECTED TOPIC : ref_ex(sn8, waiter, _, 3, m, s, [hum, social], subj/th_nonaff)
MAIN TOPIC : ref_ex(sn3, john, _, 3, m, s, _, _)
POTENTIAL TOPICS : ref_ex(sn37, corner, _, 3, _, s, [place, obj], obj/nil)

state(2, retaining)

Text 4
This is like text 2 up to utterance 4 which is now a new one.

r04.obj - [john, waved, at, him]
This is equal to Text two, in which a man is introduced on the scene. In particular the last utterance is not included because it is the same as the one ending text 2. This is the continuation:

r03.obj - [he, waved, at, him]
WEIGHTED LIST OF TOPICS :
ref_ex(sn2, he, [+ ref, + def, nil, nil, + pro, + ana, + me], 3, m, s, [hum], subj/actor)/ -19
ref_ex(sn6, him, [+ ref, + def, nil, nil, + pro, + ana, + me], 3, m, s, [hum], obj/actor)/981
TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn6, john, _, 3, m, s, [hum], _)
SECONDARY TOPIC : ref_ex(sn2, man, _, 3, m, s, [hum], _)
state(3, continue)

r04.obj - [the, waiter, took, the, order, ]
WEIGHTED LIST OF TOPICS :
ref_ex(sn1, waiter, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [hum, social], subj/agent)/10
ref_ex(sn33, order, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, s, [activ], obj/th_aff)/115
EXPECTED TOPIC : ref_ex(sn1, waiter, _, 3, m, s, [hum, social], subj/agent)
MAIN TOPIC : ref_ex(sn6, john, _, 3, m, s, [hum], obj/actor)
SECONDARY TOPIC : ref_ex(sn2, man, _, 3, m, s, [hum], _)
POTENTIAL TOPICS : ref_ex(sn33, order, _, 3, s, [activ], obj/th_aff)
state(4, retaining)

r05.obj - [the, atmosphere, was, warm, and, friendly, ]
WEIGHTED LIST OF TOPICS :
ref_ex(sn1, atmosphere, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [hum], subj/th_bound)/114
TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(_, john, _, 3, m, s, [hum], obj/actor)
SECONDARY TOPIC : ref_ex(_, man, _, 3, m, s, [hum], _)
state(5, retaining)

Text 6
This is similar to Text 3. up to utterance No 2 where a waiter is introduced on the scene. This is the continuation:

r03.obj - [he, waved, at, him]
WEIGHTED LIST OF TOPICS :
ref_ex(sn2, he, [+ ref, + def, nil, nil, + pro, + ana, + me], 3, m, s, [hum], subj/actor)/19
ref_ex(sn6, him, [+ ref, + def, nil, nil, + pro, + ana, + me], 3, m, s, [hum], sn/obj/actor)/981
TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn2, john, _, 3, m, s, [hum], _)
SECONDARY TOPIC : ref_ex(sn6, waiter, _, 3, m, s, [hum], sn/obj/actor)
state(3, continue)

r04.obj - [the, waiter, took, the, order, .]
WEIGHTED LIST OF TOPICS :
ref_ex(sn1, waiter, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, m, s, [hum, social], subj/agent)/10
ref_ex(sn33, order, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, _, s, [activ], obj/th_aff)/115
TOPICS HIERARCHY:
EXPECTED TOPIC : ref_ex(sn1, waiter, _, 3, m, s, [hum, social], subj/agent)
MAIN TOPIC : ref_ex(sn2, john, _, 3, m, s, [hum], _)
POTENTIAL TOPICS : ref_ex(sn33, order, _, 3, _, s, [activ], obj/th_aff)
state(4, continue)

r05.obj - [the, atmosphere, was, warm, and, friendly, .]
WEIGHTED LIST OF TOPICS :
ref_ex(sn1, atmosphere, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, _, s, [subst], subj/th_bound)/114
TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn2, john, _, 3, m, s, [hum], _)
state(5, continue)

r06.obj - [he, began, to, read, his, book, .]
WEIGHTED LIST OF TOPICS :
ref_ex(sn2, he, [+ ref, + def, nil, nil, + pro, + ana, + me], 3, m, s, [hum], subj/agent)/-20
ref_ex(sn13, book, [+ ref, + def, nil, nil, - pro, - ana, + class], 3, _, s, [obj], obj/th_aff)/125
TOPICS HIERARCHY:
MAIN TOPIC : ref_ex(sn2, john, _, 3, m, s, [hum], _)
POTENTIAL TOPICS : ref_ex(sn13, book, _, 3, _, s, [obj], obj/th_aff)
state(6, continue)

Text 7
This text is the same as text 4 up to utterance No3 but the continuation is different: here in particular we have substituted John with a pronoun, thus increasing the overall degree of ambiguity.

r04.obj - [he, waved, at, him]
WEIGHTED LIST OF TOPICS :
The problem of anaphora resolution (hence AR) looms more and more as a prominent one in unrestricted text processing due to the need to recover semantically consistent information in most current NLP applications. This problem does not lend itself easily to a statistical approach so that rule-based approaches seem the only viable solution.

At first we shall present three state-of-the-art algorithms for anaphora resolution—GuiTAR, JavaRAP, MARS—on the basis of a portion of Susan Corpus (derived from Brown Corpus) a much richer testbed than the ones previously used for evaluation, and in any case a much more comparable source with such texts as newspaper articles and stories. The portion of text chosen has an adequate size which lends itself to significant statistical measurements: it is portion A, counting 35,000 tokens and some 1000 third person pronominal expressions. Texts used previously ranged from scientific manuals to descriptive scientific texts and were generally poor on pronouns and rich on nominal descriptions. Two of the algorithms—GuiTAR and JavaRAP—use Charniak’s parser output, which contributes to the homogeneity of the type of knowledge passed to the resolution procedure. MARS, on the contrary, uses a more sophisticated input, the one provided by Connexor FDG-parser.

We will then present the partial version of the algorithm for anaphora resolution of GETARUNS. The version of the algorithm presented here is a newly elaborated one, and is devoted to unrestricted text processing. It is an upgraded version from the one discussed in Delmonte (1999; 2002a; 2002b) and tries to incorporate as much as possible of the more
In the system, three levels are indicated: Clause level, i.e. simple sentences; Utterance level, i.e. complex sentences; Discourse level, i.e. intersententially. This will constitute the main difference from other systems for anaphora resolution that we outline below. So first of all, the subdivision of the system into two levels: Clause level – intrasentential pronominal phenomena – where all pronominal expressions contained in modifiers, adjuncts or complement clauses receive their antecedent locally. Possessive pronouns, pronouns contained in relative clauses and complement clauses choose preferentially their antecedents from list of higher level referring expressions. Not so for those pronouns contained in matrix clauses. In particular the ones in subject position are to be coreferred in the discourse. This requires the system to be equipped with a History List of all referring expressions to be used when needed.

7.1. JavaRAP

As reported by the authors (Long Qiu, Min-Yen Kan, Tat-Seng Chua, 2004) of the JAVA implementation, head-dependent relations required by RAP are provided by looking into the structural “argument domain” for arguments and into the structural “adjunct domain” for adjuncts. Domain information is important to establish disjunction relations, i.e. to tell whether a third person pronoun can look for antecedents within a certain structural domain or not. According to Binding Principles, Anaphors (i.e. reciprocal and reflexive pronouns), must
be bound – search for their binder-antecedent – in their same binding domain – roughly corresponding to the notion of structural “argument/adjunct domain”. Within the same domains, Pronouns must be free. Head-argument or head-adjunct relation is determined whenever two or more NPs are sibling of the same VP.

Additional information is related to agreement features, which in the case of pronominal expressions are directly derived. As for nominal expressions, features are expressed in case they are either available on the verb – for SUBJect NPs – or else if they are expressed on the noun and some other tricks are performed for conjoined nouns. Gender is looked up in the list of names available on the web. This list is also used to provide the semantic feature of animacy.

RAP is also used to find pleonastic pronouns, i.e. pronouns which have no referents. To detect conditions for pleonastic pronouns a list of patterns is indicated, which used both lexical and structural information.

Salience weight is produced for each candidate antecedent from a set of salience factors. These factors include main Grammatical Relations, Headedness, non Adverbiality, belonging to the same sentence. The information is computed again by RAP, directly on the syntactic structure. The weight computed for each noun phrase is divided by two in case the distance from the current sentence increases. Only NPs contained within a distance of three sentences preceding the anaphor are considered by JavaRAP.

7.2. GuiTAR

The authors (Poesio, M. and Mijail A. Kabadjov 2004) present their algorithm as an attempt at providing a domain independent anaphora resolution module, “that developers of NLE applications can pick off the shelf in the way of tokenizers, POS taggers, parsers, or Named Entity classifiers”. For these reasons, GuiTAR has been designed to be as independent as possible from other modules, and to be as modular as possible, thus “allowing for the possibility of replacing specific components (e.g., the pronoun resolution component)”.

The authors have also made an attempt at specifying what they call the Minimal Anaphoric Syntax (MAS) and have devised a markup language based on GNOME mark-up scheme. In MAS, Nominal Expressions constitute the main processing units, and are identified with the tag NE <ne>, which have a CAT attribute, specifying the NP type: the-np, pronoun etc., as well as Person, Number and Gender attributes for agreement features. Also the internal structure of the NP is marked with Mod and NPHead tags.

The pre-processing phase uses a syntactic guesser which is a chunker of NPs based on heuristics. All NEs add up to a discourse model – or better History List - which is then used as the basic domain where Discourse Segments are contained. Each Discourse Segment in turn may be constituted by one or more Utterances. Each Utterance in turn contains a list of forward looking centers Cfs.

The Anaphora Resolution algorithm implemented is the one proposed by MARS which will be commented below. The authors also implemented a simple algorithm for resolving Definite Descriptions on the basis of the History List by a same head matching approach.
7.3. MARS

The approach is presented as a knowledge poor anaphora resolution algorithm (Mitkov R. [1995;1998]), which makes use of POS and NP chunking, it tries to individuate pleonastic “it” occurrences, and assigns animacy. The weighting algorithm seems to contain the most original approach. It is organized with a filtering approach by a series of indicators that are used to boost or reduce the score for antecedenthood to a given NP. The indicators are the following ones:

FNP (First NP); INDEF (Indefinite NP); IV (Indicating Verbs); REI (Lexical Reiteration); SH (Section Heading Preference); CM (Collocation Match); PNP (Prepositional Noun Phrases); IR (Immediate Reference); SI (Sequential Instructions); RD (Referential Distance); TP (Term Preference), As the author comments, antecedent indicators (preferences) play a decisive role in tracking down the antecedent from a set of possible candidates. Candidates are assigned a score (-1, 0, 1 or 2) for each indicator; the candidate with the highest aggregate score is proposed as the antecedent.

The authors comment is that antecedent indicators have been identified empirically and are related to salience (definiteness, givenness, indicating verbs, lexical reiteration, section heading preference, "non-prepositional" noun phrases), to structural matches (collocation, immediate reference), to referential distance or to preference of terms. Whilst some of the indicators are more genre-specific (term preference) and others are less genre-specific ("immediate reference"), the majority appear to be genre-independent. However it is clear that most of the indicators have been suggested for lack of better information, in particular no syntactic constituency was available.

In a more recent paper (Mitkov et al., 2003) MARS has been fully reimplemented and the indicators updated. The authors seem to acknowledge the fact that anaphora resolution is a much more difficult task than previous work had suggested. In unrestricted text analysis, the tasks involved in the anaphora resolution process contribute a lot of uncertainty and errors that may be the cause for low performance measures.

The actual algorithm uses the output of Connexor’s FDG Parser, filters instances of “it” and eliminates pleonastic cases, then produces a list of potential antecedents by extracting nominal and pronominal heads from NPs preceding the pronoun. Constraints are then applied to this list in order to produce the “set of competing candidates” to be considered further, i.e. those candidates that agree in number and gender with the pronoun, and also obey syntactic constraints. Candidates are assigned a score according to indicators, and the candidate with the highest score is selected as the antecedent of the pronoun.

MARS was tested on a set of different files from technical domains where over 2000 pronouns were present and the average success rate was about 60%. A slight improvement of 1.5% was achieved by introducing the use of Genetic Algorithms in the evaluation phase.

The new version of MARS includes three new indicators which seem more general and applicable to any text, so we shall comment on them.

- Frequent Candidates (FC) – this is a boosting score for most frequent three NPs
- Syntactic Parallelism (SP) – this is a boosting score for NPs with the same syntactic role as the pronoun, roles provided by the FDG-Parser
- Boost Pronoun (BP) – pronoun candidates are given a bonus (no indication of conditions for such a bonus)
The authors also reimplemented in a significant way the indicator First NPs which has been renamed,

- Obliqueness (OBL) – score grammatical functions, SUBJect > OBJect > IndirectOBJect > Undefined

The system has no clause splitter so that what is actually computed is not the clause distance but sentence distance. MARS has a procedure for automatically identifying pleonastic pronouns: the classification is done by means of 35 features organized into 6 types and are expressed by a mixture of lexical and grammatical heuristics. The output should be a fine-grained characterization of the phenomenon of the use of pleonastic pronouns which includes, among others, discourse anaphora, clause level anaphora and idiomatic cases.

In the same paper, the authors deal with two more important topics: syntactic constraints and animacy identification. They implemented a type of constraint which refers explicitly to constituency domains, to argument structure and to command and precedence relations which must be derived from the output of the parser: however they don’t give any indication of how this has been done.

7.4. GETARUNS

We already described our algorithms and the theoretical background which inspired it in the previous portion of this chapter. Whereas the old version of the system had a limited vocabulary and was intended to work only in limited domains with high precision, the partial version of the system has been created to cope with unrestricted text. In Delmonte (2002), we reported preliminary results obtained on a corpus of anaphorically annotated texts made available by R.Mitkov on his website. Both definite descriptions and pronominal expressions were considered, success rate was at 75% F-measure. In that case we used the shallow and robust parser which produced only NP chunks which were then used to fire anaphoric processes. However the texts making up the corpus were technical manuals, where the scope and usage of pronominal expressions is very limited.

The current algorithm for anaphora resolution works on the output of a partial deep robust parser already presented in Chapter 2, which builds an indexed linear list of dependency structures where clause boundaries are clearly indicated; differently from Connexor, our system elaborates both grammatical relations and semantic roles information for arguments and adjuncts. Semantic roles are very important in the weighting procedures. Our system also produces implicit grammatical relations which are either controlled SUBJects of untensed clauses, arguments or adjuncts of relative clauses.

As to the anaphoric resolution algorithm, it is based on the original Sidner’s (1983:Chapter 5) and Webber’s (1983:Chapter 6) intuitions on Focussing in Discourse, and follows strictly the indication already reported in this chapter and the previous one. We find distributed, local approaches to anaphora resolution more efficient than monolithic, global ones. In particular we believe that due to the relevance of structural constraints in the treatment of locally restricted classes of pronominal expressions, it is more appropriate to activate different procedures which by dealing separately with non-locally restricted classes
also afford separate evaluation procedures. There are also at least two principled reasons for the separation into two classes.

The first reason is a theoretical one and has already been discussed in previous chapters. Linguistic theory has long since established without any doubt the existence in most languages of the world of at least two classes: the class of pronouns which must be bound locally in a given domain and the class of pronouns which must be left free in the same domain – as a matter of fact, English also has a third class of pronominals, the so-called long-distance subject-of-consciousness bound pronouns (see Zribi-Hertz A., 1989).

The second reason is empirical. Anaphora resolution is usually carried out by searching antecedents backward w.r.t. the position of the current anaphoric expression. In our approach, we proceed in a clause by clause fashion, weighting each candidate antecedent w.r.t. that domain, trying to resolve it locally. Weighting criteria are amenable on the one hand to linear precedence constraints, with scores assigned on a functional/semantic basis. On the other hand, these criteria may be overrun by a functional ranking of clauses which requires to treat main clauses differently from secondary clauses, and these two differently from complement clauses. On the contrary, global algorithms neglect altogether such requirements: they weight each referring expression w.r.t. the utterance, linear precedence is only physically evaluated, no functional correction is introduced.

### 8. REFERENTIAL POLICIES AND ALGORITHMS

There are also two general referential policy assumption that we adopt in our approach: The first one is related to pronominal expressions, the second one to referring expressions or entities to be asserted in the History List, and are expressed as follows:

- no more than two pronominal expressions are allowed to refer back in the previous discourse portion;
- at discourse level, referring expressions are stored in a push-down stack according to Persistence principles.

Persistence principles respond to psychological principles and limit the topicality space available to user w.r.t. a given text. It has a bidimensional nature: it is determined both in relation to an overall topicality frequency value and to an utterance number proximity value.

Only “persistent” referring expressions are allowed to build up the History List, where persistence is established on the basis of the frequency of topicality for each referring expression which must be higher than 1. All referring expression asserted as Topic (Secondary, Potential) only once are discarded in case they appeared at a distance measured in 5 previous utterances. Proximate referring expressions are allowed to be asserted in the History List.

In particular, if Mitkov considers the paragraph as the discourse unit most suitable for coreferring and cospecifying operation at discourse level, we prefer to adopt a parameterized procedure which is definable by the user and activated automatically: it can be fired within a number that can vary from every 10 up to 50 sentences. Our procedure has the task to prune the topicality space and reduce the number of perspective topic for Main and Secondary
Topic. Thus we garbage-collect all non-relevant entities. This responds to the empirically validated fact that as the distance between first and second mention of the same referring expression increases, people are obliged to repeat the same linguistic description, using a definite expression or a bare NP. Indefinites are unallowed and may only serve as first mention; they can also be used as bridging expression within opaque propositions. The first procedure is organized as follows:

A. For each clause,
- we collect all referential expressions and weight them (see B below for criteria) – this is followed by an automatic ranking;
- then we subtract pronominal expressions;
- at clause level, we try to bind personal and possessive pronouns obeying specific structural properties; we also bind reflexive pronouns and reciprocals if any, which must be bound obligatorily in this domain;
- when binding a pronoun, we check for disjointness w.r.t. a previously bound pronoun if any;
- all unbound pronouns and all remaining personal pronouns are asserted as “externals”, and are passed up to the higher clause levels;

B. Weighting is carried out by taking into account the following linguistic properties associated to each referring expression:
- Grammatical Function with usual hierarchy (SUBJ > ARG_MOD > OBJ > OBJ2 > IOBJ > NCMOD);
- Semantic Roles, as they have been labelled in FrameNet, and in our manually produced frequency lexicon of English;
- Animacy: we use 75 semantic features derived from WordNet general concepts, and reward Human and Institution/Company labelled referring expressions;
- Functional Clause Type is further used to introduce penalties associated to those referring expressions which don’t belong to main clause.

C. Then we turn at the higher level – if any -, and we proceed as in A., in addition
- we try to bind pronouns passed up by the lower clause levels
- if successful, this will activate a retract of the “external” label and a label of “antecedenthood” for the current pronoun with a given antecedent;
- the best antecedent is chosen by recursively trying to match features of the pronoun with the first available antecedent previously ranked by weighting;

D. This is repeated until all clauses are examined and all pronouns are scrutinised and bound or left free.

E. Pronouns left free – those asserted as externals – will be matched tentatively with the best candidates provided this time by a “centering-like” algorithm.

Step A. is identical and is recursively repeated until all clauses are processed.

Then, we move to step B. which in this case will use all referring expressions present in the utterance, rather than only those available locally.

8.1. Focussing Revisited
Our version of the focussing algorithm follows Sidner’s proposal (Sidner C., 1983; Grosz B., Sidner C., 1986), to use a Focus Stack, a certain Focus Algorithm with Focus movements and data structures to allow for processing simple inferential relations between different linguistic descriptions co-specifying or coreferring to a given entity.

Our Focus Algorithm is organized as follows: for each utterance, we assert three “centers” that we call Main, Secondary and the first Potential Topic, which represent the best three referring expressions as they have been weighted in the candidate list used for pronominal binding; then we also keep a list of Potential Topics for the remaining best candidates. These three best candidates repositories are renovated at each new utterance, and are used both to resolve pronominal and nominal co-specification and coreference: this is done both in case of strict identity of linguistic description and of non-identity. The second case may occur either when derivational morphological properties allow the two referring expressions to be matched successfully, or when a simple hyponym/hypernym relation is entertained by two terms, one of which is contained in the list of referring expressions collected from the current sentence, and the other is among one of the entities stored in the focus list.

The Main Topic may be regarded the Forward Looking Center in the centering terminology or the Current Focus. All entities are stored in the History List (HL) which is a stack containing their morphological and semantic features: this is not to be confused with a Discourse Model - what we do in the deep complete system anaphora resolution module – which is a highly semantically wrought elaboration of the current text. In the HL every new entity is assigned a semantic index which identifies it uniquely. To allow for Persistence evaluation, we also assert semantic properties associated to each entity, i.e. we store the information of topicality (i.e. whether it has been evaluated as Main, Secondary or Potential Topic), together with the semantic ID and the number of the current utterance. This is subsequently used to measure the degree of Persistence in the overall text of a given entity, as explained below.

In order to decide which entity has to become Main, Secondary or Potential Topic we proceed as follows:

- we collect all entities present in the History List with their semantic identifier and feature list and proceed to an additional weighting procedure;
- nominal expressions are divided up into four semantic types: definite, indefinite, bare NPs, quantified NPs. Both definite and indefinite NP may be computed as new or old entity according to contextual conditions as will be discussed below and are given a rewarding score;
- we enumerate for each entity its persistence in the previous text, and keep entities which have frequency higher than 1, we discard the others;
- we recover entities which have been asserted in the HL in proximity to the current utterance, up to four utterances back;
- we use this list to “resolve” referring expressions contained in the current utterance;
- if this succeeds, we use the “resolved” entities as new Main, Secondary, and Potential Topics and assert the rest in the Potential Topics stack;
- if this fails – also partially – we use the best candidates in the weighted list of referring expressions to assert the new Topics. It may be the case that both resolved and current best candidates are used, and this is by far the most common case.
9. EVALUATION AND GENERAL DISCUSSION

Evaluating anaphora resolution systems calls for a reformulation of the usual parameters of Precision and Recall as introduced in IR/IE field: in that case, there are two levels that are used as valuable results; a first stage where systems are measured for their capacity to retrieve/extract relevant items from the corpus/web (coverage-recall). Then a second stage follows in which systems are evaluated for their capacity to match the content of the query (accuracy-precision). In the field of IR/IE items to be matched are usually constituted by words/phrases and pattern-matching procedures are the norm. However, for AR systems this is not sufficient and NLP heavy techniques are used to get evaluable results. As Mitkov also notes, this phase jeopardizes the capacity of AR systems to reach satisfactory accuracy scores simply because of its intrinsic weakness: none of the off-the-shelf parsers currently available overcomes 90% accuracy.

To clarify these issues, we present here below two Tables: in the first one we report data related to the vexed question of whether pleonastic “it” should be regarded as part of the task of anaphora resolution or rather part of a separate classification task – as suggested in a number of papers by Mitkov. In the former case, they should contribute to the overall anaphora resolution evaluation metrics; in the latter case they should be compute separately as a case of classification over all occurrences of “it” in the current dataset and discarded from the overall count. Even though we don’t agree fully with Mitkov’s position, we find it useful to deal with “it” separately, due to its high inherent ambiguity. Besides, it is true that the AR task is not like any Information Retrieval task.

In Table 32 below we reported figures for “it” in order to evaluate the three algorithms in relation to the classification task. Then in Table 33 we report general data where we computed the two types of accuracy reported in the literature. In Table 32 we split results for “it” into Wrong Reference vs. Wrong Classification: following Mitkov, in case we only computed anaphora related cases and disregarded those cases of “it” which were wrongly classified as expletives. Expletive “it” present in the text are 189: so at first we computed coverage and accuracy with the usual formula that we report below. Then we subtracted wrongly classified cases from the number of total “it” found in one case (following Mitkov who claims that wrongly classified “it” found by the system should not count; in another case, this number is subtracted from the total number of “it” to be found in the text. Only for MARS we then computed different measures of Coverage and Accuracy. If we regard this approach worth pursuing, we come up with two Adjusted Accuracy measures which are related to the revised total numbers of anaphors by the two subtractions indicated above.

We computed manually all third person pronominal expressions and came up with a figure 982 which is only confirmed by one of the three systems considered: JavaRAP. Pronouns considered are the following one, lower case and upper case included:

- Possessives – his, its, her, hers, their, theirs
- Personals – he, she, it, they, him, her, it, them (where “it” and “her” have to be disambiguated)
- Reflexives – himself, itself, herself, themselves
Table 32. Compared Results for Expletive “it”

<table>
<thead>
<tr>
<th></th>
<th>MARS</th>
<th>JavaRAP</th>
<th>GuiTAR</th>
<th>GETARUNS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coverage</td>
<td>163 (86.2%)</td>
<td>188 (99.5%)</td>
<td>188 (99.5%)</td>
<td>171 (91%)</td>
</tr>
<tr>
<td>Accuracy 1</td>
<td>63 (33.3%)</td>
<td>73 (38.6%)</td>
<td>75 (39.7%)</td>
<td>87 (46%)</td>
</tr>
<tr>
<td>Wrong</td>
<td>44</td>
<td>49</td>
<td>64</td>
<td>53</td>
</tr>
<tr>
<td>Classification</td>
<td>163-44=119</td>
<td>189-49=140</td>
<td>189-64=125</td>
<td>189-53=136</td>
</tr>
<tr>
<td>Wrong Reference</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Accuracy 2</td>
<td>63 (38.6%)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adjusted</td>
<td>63 (52.9%)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Accuracy 2</td>
<td>63 (43.4%)</td>
<td>73 (52.1%)</td>
<td>75 (60%)</td>
<td>87 (64 %)</td>
</tr>
</tbody>
</table>

There are 16 different wordforms. As can be seen from the table below, apart from JavaRAP, none of the other systems considered comes close to 100% coverage.

Computing general measures for Precision and Recall we have three quantities (see also Poesio & Kabadjov):
- total number of anaphors present in the text;
- anaphors identified by the system;
- correctly resolved anaphors.

Formulas related to Accuracy/Success Rate or Precision are as follows: Accuracy1 = number of successfully resolved anaphors/number of all anaphors; Accuracy2 = number of successfully resolved anaphors/number of anaphors found (attempted to be resolved). Recall - which should correspond to Coverage - we come up with formula: \( R = \frac{\text{number of anaphors found}}{\text{number of all anaphors to be resolved (present in the text)}} \). Finally the formula for F-measure is as follows: \( 2 \cdot P \cdot R / (P + R) \) where \( P \) is chosen as Accuracy 2.

Table 33. Overall results Coverage/Accuracy

<table>
<thead>
<tr>
<th></th>
<th>COVERAGE</th>
<th>ACCURACY 1</th>
<th>ACCURACY 2</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>MARS</td>
<td>936 (95.3%)</td>
<td>403/982 (41.5%)</td>
<td>403/903 (43%)</td>
<td>59.26%</td>
</tr>
<tr>
<td>JavaRAP</td>
<td>981 (100%)</td>
<td>490/982 (49.9%)</td>
<td>490/981 (50%)</td>
<td>66.7%</td>
</tr>
<tr>
<td>GUITAR</td>
<td>824 (84.8%)</td>
<td>445/982 (45.8%)</td>
<td>445/824 (54%)</td>
<td>65.98%</td>
</tr>
<tr>
<td>GETARUNS</td>
<td>885 (90.1%)</td>
<td>555/982 (56.5%)</td>
<td>555/885 (62.7%)</td>
<td>73.94%</td>
</tr>
</tbody>
</table>

In absolute terms best accuracy figures have been obtained by GETARUNS, followed by JavaRAP. So it is still thanks to the classic Recall formula that this result stands out clearly. We also produced another table which can however only be worked out for our system, which uses a distributed approach. We managed to separate pronominal expressions in relation to their contribution at the different levels of anaphora resolution considered: clause level, utterance level, discourse level. At clause level, only those pronouns which must be bound
locally are checked, as is the case with reflexive pronouns, possessives, some cases of expletive ‘it’: both arguments and adjuncts may contribute the appropriate antecedent. At utterance level, in case the sentence is complex or there is more than one clause, also personal subject/object pronouns may be bound (if only preferentially so). Eventually, those pronouns which do not find an antecedent are regarded discourse level pronouns.

We collapsed under CLAUSE all pronouns bound at clause and utterance level; DISCOURSE contains only sentence external pronouns. Expletives have been computed in a separate column.

Table 34. GETARUNS pronoun collapses at structural level

<table>
<thead>
<tr>
<th></th>
<th>CLAUSE</th>
<th>DISCOURSE</th>
<th>EXPLETIVES</th>
<th>TOTALS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pronouns found</td>
<td>410</td>
<td>366</td>
<td>109</td>
<td>885</td>
</tr>
<tr>
<td>Correct</td>
<td>266</td>
<td>222</td>
<td>67</td>
<td>555</td>
</tr>
<tr>
<td>Errors made</td>
<td>144</td>
<td>144</td>
<td>42</td>
<td>330</td>
</tr>
</tbody>
</table>

As can be easily noticed, the highest percentage of pronouns found is at Clause level: this is not however the best performance of the system, which on the contrary performs better at discourse level. Expletives contribute by far the highest correct result. We also found correctly 47 ‘there’ expletives and 6 correctly classified pronominal ‘there’ which however have been left unbound. The system also found 48 occurrences of deictic discourse bound “this” and “that”, which corresponds to the full coverage.

Finally, nominal expressions: the History List (HL) has been incremented up to 2243 new entities. The system identified 2773 entities from the HL by matching their linguistic description. The overall number of resolution actions taken by the Discourse Level algorithm is 1861: this includes both cases of nominal and pronominal expressions. However, since only 366 can be pronouns, the remaining 1500 resolution actions have been carried out on nominal expressions present in the HL. If we compare these results to the ones computed by GuiTAR, which assign semantic indices to NamedEntities disregarding their status of anaphora, we can see that the whole text is made up of 12731 NEs. GuiTAR finds 1585 cases of identity relations between a NE and an antecedent. However, GuiTAR introduces always new indices and creates local antecedent-referring expression chains rather than repeating the same index of the chain head. In this way, it is difficult if not impossible to compute how many times the text corefers/cospecifies to the same referring expressions. On the contrary, in our case, this can be easily computed by counting how many times the same semantic index is being repeated in a “resolution” or “identity” action of the anaphora resolution algorithm. For instance, the Jury is coreferred/cospecified 12 times; Price Daniel also 12 times and so on.

The error rate of both Charniak’s and Connexor’s systems as reported in the literature, is approximately the same, 20%; this notwithstanding, MARS has a slightly reduced coverage when compared with JavaRAP, 96%. GuiTAR has the worst coverage, 85%. As to accuracy, none of the three algorithms overruns 50%: JavaRAP has the best score 49.9%. However GETARUNS has 63% correct score, with 90% coverage.
9.1. Anaphora Resolution in RTE

RTE3 (2006 – Venice) introduced as a novelty - with regard to the previous two editions - a certain number (117 in the Test set – 135 in the Dev set) of long Texts, of paragraph length. This move is justified by the need to address more realistic data, and consequently to tune the whole process of semantic evaluation to the problems related to such data. Thus more relevance is given to empirical issues to be tackled, rather than to the theoretical ones, which however don’t disappear but may assume less importance.

When a system has to cope with paragraph length texts, the basic difference with short texts regards the problem of anaphora resolution. In short texts, pronominal expressions constituted a minor problem and all referring expressions were specified fully. Not so in long texts, as can be seen from the Table below, where we list pronominals from both the Test and the Development Set texts:

<table>
<thead>
<tr>
<th></th>
<th>He</th>
<th>Him</th>
<th>His</th>
<th>She</th>
<th>Her</th>
<th>It</th>
<th>Its</th>
<th>They</th>
<th>Their</th>
<th>Them</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
<td>80</td>
<td>15</td>
<td>91</td>
<td>19</td>
<td>18</td>
<td>91</td>
<td>68</td>
<td>43</td>
<td>63</td>
<td>15</td>
<td>485</td>
</tr>
<tr>
<td>Dev.</td>
<td>113</td>
<td>16</td>
<td>136</td>
<td>27</td>
<td>35</td>
<td>123</td>
<td>76</td>
<td>44</td>
<td>64</td>
<td>18</td>
<td>652</td>
</tr>
<tr>
<td>Total</td>
<td>193</td>
<td>31</td>
<td>227</td>
<td>46</td>
<td>53</td>
<td>214</td>
<td>144</td>
<td>87</td>
<td>127</td>
<td>33</td>
<td>1137</td>
</tr>
</tbody>
</table>

As can be seen from this table, the problem a system is faced with is not just to cope with an ad hoc solution for single cases where the pronoun is placed, for instance, in sentence first position and it might be easy to recover its antecedent by some empirical ad hoc procedure. The problem needs to be addressed fully and this requires a full-fledged system for anaphora resolution. One such system is shown in Fig. 3 above, where we highlight the architecture and main processes undergoing at the anaphora level. Our system computes semantic structures in a sentence by sentence fashion and any information useful to carry out anaphoric processes needs to be made available to the following portion of text, and eventually to the Semantic Evaluation that computes entailment. We will comment a number of significant examples to clarify the way in which our system operates.

In example n.3 below, the first possessive pronoun “his” is met at Utterance level – the first sentence has two clauses: clause 1, headed by the predicate DIVORCE, and clause 2, headed by MARRY. “His” will look for a masculine antecedent and Chabrol will be chosen, also for weights associated to it, being the higher subject. This will produce the following semantic structure, which is made of a Head, a Semantic Role and an Index,

- Chabrol-poss-sn2

which is the output of the substitution of “his” present in the same structure by means of information made available by the Anaphoric module. Note that the index of a modifier points to the governing head, in this case “wife”, the apposition associated to “Agnes”, which in turn is the OBJect of DIVORCE.
T/H Pair n. 3

T: Claude Chabrol divorced Agnes, his first wife, to marry the actress Stèphane Audran.

His third wife is Aurore Paquiss.

H: Aurore Paquiss married Chabrol.

When the first sentence is passed to the semantic interpreter, anaphoric processes have already been completed and the information is then transferred to semantic structure which will register the anaphoric relation by the substitution operation. However this specific relation is not the one that really matters in the current T/H pair. When the system passes to the analysis of the following sentence it has another possessive pronoun which is contained in a SUBJect NP. By definition, these pronouns take their antecedent from the discourse level. To have the system do that, the pronoun has to be left free at sentence level, i.e. it must be computed as “external” to the current sentence, and not bound locally. Discourse level processes will look for antecedents from the History list and from the so-called Topic Hierarchy, our way to compute centering (but see again Delmonte, 2006). This is shown schematically in the output of the Anaphora Resolution module shown here below, which reports the listing of pronouns, Topic Hierarchy, and Anaphora Resolution processes carried out. In this case, every referring expression will have a semantic index (SI) associated which is unique in the History List.

id_3_1

All pronouns:
1-[his-0-[[[human], cat=poss, disj=[sn1-wife], gen=mas, num=sing, pers=3, pred=he/be]]

Pronominal Binding:

external_pronoun(be, his)

ANAPHORA RESOLUTION

his resolved_as 'Claude_Chabrol'
main_resolved_as Claude_Chabrol - SI=id1
SECONDARY TOPIC: wife

id_3_3

main_resolved_as Aurore_Paquiss - SI=id4
second_resolved_as Claude_Chabrol - SI=id1

TOPIC HIERARCHY

MAIN TOPIC: 'Aurore_Paquiss'
SECONDARY TOPIC: 'Claude_Chabrol'

In the example below, n.31, the pronominal expressions are two: an Utterance level possessive pronoun bound to the local SUBJect; and a Discourse level personal pronoun “He” which receives its antecedent from the History List. In both cases, substitution with their antecedents’ head will take place in the semantic interpretation level.
T/H Pair n. 31

T: Upon receiving his Ph.D., Wetherill became a staff member at Carnegie's Department of Terrestrial Magnetism (DTM) in Washington, D.C. He originated the concept of the Concordia Diagram for the uranium-lead isotopic system.

H: Wetherill was the inventor of the concept of the Concordia Diagram.

A similar case is presented below with pair n.81, which is however must more complex than the previous example. Not only the personal pronoun appears in a subordinate clause, but the following matrix sentence contains three possessive pronouns which need to be bound to the appropriate antecedent. Eventually, the relevant pronoun, is in the following sentence which needs to recover the antecedent from the History List. But the situation now is complicated by the presence of more than one possible antecedent. This requires the activation of weighting mechanisms to evaluate the one that is higher in the hierarchy.

T/H Pair n. 81

T: The title was again created for John Holles. When he died in 1711 the title became extinct but his estates passed to his nephew Thomas Pelham, who three years later upon coming of age received the title in its third creation. In 1757 he received the additional title of "Newcastle-under-Lyne".

H: Holles received the title of "Newcastle-under-Lyne".

id_81_1

######### TOPIC HIERARCHY #########

MAIN TOPIC: 'John_Holles'
POTENTIAL TOPICS: title-def

id_81_2

All pronouns:
3-[[he]-0-[pers=3, mas, sing, [human], agent]/sn1],
1-[its-0-[object], cat=poss, disj=[sn7-creation, sn9-who], gen=neu, num=sing, pers=3, pred=it]/die],
0-[his-0-[human], cat=poss, disj=[sn4-estate], gen=mas, num=sing, pers=3, pred=he]/pass;
his-0-[disj=[sn5-Thomas_Pelham], cat=poss, pred=he, pers=3, num=sing, gen=mas]/pass]

Pronominal Binding:
ante(die, he, John_Holles)
ante(receive, its, title)
ante(pass, his, he)
main_resolved_as Thomas_Pelham - SI=id3
second_resolved_as title - SI=id2

######### TOPIC HIERARCHY #########

MAIN TOPIC: 'John_Holles'
SECONDARY TOPIC: 'Thomas_Pelham'
POTENTIAL TOPICS: title-def estate-npr nephew-npr creation-npr

id_81_3

all_prons([0-[[he]-0-[pers=3, mas, sing, [human], agent]/sn1]])
Other relevant cases are constituted by structural anaphora relations computed by syntactic grammatical processes, as the ones related to Coordinate clauses, shown in pair 14 and 125 here below.

**T/H Pair n. 14**

T: In 1956 Accardo won the Geneva Competition and in 1958 became the first prize winner of the Paganini Competition in Genoa.

H: Accardo won the Paganini Competition in Genoa.

**T/H Pair n. 125**

T: Wilkins was chosen for the England's squad which qualified for the 1986 World Cup in Mexico, and played in the opening defeat against Portugal.

H: England defeated Portugal in the 1986 World Cup.

In the following pair, both coordination and intersentential cases must be treated by the system.

**T/H Pair n. 122**

T: Another uprising was led by the mayor of Panjamo, Guanajuato, Luis Navarro Origel, beginning on September 28. He was defeated by federal troops in the open land around the town, but retreated into the mountains, where he and his men continued as guerrillas.

H: Navarro Origel became a guerrilla.
Linguistic Information Extraction for Text Correction and Summarization

Introduction

This chapter is devoted to the use of shallow and partial NLP techniques in heavily linguistically demanding tasks such as the one posed by summarization. In so doing we intend to present work on the morphology of Italian, a morphologically rich language, which has been omitted from previous chapters in order to take the reader into the heart of the text understanding system. This chapter should not be taken as an alternative way of coping with the same problem, but just as a proposal in line with current research in NLP that assumes that partial processing can be the only suitable but nonetheless useful way for better satisfaction of certain requirements. In particular, when dealing with text processing with unlimited vocabulary morphological analysis is a prerequisite in order to better cope with Out of Vocabulary Words (OOW); statistical processing could then be useful for tagging disambiguation. As to syntactic parsing, robust approaches should be adopted in order to allow for structure building even in the case of local failures. Eventually, partial semantic interpretation has to be carried out in order to execute anaphora resolution and a Discourse Model has to be built with a reduced ontology and a limited number of relations and properties. This will be discussed in detail in the Sections into which the Chapter has been subdivided. All these approaches can conspire to produce a linguistically-based solution to the problems of summarization. Our approach is totally opposed to the BOWs (Bag of Words) approach, where words are only conceived as belonging either to the class of content words – and thus become object of analysis – or else be discarded from the text, as member of the set of stopwords also called function or grammatical words. We already commented on this problem in previous chapters above.

1. Techniques for Topic Identification

Techniques for text summarization (hence TS) as described in the literature, fall within two broad categories: either as an extraction process of a list of single sentences that are
meant to express the main contents of a document or set of documents; and as a fully coherent, planned and generated synopsis that compresses the document by topic fusion and paraphrasing. While the former technique is by far representative of the majority of attempts at TS, the latter is hard to attain and is often quoted as a goal for work in progress. There is another technique that relies on template instantiation which we will not comment on in the following, and is usually associated with TREC events.

In this chapter we shall concern ourselves with an intermediate way to produce a summary which is strongly based on the use of NLP techniques: this is intended as a first step towards the final goal, the actual fully generated automatic summary, which will be presented as work in progress in a final section. This proposal is also put forward by K. Spark Jones in her Keynote Address, where she refers to intermediate techniques.

The use of NLP techniques in IR/IE is in our opinion mandatory. Ellen M. Voorhees argues in her paper, the talks and the resulting discussions at SCIE’99 in Rome, in favour of the use of nonlinguistic techniques because,

"Even if done perfectly, linguistic techniques may provide little benefit over appropriate statistical techniques because the statistical techniques implicitly exploit the same information the linguistic techniques make explicit."

On the contrary, we assume that linguistic techniques do not have to be done "perfectly", and the use of shallow parsing is by definition an imperfect linguistic technique which however is successful, and may provide a lot more information and constitutes the only viable technique to satisfy high recall.

We are in full agreement with E. M. Vorhees's comment,

"The question of statistical vs. NLP retrieval systems is miscast however. It is not a question of either one or the other, but rather a question of how accurate an approximation to explicit linguistic processing is required for good retrieval performance. The techniques used by the statistical systems are based on linguistic theory in that they are effective retrieval measures precisely because they capture important aspects of the way natural language is used:
- Stemming is an approximation to morphological processing;
- Finding frequently co-occurring word pairs is an approximation to finding collocations and other compound structures."

As far as we know, in order for the statistics to produce useful linguistic predictions, there must be some such annotated corpora available for training. Currently available annotated corpora are not many and do not cover all facets of the linguistic knowledge we need. Training data are by definition insufficient when the information to be searched is at higher abstract levels of computation. We have already discussed these matters in previous chapters on parsing.

In this chapter we will be concerned with presenting partial, shallow and robust techniques for textual analysis with unbounded vocabulary which are basically related to linguistically rules-based algorithms in the fields of morphological analysis, syntactic analysis, semantic and discourse analysis, and lexical lookup. In Section I, we review a set of NLP techniques proposed in the literature to help IR or better Information Extraction, in the form of Intelligent Sentence Extraction (hence ISE). The proposed NLP techniques may be
regarded as the ancestors from which IR engineering techniques are offsprings, meant to substitute them in order to improve in efficiency over their predecessors. In Section II we will discuss in some detail our morphological analyser for Italian.

As a test to gauge and evaluate the performance of our system for Automatic Summarization (hence AS), we will compare the output of MS Word AS tool with ours: as an evaluation metric we shall compare the number and type of keywords extracted as well as the number of incoherent or non cohesive sentences as measured by the presence of an unbound or dangling anaphoric discourse marker.

We shall start by quoting from K. Spark Jones:

"... what heuristics might be developed for improving readability and coherence of "narratives" made up of discontiguous source document chunks".

As the authors comment, the primary consideration is a generality criterion:
- content characterisation methods should be general enough to apply to any document. This emphasis on domain independence translates into a processing model that requires that we only realize a partially instantiated semantic representation. In this sense, even if we find the approach based on the use of knowledge-bases is important and relevant, we find it hard to satisfy coverage criteria even if in limited domains. At the same time, we don't find the criterion of establishing topic saliency solely on the basis of the numerosity of related concepts or property values, a satisfactory one. As a matter of fact, the paper lacks a discussion of the problem of coherence and cohesion at discourse level which might badly undermine the conceptual descriptive approach. To this end we shall present our approach based on the derivation of Discourse Semantic Relation below.

Requirement for efficient, and scalable, technology necessitates operating from a shallow syntactic base; thus our system tries to circumvent the need for a comprehensive parsing engine.

Of course we agree that, "the cost of avoiding the requirement for a language-aware front end is the complete lack of intelligence at the back end".

However, we assume that Sentence Extraction could be made coherent and cohesive enough in order not to oblige the user to go back to the original text to recover missing discourse relations. The following step, summarization, has to be fully knowledgeable and intelligent in order to produce a coherent and cohesive abstract or summary.

1.1. Non-linguistic vs. Linguistic Techniques in IE

Systems for IR and IE adopt some typical search task, as listed below:
- they tokenize the input text to extract keywords;
- they produce stems to collect keywords under the same stem;
- they exploit some Positional Strategy to individuate best positions for selected keywords;
- they produce phrases on a statistical basis, for extraction purposes.
- they try topic fusion and use some terminological logic for summary generation.
1.1.1. Tokenization and POS Tagging

Text tokenization for keyword extraction is done without any POS tagging which prevents the system to tell NOUNS apart from other wordforms. What most systems do, is to use lists of stopwords to eliminate grammatical wordforms from the final output.

We shall comment on each such techniques individually and propose our NLP-based alternative. We assume that POS tagging is essential in order to focus on the type of lexical category one is interested in: the user might wish to make a search on the basis of Verb categories rather than Nouns. And in case of Nouns, it is important to be able to separate the Noun category into a number of important subcategories, which are semantically relevant – where the last three categories can be just short hand-built lists:

- Proper Nouns;
- Common Nouns;
- Human Names;
- Temporal Nouns;
- Color Nouns;
- Factive Nouns;

In turn, common nouns may be subdivided into two morphologically related categories, simply by noting the presence of multiple tags of these categories,

- Deverbal Nouns;
- Deadjectival Nouns.

In this way a first semantic filtering may be applied to the keywords list, so that the ISE procedure will focus on the right type of content bearing sentences. In order to make the selection more accurate and perspicuous, derivational processes may have to be used, given that many common nouns derive from other more basic ones which are conceptually and semantically related. In our system for Italian, POS tagging uses a tagset of 90 tags which allows us to tell non functional adjectives apart from functional ones. Lexical adjectives and adverbs are then used to contribute topic indexing by means of derivational procedures. This will be discussed in the following section.

1.1.2. Stemming vs. Morphological Decomposition

Many retrieval systems perform stemming, i.e. a type of morphological processing which is however highly error prone. For instance, in the following examples a semantically wrong stem is associated to two separate series of wordforms which do not share the same meaning:

- profess(profess, professed, professes, profession, professionally)
- compet(compet, competence, competition, competitive, competitivity)

The use of NLP techniques is directed in this case to overcome deficiencies of word stem indexing, if we accept the fact the rather intuitive fact that term normalization, i.e. mapping variant spelling or formulation of the same lexical item to a common form is beneficial. However, without a morphological processor which uses linguistic constraints to associate root and lemma to a given wordform, derivationally similar wordforms will always be associated to the smaller available stem. If you consider the case of Italian, where monosyllabic two character words have a one character root, like for instance:

- fa, f, fare "make"

where we listed the wordform "fa"/makes, its root "f"/make, and its lemma "fare"/make, in case we did not want to use morphological processing we will easily end up with a lot stemmed words in the alphabetical list for "f"!
Derivational processes may be very time-consuming as a whole: we only operate derivations when the most frequent topics have been independently detected by the statistical procedure based on root normalization. At the same time, we record on a database all lexical adjectives and adverbs which cannot be captured by matching with nominal derived roots. In case this list is higher than a given threshold - which is computed as 25% of all types in a given document - we proceed by string comparison. We describe the procedure fully below in a separate section – more on this topic in a following section.

1.1.3. Text Segmentation and Positional Criteria

As to text segmentation, we not only divide the text into sentences, but also into paragraphs: the reasons for this second segmentation is related to the decision to include into the final sentence extraction index list all paragraph initial sentences. This decision is based on the intuition that, whereas each paragraph may legally start a new topic, cross-paragraph topic continuations are infrequent. At present we simply add sentences in strategic positions only in case they are part of the final topic identification list, which has been created on keyword frequency criteria, as explained below.

1.1.4. Creation of "Phrases" or Collocates, Polywords or Multiwords

As they are sometimes called, on a statistical basis, this is one of the IR/IE most typical task which can be used to better characterize the document in terms of its informational content. Phrase is used here to refer to any pair of words that co-occur in documents with a frequency higher than a given threshold. In statistically based approaches, phrases longer than two words are ignored, and also the internal structure of the phrase is ignored. As a result, some frequently co-occurring pairs are not phrases at all, and phrases longer than two words in certain documents might be the best approximation and are left out of the list. In addition, since the component words are coindexed with the collocation, in case this is a non-compositional "phrase", it will get a completely incorrect association.

We use Finite State Automata (hence FSA) to find MultiWords from a list of 120 thousand such elements. Such lists taken from what are usually called Gazeteers – which may contain city names classified by their country, but also by their size - may be updated only manually, given the fact that we need to know whether we are dealing with compositional or non-compositional compounds, as the distinction between i. and ii. below clearly shows:

i. the Ministry of Education
ii. the White House

where in i. we would like to use the component words to corefer whereas in ii. we would rather not use them. Multiwords may also be referred as Named Entities and may regard names of companies, institutions, etc. besides geographical names.

1.1.5. Concept Fusion and Topic Identification

There have been various attempts at topic identification and fusion by means of some concept interpretation in terms of a structure concept database like WordNet, which allows the system to establish inferential links between related concepts. The problem in this case is to acquire large enough linguistic material to provide coverage and robustness.

An empirical method is suggested by Hovy E. & C.Y.Lin (1997) where they describe their method of Interpretation using Signatures, which answers the following question: "Can
one automatically find a set of related words than can collectively be fused into a single concept?" (ibid., 22). One of the main problems to be faced by using this method is a way to establish a confident enough measure for the "strength of association" of words with the main concept. We put "strength of association" in inverted commas because it is used as a cover term for a number of important procedures which are heavily linguistically characterized: like for instance the Multi-Word creation described above; but also and foremost the use of the Phrases to be able to establish Functional Dependency Relations (subject-of, object-of, etc.) between concepts, which alone would guarantee the entire coverage of the text or document under analysis.

So eventually, words classification is not just content vs function words: they are also computable as heads vs dependents in some syntactic representation; relation governors in Predicate Argument Structures vs Modifiers; antecedents of pronominal expressions, coreferents or cospecifiers of other nominals expressions, etc.

2. Topic Density Determination

In our system we rely on usual quantitative techniques for extracting keywords to be used as topics. However before Sentence Extraction takes place, we perform a search for Topic Density which we explain in detail below.

As said before, we filter nouns by means of our tagset. Nouns [n] receive a fine-grained tagset which divides them into nh, np, nt, nf, nc, npro, as explained above. Indexed nouns are represented as a triple, Root – Sentence Indices List – Wordform as shown in the table below:

<table>
<thead>
<tr>
<th>Table 36. PHASE I: nouns loci index</th>
</tr>
</thead>
<tbody>
<tr>
<td>wf(infrastruttur-[1, 3, 4, 4, 4, 4, 4, 6, 10, 10, 12, 13, 13, 15, 15, 17, 19, 19, 20, 20, 20, 21, 21, 22, 22, 24, 26, 26, 26, 28, 28, 29, 29, 30, 30, 30, 31, 31, 34, 34, 34, 34, 35, 34, 35, 36, 39, 40, 40, 40, 42, 42, 42, 43, 43, 45, 50, 50, 50, 51, 51, 51, 51, 52, 52, 54, 54, 57, 57, 57, 60, 60, 64, 65, 66, 66, 67, 67, 71, 72, 78, 80, 85]-80-infrastrutture).</td>
</tr>
<tr>
<td>wf(impres-[3, 19, 21, 30, 30, 34, 34, 40, 42, 44, 48, 44, 48, 44, 48, 55, 56, 57, 58, 58, 58, 60, 60, 62, 63, 64, 65, 66, 71, 75, 78, 83, 83, 84, 85]-29-imprese).</td>
</tr>
<tr>
<td>wf(trasport-[7, 10, 17, 18, 33, 34, 50, 50, 59, 59, 59, 60, 61, 62, 62, 67, 69, 69, 81, 82, 83]-18-trasporto).</td>
</tr>
<tr>
<td>wf(sistem-[6, 12, 16, 26, 30, 32, 37, 40, 41, 42, 45, 45, 67, 68, 71]-14-sistemi).</td>
</tr>
<tr>
<td>wf(città-[7, 8, 9, 10, 12, 13, 20, 21, 47, 72, 78, 82]-12-città).</td>
</tr>
<tr>
<td>wf(cost-[29, 31, 34, 37, 37, 37, 39, 50, 51, 51, 51, 80]-11-costi).</td>
</tr>
<tr>
<td>wf(are-[12, 13, 14, 19, 44, 45, 52, 52, 55, 77, 82, 84]-11-area).</td>
</tr>
<tr>
<td>wf(servizi-[18, 18, 29, 35, 51, 63, 69, 74, 81, 82]-10-servizi).</td>
</tr>
<tr>
<td>wf(economie-[21, 21, 21, 30, 31, 32, 34, 36, 37, 51]-9-economie).</td>
</tr>
</tbody>
</table>

Rather than using rank list of most frequent word form and do sentence extraction from that single list, we create Rank List of Sentence Indices according to their frequency as derived from Roots. To create a Rank List of Sentence Indices (RLSI) we merge all indices
and then count their occurrence. Topic Density is determined in terms of sentences containing as many topics as possible as derived from the RLSI. Sentence extraction starts from the most highly ranked index and stops at a given threshold.

Topic Density is determined by the following procedural steps shown in Fig. 4 below:

- Sentence Indices from different Wordforms are merged into one single Root List;
- ON A SECOND PASS into the Tokens file we extract all derivationally definable Wordforms;
- The new set of Sentence Indices is merged into the previous ones on the basis of Root information.

As can be seen from the following table 37, the number of new topical sentences indexed by derivational processes varies remarkably in relation to two variables: genre and text length in terms of tokens.

<table>
<thead>
<tr>
<th>Genre</th>
<th>TOTAL NO. TOKENS</th>
<th>TOTAL NO. TYPES</th>
<th>TOTAL NO. SENTENCES</th>
<th>New Topical SENTENCES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Culture</td>
<td>3493</td>
<td>1274</td>
<td>175</td>
<td>0</td>
</tr>
<tr>
<td>Economics</td>
<td>2028</td>
<td>706</td>
<td>42</td>
<td>23</td>
</tr>
<tr>
<td>Science</td>
<td>3601</td>
<td>1323</td>
<td>84</td>
<td>4</td>
</tr>
<tr>
<td>Politics</td>
<td>5321</td>
<td>1562</td>
<td>166</td>
<td>17</td>
</tr>
<tr>
<td>Burocracy</td>
<td>13815</td>
<td>2831</td>
<td>475</td>
<td>110</td>
</tr>
<tr>
<td>TOTALS</td>
<td>28258</td>
<td>7696</td>
<td>942</td>
<td>154</td>
</tr>
</tbody>
</table>

Figure 4. PHASE II: topic density determination.

Table 37. Increase in Topical Sentences from Derivational Indexing

3. Discourse Analysis
The most important contribution of our system is to Anaphoric and Discourse Relations. Sentences may be discourse dependent or discourse independent: in the former case they need some previous portion of text in order to be fully cohesive and coherent. In the latter case, they simply constitute a semantically complete unit. Anaphors can belong to two different types: discourse deictic pronouns (DDP) and discourse bound sentential markers (DBSM) – more on this topic on a chapter on Discourse in Book 2. They address two types of common problems in ISE tasks: the Missing Sentence Problem and the Dangling Anaphor Problem.

As to the former problem, coherent arguments from the source text appear distorted in the extract due to incoherent sentence selection: e.g. when the sentences S1, S2 and S3 together make a coherent argument, and the extract contains only S2, then the argument is no longer recoverable, hence it becomes invalid. This can be amended by making recourse to the algorithm that checks whether the sentence starts with or contains a DBSM which has anaphoric rather than a cataphoric import: to this end, all conjunctions have been classified as anaphoric - value 1 - or cataphoric - value 0.

The same approach has been taken for indicating the presence of a DDP which can either be a deictic pronoun like "this", "that", or simply a lexically unexpressed subject. In the latter case - only related to languages like Italian, all Romance languages, French excluded, but also Japanese and Chinese - the verbal group has to be detected and in case the sentence starts with such a syntactic constituent it will be tagged as Anaphoric, value 1. In all these cases, we manage to overcome the problem of the Dangling Anaphor: whenever the discourse referents referred by an anaphoric expression in one of the selected sentences are not accessible in the extract due to incohesive sentence selection, we simply add the previous sentence and check whether it is not the case that the newly selected sentence is dependent on the previous one too. Note that this is an overgeneralization of the structural properties of these languages, and that the subject NP may appear in postverbal position.

Building the VP requires the algorithm to use Shallow Parsing as proposed in (Delmonte 1999; Delmonte & Pianta 1999), relatively only to Verb Phrase internal components: they can be either one of the following tag:

- negation, modals, auxiliaries, main verb, adverbs, clitic pronouns

Whenever the VP is built we analyze it in order to compute Sentence Relevance through its Discourse Relation as proposed in previous chapters on Discourse.

More details on the Discourse level will be discussed in the following section, where partial semantic interpretation will be presented and fully discussed with an example.

4. PARTIAL SEMANTIC INTERPRETATION AND SUMMARY CREATION

For a set of sentences to be truly representative of document content, it must provide an exhaustive description of the most relevant entities presented in the document. Even though Information Extraction may not be always perfectly tuned in that sense, we think that up to a certain threshold - the 25%, the automatic extraction process described above can be regarded a sound procedure. Below that threshold, statistically based automatic extraction procedures will cause the set of sentences to be no longer sufficient to allow the reader to grasp the meaning of the topics discussed in the document. Reasons are of two types:
• a quantitative one, which requires that a certain number of sentences be extracted to present an argument with sufficient depth to make it understandable to the reader. Below that threshold, important subtopics semantically related to the main topics will simply be lost;
• a more substantial one, being that automatic statistically based extraction cannot distinguish relevant from non-relevant sentences, even if linguistic intensive procedures are used.

To cope with the problem of summarization we proceed by allowing only the highest scoring keywords to contribute sentence indices for the final extraction process. The result will be a set of locally connected sentences concerned with the Main Topics of a given document or text.

Producing an abstract or a summary requires a topic condensation process made on a conceptual basis of the resulting extract. We then produce Predicate Argument Structures from all the sentences with the help of a Shallow Parser we follow (Delmonte 1999), the output is then passed on to an ATN-like processor where we use subcategorization information to interpret the input and produce a Discourse Model. We use PASs because we believe that what people remember when reading or listening to a long complex text is only PASs of most frequent and most relevant sentences. Generation of a summary is done from the output of the shallow parser, as proposed in (Delmonte and Bianchi 1998; Delmonte 2000). This is described in detail in the following section.

We organized an experiment on Italian texts, using the Automatic Summarizer available under MSWord™1998 and comparing the results with our summaries. We tabulate in Table 38 the number of Extracted sentences respectively with MSWord™ and Getaruns: then in a following column we indicate the number of overlapping sentences. Finally we indicate in two separate columns the number of Discourse Dependent Sentences (hence DDS) over all text span, and the number of DDS in the extracted set.

From the data we can draw a number of interesting conclusions: first of all, the number of DDS is quite high and it amounts to 25% of all sentences. We can also note that there can be remarkable differences depending on genre and number of tokens.

As to the number of extracted sentences in the two systems the number of exceeding sentences in our systems is related to the fact that we add sentences whenever the DDS requires it; so, we end up with a slightly higher number of tokens/sentences than the amount required by the specified percentage. Apart from that, the amount of overlapping sentences is quite low: less than one third of all extracted sentences. In the three final columns we listed DDS for all texts and Dangling DDS in Extracts by MSWORD™. Figures are not particularly high - on the average only 14/15% of all DDS are included in the summaries. However, we would like to stress the fact that the presence of even only one Dangling DDS causes the text to become unreadable!

Apart from solving the main problem of dangling topics, we can now also produce short summaries which are very hard to produce by means of the sentence extraction technique. In fact, whenever we tried to reduce the extraction percentage below the 15% nothing understandable came out apart from the first sentence of the text. Also collecting first sentences for each paragraph did not help at all seen that all the intervening linguistic material was missing. Thus, we assume that for “abstract”-like summaries, generating in the vein that we propose is mandatory.
5. Tagging and Morphological Analysis

In this section we argue in favour of an integration between statistically and syntactically based parsing by presenting data from a study of a 500,000 word corpus of Italian. Most papers present approaches on tagging which are statistically based. None of the statistically based analyses, however, produce an accuracy level comparable to the one obtained by means of linguistic rules. Of course their data are strictly referred to English. As to Italian, we argue that purely statistically based approaches are inefficient basically due to great sparsity of tag distribution – 50% or less of unambiguous tags when punctuation is subtracted from the total count. In addition, the level of homography is also very high: readings per word are 1.7 compared to 1.07 computed for English by Brants T. & C. Samuelsson with a similar tagset. Our tagsets for Italian and English have already been presented in previous chapters.

Most of the published papers on the subject deal with English due to the availability of tagged corpora for training and testing. However, English is not a good representative of European languages in that it can be regarded a "morphologically poor" language, whereas the remaining languages are "rich" or even "extremely rich" in morphology. This amounts to a lot of differences in processing: as can be easily surmised the first difference is in the number of different wordforms each lemma can produce; then a second important difference lies in the level of homography of each wordform. In the following, we shall present general data on a corpus of Italian and then discuss our STD (Syntactic Tag Disambiguator).

In our model we assume that lexicon and morphotactics/phonotactics should be kept distinct; in addition, we assume that morphological rules should address morpheme-like portions of words rather than single graphemes.

In KIMMO-like systems, morphological parsing is performed with a scheme whereby the morphotactics is folded in with lexical access. The lexicon itself is organized as a letter tree which is traversed while spelling the input string into each single orthographic character or grapheme. We also tried to use such a procedure in a previous morphological analyzer which was intended to serve as a tagger for a system of speech synthesis with unlimited vocabulary (see Delmonte, Mian, Tisato 1985). We found that such a system had no flexibility.

---

Table 38. Extracted Sentences and Discourse Dependency

<table>
<thead>
<tr>
<th>No. Extracted Sentences</th>
<th>MS WORD™</th>
<th>GETARUNS</th>
<th>OVER-LAP</th>
<th>DDS for all text</th>
<th>Dangling DDS in Extracts</th>
<th>% of all DDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Culture</td>
<td>37</td>
<td>41</td>
<td>12</td>
<td>90</td>
<td>7</td>
<td>7.8%</td>
</tr>
<tr>
<td>Economics</td>
<td>11</td>
<td>12</td>
<td>5</td>
<td>17</td>
<td>4</td>
<td>23%</td>
</tr>
<tr>
<td>Science</td>
<td>20</td>
<td>22</td>
<td>8</td>
<td>38</td>
<td>6</td>
<td>15%</td>
</tr>
<tr>
<td>Politics</td>
<td>34</td>
<td>30</td>
<td>7</td>
<td>68</td>
<td>10</td>
<td>14%</td>
</tr>
<tr>
<td>Burocracy</td>
<td>67</td>
<td>70</td>
<td>21</td>
<td>56</td>
<td>8</td>
<td>14%</td>
</tr>
<tr>
<td>Legal</td>
<td>82</td>
<td>118</td>
<td>30</td>
<td>104</td>
<td>18</td>
<td>17%</td>
</tr>
<tr>
<td>TOTALS</td>
<td>251</td>
<td>303</td>
<td>83</td>
<td>373</td>
<td>53</td>
<td>14%</td>
</tr>
</tbody>
</table>
at all in that the peculiarities of Italian morphology were such that there was no way to get a
general enough representation of their behaviour.

The current system decomposes the input string from right to left, trying to strip off a
portion of word corresponding to a possible suffix; in order to do this, at each step of the
decomposition process the sequence of characters is recomposed into what might be treated
as an affix by the system and the remainder is then checked in the root dictionary for a match.
This procedure is then repeated recursively up to a given threshold determined by word length
and maximum length of a suffix in Italian. In order to find other substrings that can match a
possible suffix and then perform dictionary lookup with the remainder for a match. At the end
of each successful search, combinatorial rules check to see whether the suffix can be
combined with the root.

5.1. Bound vs. Free Morphemes

Our approach is motivated by psychological and linguistic reasons, besides processing
ones. The system may differentiate accessibility for function words vs content words, so that
we can simulate short term memory effects which we assume are relevant in trying to achieve
psychological plausibility. In case we adopted a letter tree there would be no possibility to
differentiate lexical access since all words in the dictionary are available at the same time.
Italian not belonging to the type of agglutinative languages like Finnish, has two different sets
of morphemes.

We follow linguistic theories which assume that morphemes can be divided up into two
different classes: bound morphemes or affixes cannot be freely augmented and constitute a
closed list; on the contrary free morphemes mainly constituted by roots can be increased
freely and constitute an open list. In order to ascertain whether we have decomposed a
possible affix we access the closed list of bound morphemes which is very short for Italian
and amounts to approximately 1,000 different affixes including enclitic pronouns. Only in
case we find one such morpheme the main dictionary is accessed. At this point of the
analysis, we apply to the pair root-affix a set of constraints which are based on paradigmatic
and syntactic information. To this end we classified nouns, verbs and adjectives as belonging
to one class or another according to their inflectional behaviour. Here below we include a
small excerpts from the lexicon where we show for each root morpheme, the lemma and the
associated set of linguistic information:

  l(altr, altro, ['agg:alt']).     other
  l(stess, stesso, ['agg:an']).   same
  l(tal, tale, ['agg:dim']).     such
  l(convenzional, convenzionale, ['agg:e']). conventional
  l(identic, identico, ['agg:ico']). identical
  l(poc, poco, ['agg:ind']).     little
  l(qual, quale, ['agg:int']).   which
  l(comprensiv, comprensivo, ['agg:o']). comprehensive
  l(spesso, spesso, ['avv:ti']).  often
  l(malgrado, malgrado, ['cong:av']). though
  l(cas, casa, ['s:a:f']).     house
We also keep in short term memory all function words of Italian, amounting approximately to 2,000 entries, so that they may be accessed first, before attempting any decomposition of the input string. Since they constitute the most frequently used words of the language they should be readily available to the system. The same situation is covered for frequent collocations and other types of polywords which are all accessed first, before attempting any dictionary lookup.

In order to ascertain whether we should work with a fixed wordform list as do most researchers in the English language, we decided to study in detail the distribution of wordforms and lemmata in a statistically significant corpus.

5.2. Lemmata and Wordforms, Stems and Affixes

Now we shall consider the ratio lemmata/wordform as indicating the morphological richness of the language: suppose now that in our corpora more than half of all wordforms or types uniquely individuate its lemma and viceversa, we might conclude that even though Italian has a potentially rich morphology it uses it in a poor manner.

Data commented here below are taken from the morphological analysis of a corpus of half a million word corpus of Italian. From the computation of lemmata we ended up with the following data:

Total number of lemmata is 24666. Non-rich lemmata constitute a big percentage of the total number of lemmata: lemmata with only one wordform associated amount to 17464, which corresponds to 70% of the overall number of lemmata. Here below is the count for lemmata with two, three or four wordforms associated:

Ambiguity = 2  Number of lemmata 4354
Ambiguity = 3  Number of lemmata 962
Ambiguity = 4  Number of lemmata 879

Finally we end up with 1007 lemmata with more than four types associated, the great majority of which are verbs. However when we look down in the rank list starting from lemmata with 8 types associated, the number of past participles/adjectives increases until they become the majority of lemmata. It is interesting to see the type of derivation that have been produced by rules automatically by the generation process in the following examples:

\[
\text{l(lungo, 8, \{lunga, lunghe, lunghi, lunghissima, lunghissime, lunghissimi, lunghissimo, lungo\}). /\text{long}}
\]
The rank list has the two auxiliary verbs have (avere) and be (essere) at the top, respectively with 50 and 48 word forms associated. We may note that "avere" has 13 cliticized forms and that "essere" has only 10 such forms.

As said above, our system decomposes the input string from right to left, trying to strip off a portion of word corresponding to a possible suffix. As an example consider now the word "fatti"/facts; possible interpretations for this word are as follows:

1. fatt - noun, [pred=fatt+o, gen=masc, num=plur]
2. fatti - past-participle, [pred=fatt+o, gen=masc, num=plur]
3. fatti - verb, [pred=f+are, mood=part, tense=pass, gen=m, num=p]
4. fatti - verb, [pred=f+are, vform=fa', encl=ti, case=dat, mood=imper, tense=pres,pers=2, num=sing]

The following is the actual analysis of “fatti / facts, do it (to) yourself, made” performed by the system:

[cat=ppas, pred=fatt+o, gen=m, num=p]
[cat=nome, type=com, pred=fatt+o, gen=m, num=p]
The first match is tried with a fail in the list of function words; the second match is tried in the dictionary whether there is some invariable word matching the whole string "fatti", again a fail.

After the first character has been stripped off, we are left with two substrings 'i' and 'fatt': 'i' is a legal suffix and the information it carries is made available locally. In fact, this inflectional suffix can be attached to any category, be it a verb, a noun or an adjective.

However the hardest interpretation to compute is the one associated with the meaning “fa’ – ti” / make to you, roughly, imperative form of the verb “fare” followed by an enclitic pronoun in the dative case. In order to achieve this interpretation the clitic must be stripped off thus obtaining the two following substrings “fat” and “ti”. “Ti” is correctly computed as belonging to the subset of enclitics that can undergo consonant doubling – non all enclitic pronouns do, e.g. “vi” (to you plural), “gli” (to him) “si” (-self). An attempt is made to strip off a consonant which is the same as the one starting the enclitic and if it succeeds the rest of the string is passed to the morphological analyzer which will recursively try to find a legal decomposition of the verb “fa”. This will be successful and the result of the analysis will be “f-a’ “ the root “f” for the verb “fare”/make, the suffix “a”/imperative second person singular with the addition of an apostrophe for the phenomenon of apocope. Notice that the final substring “ti” is very common in Italian and is present in all past participle plural masculine verbs belonging to first conjugation. However “fatti” has to be distinguished from “fati” being the plural of the noun “fato”/destiny: there are a couple such pairs “statti/stati”, “datti/dati” etc.

Consider now the problem of generating the analysis of new legal words which do not belong to the current dictionary and need to be generated by derivational rules. We make use of prefixes and suffixes which are matched against the appropriate portion of the input string, left and right respectively. In case of successful match, the remaining portion of input string is analysed as possible root. In the 6,000 OVW, prefixed words covered 50% of the total, roughly 30% was covered by suffixed words. The remaining 20% is composed of parasynthetic words: they require the system to strip a derivational suffix and a prefix from the same input string, as for instance in "neotrasformismo" /neotransformism. After these two operations have been successfully carried out, the root may be searched for.

We follow a strategy by which in case of lack of match of the root morpheme, and all possible stripping operations have been tried, the partial results may still be used to issue the appropriate message to the user.
6. Tagset and Ambiguity Classes

The current system includes a syntactic shallow parser and a ATN-like grammatical function assigner that automatically classifies previously manually verified tagged corpora. In a preliminary experiment we made with automatic tagger, we obtained 99.97% accuracy in the training set and 96.03% in the test set using combined approaches: data derived from statistical tagging is well below 95% even when referred to the training set, and the same applies to syntactic tagging. As to the shallow parser, we report on a first preliminary experiment on a manually verified subset made of 10,000 words.

We assume, together with P. Tapanainen and A. Voutilainen that POS tagging is essentially a syntactically-based phenomenon and that by cleverly coupling stochastic and linguistic processing one should be able to remedy some if not all of the drawbacks usually associated with the two approaches, when used in isolation. However, as will be shown in detail in the following section, rather than using FSA we use an RTN both for training and for parsing. As to the statistical part, we don't use HMMs but only conditional probabilities on the basis of trigram information as discussed below.

Syntactic shallow parsing is accomplished in our case by using the same RTN we use for tag disambiguation. It is made up of 1700 arcs and 24 nets, and we use it in a non-recursive way, as explained below. Data for the construction of the RTN were derived from the manual annotation of our treebank of 320,000 tokens as discussed in a chapter above, which is then used as test set. Frequency of occurrence associated to each rewrite rule is used as organizing criteria in the ordering of the arcs contained in each node of each net.

6.1. Statistical Vs. Syntactic Disambiguation

Rather than flattening the Phrase Structure Grammar as Pereira F. and R. Wright suggest in their shift-reduce algorithm, we only check for reachability in nonterminal symbols. So, even though the formal structure of RTN is recursive, the disambiguating algorithm does not use recursive calls and all computation is flattened down to one level, that of tags corresponding to preterminals in the RTN. The STD can be defined as a slightly augmented finite state transducer which works at a single level of computation and has access to higher level information when needed. Details of the implementations will be presented in following sections.

Our syntactic tag disambiguator (STD) is the final modul of our syntactic tagger of Italian. Input to the SSD is the complete and redundant output of the morphological analyser and lemmatizer, IMORTALE (Italian Morphological Tagger and Lemmatizer). IMORTALE finds all possible and legal tags for the word/token under analysis on the basis of morphological generation from a root dictionary of Italian made up of 80,000 entries and a dictionary of invariant words - function words, polywords, names and surnames, abbreviations etc. - of over 12,000 entries.

As commented by Brill, the application of stochastic techniques in automatic part-of-speech tagging is particularly appealing given the ease with which the necessary statistics can be automatically acquired and the fact that very little handcrafted knowledge need to be built
into the system. However both probabilistic models and Brill’s algorithm need a large tagged corpus where to derive most likely tagging information. It is a well known fact that in lack of sufficient training data, sparsity in the probabilistic matrix will cause many bigrams or trigrams to be insufficiently characterized and prone to generate wrong hypotheses. This in turn will introduce errors in the tagging prediction procedure. Italian is a language which has not yet made available to the scientific community such large corpus, our treebank being just a small effort. In lack of such an important basic resource, there are two possibilities:

- manually building it by yourself;
- using some automatic learning procedure which in our case corresponds to the use of a syntactic tagger.

We have been working on such a corpus of Italian with the aim of achieving the above-mentioned final goal, without having to manually build it. The algorithm that we will present in this paper is partly based on stochastic techniques – thus following Church(1988, 1992), Garside et al.(1987); this is however coupled with linguistic processing by means of a CF grammar of Italian formalized as an RTN, which filters it. Statistics is usefully integrated into the syntactic disambiguator in order to reduce recursivity and allow for better predictions.

After a first fully automatic phase, we started building BIASES which are commented below and are used to correct most common errors. This second phase has taken us 3 man/months work to complete. The final result is a 95% accuracy analysis on the whole corpus. The final output has then been used to collect trigrams for the statistical tagger. Statistics and syntactic disambiguation have then been fully integrated in order to reduce recursivity and allow for better predictions and higher efficiency. Fully stochastic taggers, in case no large tagged corpora are available, may make use of SVMs. However, SVMs show some of the disadvantages present in other classifiers: they lack perspicuity, in the sense that will be better explained below, basically imposing that the data related to tags are all treated on a par. Their classificatory power is derived from the features they use to build the model. They don’t allow for biases to be implemented – biases are very similar to patches in Brill’s tagger – they are inherently incapable of capturing higher level dependencies present in natural language, and are always prone at generating wrong interpretations, i.e. accuracy never goes higher than 96-97%. Of course it is a good statistical result, but a poor linguistic result, seen the premises, i.e. the need to use tagging information for further syntactic processing.

We studied our training corpus in order to ascertain what level of ambiguity was present and where, seen that our corpus is made up of sub-corpora from different domains and genres. Our tagset, presented above, is made up of 91 tags thus subdivided: 10 for punctuation; 4 for abbreviations, titles, dates, numbers; 19 for verbs including three syntactic types of subcategorization – transitive, intransitives, copulatives – and tensed cliticized verbs; 47 for function closed class words subdivided into 18 for pronouns, clitics, determiners and quantifiers - 18 for adverbs conjuctions and prepositions - 11 for auxiliaries and modals; 11 for adjectives and nouns, including special labels for colour nouns, time nouns, factive nouns, proper nouns, person names - this list includes special labels for guessed proper nouns, foreign words and misspelled words. Twenty categories from the general tagset never occur single, so they had to be converted into distributionally equivalent ones, in the statistical table.
We refer to the tagset of LOB corpus (see Johansson et al. 1986) which uses 157 tags for English: this higher number of tags is motivated by the fact that they included in their set special tags for plural forms, genitive forms both for nouns and verbs, and with adjectives also for comparative and superlative forms. In case we eliminate these duplicate forms the total number of tags is 107.

The general criteria we adopted for including or not a tag in our tagset obeys the following principles:

- a tag must be unambiguously associated to a wordform or class of wordforms;
- a tag must be motivated by unique distributional properties, i.e. must be in complementary distribution with other similar tags: for instance, we collapsed the tag for common nouns [n] with a number of "allotags": [nt] for temporal nouns, [nf] for factive nouns, [np] for proper nouns (geographical and others), [nh] for person names: this subdivision is relevant both for semantic and for syntactic reasons;
- don't use new tags when there is no need to: tagsets for English usually include 3rd person verbs and plural for nouns. We don't see any reason to introduce such morphologically-based tags, which in the case of Italian or other such languages would make the tagset explode. In addition, agreement cannot be used for disambiguation purposes in Italian at such a low level of analysis – i.e. without syntactic constituents being previously built.

As in Reimer et al. 1997, also our implementation allows for two sorts of biasing of starting values:

- a given word may be assigned to a single tag owing to domain, genre, corpora etc. preferences – see for instance the case of “la/the_sing_fem” where we omit the tag for noun (musical note A);
- a trigram can be assigned to favoured tags, on the basis of syntactic processing, as discussed in detail in the section below.

These biases may be specified either as sets or as set membership and are used both in the statistical and in the syntactic procedure. Total number of syntactic biases present in our algorithm is 340.
### Conditional Probability

#### Number of Occurrences

<table>
<thead>
<tr>
<th>Maximum Potential Number of Occurrences</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Digram Structures</th>
<th>Trigram Structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P(ij) = P(i) \times P(j) )</td>
<td>( P(i,j,k) = P(j) \times P(i,k) )</td>
</tr>
</tbody>
</table>

#### Tag Digram Sequences

- frequency of the tag sequence \( i \) followed by \( j \)
- frequency of the tag \( i \)

#### Tag Sequence Probability

- **Digram frequency of Candidate Tag Sequence**
- **Summed Digram Frequencies of All Candidate Tag Sequences**

| Ps = \sum_{(T_l T_r) \in C} P(T | C) \times P(T | T_l T_r) | P((T_l T_r) | C) |

In a similar way to the reductionist statistical approach proposed by (Vorces, 1999; Delmonte 1999) we induce the best tag from the set of available tags in the context of an unambiguous tag by recursively calling all contextually allowable combinations, from where we select the ones corresponding to the current ambiguity class; we then compute probabilities, according to the formula reported below. In case some data are not available and the system is working in the semi-automatic mode, a friendly interface is activated and the user may provide the required unambiguous tag by looking at the sentence that contains its corresponding word/s. Trigram probabilities are combined as suggested in Brants et al.(1995) and reported here,

\[
P(T | C) = \sum_{(T_l T_r) \in C} P(T | T_l T_r) \times P((T_l T_r) | C)
\]

where \( T \) denotes a candidate tag of the current word, \( T_l \) denotes a candidate tag of the immediate left neighbour, and \( T_r \) denotes a candidate tag of the immediate right neighbour. \( C \) is the set of ordered pairs \( (T_l T_r) \) drawn from the set of candidate tags of the immediate neighbours. \( P(T | T_l T_r) \) is the symmetric trigram probability. We remove low-probability candidate tags by simply ignoring the tail of the ordered output list where candidates are ranked in descending order, on the basis of a fixed threshold. As commented above, in case no data are available, rather than computing zero probability we let the current procedure fail - the algorithm is implemented in Prolog - and activate the user interface for information. In the Culture corpus made up of 28,000 tokens we were alerted 85 times.

The disambiguator is made up of two separate modules: the Probabilistic Transition Table for local tag disambiguation; the syntactic transition network where the learning phase is situated. As stated in the literature on the topic (Brants et al. 1995; Voutilainen et al. 1993), standard Markov models require at least trigrams to capture cooccurrence probabilities in an
efficient way. Besides, it is a fact that what stochastic models are actually intended to capture is nothing more than syntactic regularities of a given language. However, while working at local adjacency level, discontinuities cause finite-state processing to halt and recursivity is called in to allow for choices among different tags when ambiguous. Finite-state automata may be slightly augmented in order to take into account dependencies intervening between major constituents and at clause level.

In Cutting et al., 1992, the authors explain that they chose HMMs which can be trained on a small amount of manually encoded data, rather than simple Markov models which require a big enough corpus of tagged training data to achieve adequate probability estimation. However HMMs are then used only as a first-order model due to the sparsity of data and to the reasonable fact that tags (or their associated ambiguity classes) are actually observable and discrete. In speech recognition, HMMs are needed to hypothesize the existence of a given token in the observed set of data input B which can in fact only approximate the corresponding token in the A set of data from the training set.

As the authors comment, in the training phase, HMModeling restarts at unambiguous points where the model can be broken down: in language models, where wordforms are used, the perplexity of the model is computed on the basis of the level of ambiguous paths that can be followed by the algorithm in a given state. In this case branching stops at sentence end. However in tagging it may stop every time an unambiguous token is met. In other words the Model is dependent on the distribution and number of unambiguous tokens in a given text sample.

So for simplicity we assume a zero hypothesis for the difference of distribution of unambiguous vs ambiguous tags and we use the former distribution to disambiguate the latter. In order to take advantage of automatic tagging and also of the fact that some bigram may never occur, given the fact that we may assign zero probability to unattested bigrams, we set up the following algorithm:

- we extract all possible tags and establish ambiguity classes (ACs) where we give special status to all punctuation tags and all unambiguous tags;
- we built all possible bigrams of ACs, i.e. all pairs of adjacent ACs;
- we then dynamically compute a probability matrix from number of occurrences of unambiguous bigrams;
- we use this matrix to make choices in left- or right-ambiguous bigrams, i.e. bigrams which have ambiguous AC on one side of an unambiguous one;

- we then recompute our starting probability matrix and make a third pass to disambiguate fully ambiguous bigrams, in this case using Biases.

A similar stategy may be adopted for right-ambiguous or left-ambiguous bigrams which are however closed by a following unambiguous tag: we call these closed trigrams, i.e. trigrams which start and end with an unambiguous tag. The center ambiguous tag may be easily disambiguated without the use of probability measures.

As in (Kupiec above), also our implementation allows for two sorts of biasing of starting values: ambiguity classes can be annotated with favoured tags; and states can be annotated with favoured transitions. These biases may be specified either as sets or as set membership.
And since we are always dealing with sets of tags, a lot of uncertainty may be obviated in case one takes into account cardinality of ACs: it seems sensible to suppose that ACs with a low cardinality cost less in terms of computing time that ACs with high cardinality.

Ambiguous tags are fairly evenly distributed among the subcorpora as can be seen in Table 39a, where we tabulate ambiguity classes starting from unambiguous then cardinality 2 or twice ambiguous tokens, ending with 9 times ambiguity figuring only once. As can be seen from Table 39b and 39c, there are two domains which have a higher level of ambiguity than the remaining domains which are fairly evenly behaved, and these are Politics and Science.

### Table 39a. Ambiguity Classes: Types

<table>
<thead>
<tr>
<th>Classes</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>Tot</th>
</tr>
</thead>
<tbody>
<tr>
<td>Types</td>
<td>76</td>
<td>144</td>
<td>128</td>
<td>76</td>
<td>47</td>
<td>16</td>
<td>14</td>
<td>4</td>
<td>1</td>
<td>506</td>
</tr>
</tbody>
</table>

### Table 39b. Ambiguity Classes: Tokens

<table>
<thead>
<tr>
<th></th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Culture</td>
<td>734</td>
<td>777</td>
<td>174</td>
<td>49</td>
<td>107</td>
<td>12</td>
</tr>
<tr>
<td>Politics</td>
<td>1,323</td>
<td>1456</td>
<td>445</td>
<td>92</td>
<td>126</td>
<td>0</td>
</tr>
<tr>
<td>S.Admin.</td>
<td>1,284</td>
<td>1723</td>
<td>527</td>
<td>137</td>
<td>102</td>
<td>0</td>
</tr>
<tr>
<td>Finance</td>
<td>8,261</td>
<td>8138</td>
<td>1757</td>
<td>805</td>
<td>882</td>
<td>0</td>
</tr>
<tr>
<td>Science</td>
<td>1,919</td>
<td>2532</td>
<td>434</td>
<td>277</td>
<td>289</td>
<td>0</td>
</tr>
</tbody>
</table>

### Table 39c. Ambiguity Classes: Tokens cont

<table>
<thead>
<tr>
<th></th>
<th>2</th>
<th>% Tot</th>
<th>3</th>
<th>%Tot</th>
</tr>
</thead>
<tbody>
<tr>
<td>Culture</td>
<td>7,450</td>
<td>26.6</td>
<td>2,975</td>
<td>10.6</td>
</tr>
<tr>
<td>Politics</td>
<td>19,652</td>
<td>33.7</td>
<td>6,602</td>
<td>11.34</td>
</tr>
<tr>
<td>S.Admin.</td>
<td>12,106</td>
<td>26.6</td>
<td>4,795</td>
<td>10.76</td>
</tr>
<tr>
<td>Finance</td>
<td>86,763</td>
<td>26.4</td>
<td>35,421</td>
<td>10.78</td>
</tr>
<tr>
<td>Science</td>
<td>25,770</td>
<td>32.2</td>
<td>9,656</td>
<td>12.08</td>
</tr>
</tbody>
</table>

As can be seen from Table 39a, the number of ambiguous tags is very high and even though it decreases dramatically already with class 4 it might still constitute a serious obstacle to approaches like the ones adopted by advocates for rule-based or constraint based disambiguation. Of course what the Table tells us is that the total number of occurrences of tags belonging to a given AC is very high: however, the class might be represented by a very small number of types. In that case it would still be very convenient to manually encode biases for all types.

Unfortunately this is not the case, and is clearly shown in the following table, Table 40. We discovered that the number of possible bigrams is very high and changes a lot from one sub-corpus to another. We also discovered that the great majority of bigrams is made up by hapax-legomena, i.e. bigrams with frequency of occurrence equal to or below 3.

### Table 40. Bigrams - Types
In the following table we list all ambiguous paths according to their length and their distribution in our corpora. We define as ambiguous paths those n-gram sequences delimited by a non-ambiguous tag: in turn this tag may correspond to a punctuation mark or simply to any non-ambiguous tagged word. Tokens and Types in this table define the whole set of ambiguous n-grams thus defined. The distribution of types is very even for all sub-corpora, exception made for Finance, which has a much higher number of tokens in comparison with the other corpora. Ambiguous Paths of length equal to 5 constitutes by far the greatest subset.

Table 41a. N-Grams Ambiguous Path Distribution

<table>
<thead>
<tr>
<th></th>
<th>Tokens</th>
<th>Types</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Culture</td>
<td>9,889</td>
<td>7,659</td>
<td>1,893</td>
<td>2,824</td>
<td>1,501</td>
<td>753</td>
</tr>
<tr>
<td>Politics</td>
<td>22,061</td>
<td>12,657</td>
<td>1,677</td>
<td>4,179</td>
<td>3,240</td>
<td>1,811</td>
</tr>
<tr>
<td>S.Admin.</td>
<td>17,105</td>
<td>9,488</td>
<td>1,943</td>
<td>3,354</td>
<td>2,042</td>
<td>1,111</td>
</tr>
<tr>
<td>Finance</td>
<td>119,820</td>
<td>40,645</td>
<td>6,000</td>
<td>6,548</td>
<td>13,546</td>
<td>7,484</td>
</tr>
<tr>
<td>Science</td>
<td>29,500</td>
<td>18,802</td>
<td>2,680</td>
<td>6,077</td>
<td>4,376</td>
<td>2,452</td>
</tr>
</tbody>
</table>

Table 41b. N-Grams Ambiguous Path Distribution continued

<table>
<thead>
<tr>
<th></th>
<th>Tokens</th>
<th>Types</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
</tr>
</thead>
<tbody>
<tr>
<td>Culture</td>
<td>9,889</td>
<td>7,659</td>
<td>345</td>
<td>180</td>
<td>80</td>
<td>37</td>
<td>26</td>
<td>11</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>Politics</td>
<td>22,061</td>
<td>12,657</td>
<td>833</td>
<td>446</td>
<td>235</td>
<td>121</td>
<td>64</td>
<td>18</td>
<td>17</td>
<td>10</td>
</tr>
<tr>
<td>S/Admin.</td>
<td>17,105</td>
<td>9,488</td>
<td>543</td>
<td>283</td>
<td>118</td>
<td>41</td>
<td>24</td>
<td>9</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>Finance</td>
<td>119,820</td>
<td>40,645</td>
<td>3728</td>
<td>1756</td>
<td>822</td>
<td>418</td>
<td>172</td>
<td>76</td>
<td>42</td>
<td>21</td>
</tr>
<tr>
<td>Science</td>
<td>29,500</td>
<td>18,802</td>
<td>1332</td>
<td>693</td>
<td>342</td>
<td>155</td>
<td>89</td>
<td>52</td>
<td>22</td>
<td>17</td>
</tr>
</tbody>
</table>

7. Misspelled Words and Spelling Correction

The main problem with spelling-checkers and spelling-correctors seems to be constituted by the number of legal word alternatives the system is able to produce, and that would still need human intervention (see Oflazer, 1996). In a rule-governed generative framework like ours the problem is constituted by the quantity of linguistic information associated to each morpheme. Overgeneration which is a typical problem of all kinds of systems, should be kept under control by a substantial set of linguistic constraints in order to prevent generating a wrong guess. Consider now the problem constituted by the recognition of a New Vocabulary Entry (hence OVW). Differently from a language like English, Italian has a verb paradigm...
with 60 different wordforms each. As far as our Main Root Dictionary is concerned, we count 15,000 different verb root morphemes. By multiplying wordforms and verb roots we come very close to 1 million different wordforms. In turn, each verb may take a certain number of enclitic pronouns according to subcategorization. Transitive verbs may thus end up with a total number of 400 different wordforms each, and transitive verbs constitute the great majority of verbs. A similar situation may be surmised for nouns. Nouns may undergo alteration by evaluative suffixation: there at least 20 different evaluative suffixes which are commonly used by Italian speakers which must be added to singular and plural forms for nouns that may undergo this kind of derivational process. In our main dictionary we count approximately 75,000 noun root morphemes. Again by multiplying root morphemes and suffixes we get over 1,500,000 possible legal wordforms.

However, even though these figures constitute possible legal wordforms of the Italian language, we would like to take into account real data from corpora: the data we are interested in regard the morphological richness of the corpora we analysed in terms of types associated to each lemma. The total number of lemmata derived from the 36,578 types is 24,666, thus subdivided:

<table>
<thead>
<tr>
<th>Ambiguity</th>
<th>Lemmata</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>17464</td>
</tr>
<tr>
<td>2</td>
<td>4354</td>
</tr>
<tr>
<td>3</td>
<td>962</td>
</tr>
<tr>
<td>4</td>
<td>879</td>
</tr>
</tbody>
</table>

As the data above show, we found that lemmata associated with a single type constitute the great majority. In particular lemmata with ambiguity 2 are as a whole 4354x2=8708; lemmata with ambiguity 3 are 962x3=2886; and eventually lemmata with ambiguity 4 are 879x4=3516.

Consider now the problem of generating new legal words which do not belong to the current dictionary and need to be generated by derivational rules. We make use of prefixes and suffixes which are matched against the appropriate portion of the input string, left and right respectively. In case of successful match, the remaining portion of input string is analysed as possible root. In the 6,000 OVW, prefixed words covered 50% of the total, roughly 30% was covered by suffixed words. The remaining 20% is composed of parasynthetic words: they require the system to strip a derivational suffix and a prefix from the same input string, as for instance in "neotransformismo" /neotransformism. After these two operations have been successfully carried out, the root may be searched for.

We follow a strategy by which in case of lack of match of the root morpheme, and all possible stripping operations have been tried, the partial results may still be used to issue the appropriate message to the user.

Misspelled words constitute a subtype of unknown words which are not computable as OVW. In more detail, we compute words separately according to whether they are: class 1. lowercase letter words or class 2. words with at least an initial upper case letter. Proper Nouns is a default class after a word has not been computed as possible OVW and belongs to class 2. Class 1 words are computed either as misspelled words or as foreign words. In particular, in order for the system to recognize a string as misspelled word, it should be able to strip some affixes of a certain length. The class of Proper Nouns is a default value associated to capitalized words non interpretable by the system.
The algorithm then checks whether syllable structure of the remaining portion of the input string is compatible with the data of Italian. In order to check syllable structure we built a database made of syllables derived from 40,000 different types taken from our corpora and decomposed into legal syllables. We associated to each syllable the position/s in the word and its frequency of occurrence. We ended up with 2,000 major syllables which occur with a frequency value higher than 1. We assume that this approach is very much like positional binary n-gram arrays, except that we are using phonological and graphemic rules to detect correct letter sequences. When trying to decompose the input string of the unknown word, syllables are checked according to all possible combinations in the order found and frequencies are associated to each syllable thus derived. Then the best match is chosen and the mistaken sequence of orthographic characters individuated (see also Solak & Oflazer). In the current implementation we let the system propose the possible correction to the user in a dialogue which allows him also to overwrite it. The result of the interaction is then stored on an internal memory storage and may be reused by the system automatically on the next occurrence of the same error. We found that some errors occur quite frequently and storing the pair mistaken word-corrected word is certainly beneficial.

### 7.1. Spelling Correction and Syllable Structure

As explained above, in order for the system to come up with some hypothesis for an OVW, it must at first find some substring belonging to the language. Rather than randomly trying to generate new word hypothesis, we use morphological and phonological information to direct the correction at a certain substring. The corrector segments the input string into a sequence of syllables and associates positions and frequency values to each syllable. During this process, automatic corrections may be generated. We report here below the list of the first 20 most frequent Italian syllable as taken from our corpus of 100,000 syllables: they have been collected in phonemic transcription, by position so that for each syllable it will be possible to order them accordingly.

**Table 42. First 20 syllable frequency by position in the word**

<table>
<thead>
<tr>
<th>Syll 1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>TOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>RE</td>
<td>268</td>
<td>130</td>
<td>981</td>
<td>1102</td>
<td>404</td>
<td>45</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TO</td>
<td>24</td>
<td>412</td>
<td>1007</td>
<td>915</td>
<td>418</td>
<td>68</td>
<td>14</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TA</td>
<td>78</td>
<td>514</td>
<td>1163</td>
<td>743</td>
<td>224</td>
<td>37</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>RI</td>
<td>1007</td>
<td>449</td>
<td>405</td>
<td>219</td>
<td>75</td>
<td>32</td>
<td>5</td>
<td>4</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TI</td>
<td>59</td>
<td>574</td>
<td>693</td>
<td>445</td>
<td>151</td>
<td>29</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>NO</td>
<td>60</td>
<td>209</td>
<td>500</td>
<td>658</td>
<td>279</td>
<td>41</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TE</td>
<td>117</td>
<td>363</td>
<td>571</td>
<td>487</td>
<td>143</td>
<td>16</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>DI</td>
<td>697</td>
<td>507</td>
<td>249</td>
<td>63</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>NE</td>
<td>54</td>
<td>165</td>
<td>406</td>
<td>381</td>
<td>313</td>
<td>132</td>
<td>26</td>
<td>8</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>KA</td>
<td>366</td>
<td>350</td>
<td>336</td>
<td>277</td>
<td>93</td>
<td>20</td>
<td>8</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VA</td>
<td>115</td>
<td>194</td>
<td>533</td>
<td>462</td>
<td>101</td>
<td>9</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>RA</td>
<td>86</td>
<td>274</td>
<td>604</td>
<td>307</td>
<td>71</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SI</td>
<td>102</td>
<td>340</td>
<td>420</td>
<td>320</td>
<td>115</td>
<td>20</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Syllables are internally represented as terms with their orthographic transcription, with a rank index indicating their absolute frequency relevance and an associated list of position markers in the word again listed by frequency relevance.

re-1-[4, 3, 5, 1, 2, 6, 7, 8],
to-2-[3, 4, 2, 5, 6, 1, 7],
ta-3-[3, 4, 2, 5, 1, 6, 7],
ri-4-[1, 2, 3, 4, 5, 6, 7, 8],
ti-5-[3, 2, 4, 5, 1, 6, 7],
no-6-[4, 3, 5, 2, 1, 6],
te-7-[3, 4, 2, 5, 1, 6],
di-8-[1, 2, 3, 4, 5, 6, 7],
ne-9-[3, 4, 5, 2, 6, 1, 7, 8],
ca-10-[1, 2, 3, 4, 5, 6, 7],
va-11-[3, 4, 2, 1, 5, 6],
ra-12-[3, 4, 2, 1, 5, 6],
si-13-[3, 2, 4, 5, 1, 6, 7],
la-14-[3, 2, 4, 1, 5, 6],
na-15-[3, 4, 2, 5, 1, 6],
li-16-[3, 2, 4, 5, 1, 6, 7],
mi-17-[2, 3, 4, 1, 5, 6],
in-18-[1],
co-19-[1, 2, 3, 4, 5, 6, 7],
le-20-[3, 2, 4, 5, 6, 1, 7]

We shall discuss now some examples of correction automatically generated and suggested on the basis of syllable structure and morphological decomposition. When the spelling corrector receives a word, it makes the assumption that it is actually misspelled and it suggests which substring contains the error. Morphological information is used to generate these guesses: however, in some cases, syllable structure is sufficient in itself to suggest where the correction has to be made.

Here below we shall present some examples in which there are clear violations of syllable structure constraints which are automatically corrected by the system. In other cases, it is the position of a certain syllable in the word that will tell the system where to address its search. Finally, the guess is generated from morphological decomposition processes and random approximation with the list of affixes of the substring suspected as the one containing the
error. The approximation must obey spelling correction restrictions: a) the correction must be within one letter in length of the intended substring, either by inserting or by deleting it. Clearly the system concentrates its attention on the suspected mistaken substring - either a syllable or a morpheme.

1) Insertion: terrritorio --> [ter, *r, ri, to, rio] --> territorio

   in this word, there is a sequence made up of three consonants of the same type. This is detected automatically by the system when trying to segment the second syllable and a new word is generated.

2) Transposition: succesisvo --> [suc, ces, *is, vo] --> successivo / next

   this word contains a syllable "is" in fourth position which may only appear in first position in Italian.

3) Insertion: stabilische --> [stabil - ische] --> stabilisce / establishes

   this word contains a root morpheme "stabil" which is correctly recognized in the Main Root Dictionary as possible adjective, noun and verb root. In turn the remaining substring is computed as respecting correction restrictions and is substituted by a verb desinence "isce".

4) Deletion: segretrio --> [segret - rio] --> segretario / secretary

   this word contains a root morpheme "segret" which is correctly recognized in the Main Root Dictionary as possible adjective, noun and verb root. In turn the remaining substring is computed as respecting correction restrictions and is substituted by a noun derivational suffix "ario".

   Following the subdivision of errors into four types suggested in the literature (Kukich, 388), we derived the following data:

   deletion = 42.8 %
   insertion = 28.3 %
   substitution = 17.5 %
   transposition = 9.5 %

   As to deletion, in great majority they are constituted by phonetic errors. Phonetic errors are basically due to the lack of gemination in word internal syllables; or to the lack of non syllabic |i| in diphthongs. Other common errors are represented by the omission of an |n| or of an |r| in a consonantal cluster; there also cases of omission of an entire syllable.

   As to insertion, in great majority they are constituted by phonetically determined misspelling due to deviation from standard Italian traceable to peculiar pronunciation of regional Italian. There is also a good amount of manual errors - e.g. three consonants of the same type. As before, but with opposite effect, some words have one syllable more.

   As to transposition, in great majority they are constituted by inversion in two
adjacent characters disregarding their phonetic type - whether consonant or vowel. Marginally there are double inversions. Also marginally there are cases of syllabic inversions.

Finally, substitution do not represent cases traceable back to some phonetic or regional problem, with only very few exceptions. They are real errors due to lack of concentration or typing errors.

In summary, our data confirm Kukich findings(392) in that:

- most errors (roughly 90%) are single instances of insertions, deletions, substitutions, or transpositions;
- most errors are within one letter in length of the intended word;
- no mispelling occurs in the first letter of a word.

We also found a tendency to undouble consonant geminates and/or to triplicate them. Finally, more than one third of all errors are deletions.

In fact we did not investigate on real-word errors in order to make comparisons. However, we assume that our data fit those found for other languages.

### 7.2. Grammar Checking and Agreement

In principle, grammar checking should concern itself mainly with agreement problems. Of course, in case the text to be grammar-checked is the product of an activity of second-language learning, the number of problems increase exponentially and agreement becomes only a minor problem perhaps. We shall only deal then with native speakers' productions, which in Italy covers a quite large type of linguistic competence. Italian is a language with many different dialect and regional varieties which cause the linguistic production to take on a sort of substandard realization. In particular, the use of subjunctive mood is very highly regionally biased, so is the use of left dislocations and topicalizations.

However, we restricted ourselves to check the use of adequate agreement structures taking for granted the overall sentential organization of constituents. There are two main agreement structures to check: NP internally and NP subject-verbal agreement. Italian like Spanish and Portuguese allows the NP subject to be freely inverted in postverbal position. However, more difficult tasks are required from agreement into a predicative complement with copulative verbs; and worst of all, clitic left or right dislocation structures require agreement to be checked between clitic raised pronoun, past participle and dislocated constituent, as in the following examples where we mark with bold the linguistic elements that undergo agreement:

\*i. Maria è diventata molto alta / Mary has become very tall
\*ii. Gino non la poteva aver conosciuta Maria / John could not have known her, Mary
\*iii. Maria, Gino non la poteva aver conosciuta / As for Mary, John could not have known her

A. NP as major constituent or as internal constituent of PP
B. Subject NP / tensed Verb agreement
C. Subject NP / Copulative Verb / Predicative Complement-Adjunct
D. Raised Complement Clitic Pronoun / Past Participle Agreement
E. Tensed Verb / Inverted Subject agreement

7.2.1 Syntactic Structure Checking

We proceed as follows:

We start by postulating the existence of a sentential structure and then we postulate the existence of an empty subject in case structure building at sentence level starts with VP: this information will allow us to fire agreement procedures in postverbal position, if required. Otherwise, subject NP is checked for agreement features collected at runtime from morphological features, and attached to the new data structure used to build f-structure representation.

We show here below some examples where grammar errors have been intentionally introduced: we starred mistaken words and give correct spelling in parenthesis.

1. NP internal agreement

1.i *Lo lettera ha assunta la *sue forma *definitivo. / The+mas+sing letter+fem has taken the++sing her+sing form+fem+sing final+sing+mas

Internal NP agreement must be checked whenever the head noun has been reached and decided accordingly: notice that Italian allows adjectives to be positioned both in front and after the head noun. However, agreement of NPs governed by or object of Preposition may require a similar mechanism to the one used for Subject-Verb agreement checking. Italian, like German, has compound prepositions or amalgams, where the article carries agreement features: this information has to be passed down to the embedded NP in order for the checker to work:

1.ii *Dalle segretar*ia sono state spedite tante lettere. / From the+fem+plur secretary+fem+sing are+plur been+fem+plur sent+fem+plur many+fem letters+fem

2. Subject NP / Copulative Verb / Predicative Complement-Adjunct

3.i Maria è *bello. / Maria is beautiful+mas+sing

Where the predicative adjective has the wrong agreement feature for gender. A more complex case is constituted by the following sentence where the restrictive relative clause has an inverted subject "sua moglie" which agrees with the tensed auxiliary and a past participle which has wrong agreement. In a predicative structure, the inverted subject prevents the creation of a clitic left dislocated topic structure. The clitic "gli" can only be interpreted as a dative pronoun and not as an accusative and has an external masculine antecedent

*Sono arrivate le lettere che gli ha *scritte sua moglie.

4. Clitic Left Dislocation and Past Participle

4.i Le lettere le aveva *scritti nella sua forma *definitive. / The+fem+plur letters+fem them+accusative+fem had+3rd+sing written+mas+plur in its+fem+sing form final+fem+plur

Here, in order to correct the mistakes, the parsers has to be able to detect the presence of a left dislocated Object NP followed by a clitic pronoun which can induce agreement on the following tensed verb in case it is expressed in its compound form, i.e. auxiliary plus past participle. Italian and other Romance languages require agreement on the past participle: in addition, whenever a clitic pronoun is raised in front of the main verb, it induces agreement on the participle even though "avere" auxiliary is being used. Notice that the lack of agreement on the past participle may constitute a correct grammatical sentence which however has a different interpretation of the clitic pronoun: with dative rather than accusative case.


**BIBLIOGRAPHY**


Alshawhi H.(1990), Resolving Quasi Logical Forms, *Computational Linguistics* 16(3).


Alshawhi H., J. van EIJK(1989), Logical Forms in the Core Language Engine, in *Proceedings of the 27th Annual Meeting of ACL*, Vancouver,


Azzam, Saliha, Kevin Humphreys and Robert Gaizauskas (1998), Evaluating a focus-based approach to anaphora resolution, in COLING-ACL ’98, 74-78, Montreal, Quebec, Canada.
Barrs A.(1988), Paths, Connectivity, and featureless empty categories, in Annali di Ca’ Foscari XXVII, 4, 9- 34.
Bosch P.(1980), The modes of pronominal reference and their constraints, in J.Kreiman, A.Ojeda(eds), Papers from the Parasession on Pronouns and Anaphora, CLS, Chicago, IL, 64-78.


Bresnan J.; Per-K.Halvorsen; J.Maling(1985), Logophoricity and bound anaphora, Ms., Stanford University.


Carlson G.(1977), Reference to Kinds in English, Ph.D. Dissertation, University of Massachusetts, Amherst.


Carter, David M. (1987), Interpreting Anaphora in Natural Language Texts, Ellis Horwood, Chichester, UK.


Computational Linguistics (1988),14(2), Special Issue on Tense and Aspect.

Conference on Grammatical Relations, San Diego.


D.Hindle and M.Roth(1993), Structural Ambiguity and Lexical Relations, *Computational Linguistics* 19, 1, 103-120.


Delmonte R.(1991), Grammatica e quantificazione in LFG, Quaderni Patavini di Linguistica.


Delmonte R., D.Bianchi(1998), Dialogues From Texts: How to Generate Answers from a Discourse Model, Atti Convegno Nazionale AI*IA, Padova, 139-143.


Di Eugenio B.(1990), Centering Theory and the Italian pronominal system, COLING, Helsinki.


Dowty D.(1986(ed), Tense and Aspect in Discourse, Monograph Number of Linguistics and Philosophy 9, 1.


Hahn, Udo, Michael Strube and Katja Markert, (1996), Bridging textual ellipsis, in *COLING ’96*, 496-501, Kopenhagen, Denmark.


Hindle D. & M. Roth (1993), Structural Ambiguity and Lexical Relations, *Computational Linguistics* 19, 1, 103-120.

Hinrichs E. (1988), Tense, Quantifiers, and Contexts, in Computational Linguistics - *Special Issue on Tense and Aspect* - 14, 2, 3-14.


Jackendoff R.(1976), Towards an Explanatory Semantic Representation, Linguistic Inquiry 7, 89-150


Lecomte J. (1998), *Le Categoriseur Brill14-JL5 / WinBrill-0.3*, INaLF/CNRS.


McCawley J. (1981), *Everything that Linguistics have Always Wanted to Know about Logic* - *but were ashamed to ask*, The University of Chicago Press, Chicago.


Mitkov R. (1995), Two Engines are better than one: Generating more power and confidence in the search for the antecedent, *Proceedings of Recent Advances in Natural Language Processing*, Tzigov Chark, 87-94.


Montemagni S., F. Barsotti, M. Battista, N. Calzolari, A. Lenci O. Corazzari, A. Zampolli, F. Fanciulli, M. Massetani, R. Basili R. Raffaelli, M.T. Pazienza, D. Saracino, F. Zanzotto,


Oepen, Stephan, Dan Flickinger, Chris Manning, Kristina Toutanova, LinGO Redwoods - A Rich and Dynamic Treebank for HPSG, *webpage of the project*.


Bibliography

Pinker Steven & Ray Jackendoff, 2005. The Faculty of Language: What's Special about it?, Brandeis University, "Cognition, 95, 201-236".
Poesio, Massimo, Renata Vieira and Simone Teufel (1997), Resolving bridging references in unrestricted text, in Ruslan Mitkov (ed.), Proc. ACL Workshop on Operational Factors in Robust Anaphora Resolution, 1-6, Madrid.
Pollard C., I. Sag (1989), Anaphors in English and the scope of the binding theory, MS, Stanford University.
Puglielli A., Castelfranchi C. (1976), Sul congiuntivo nelle frasi relative e il modo ipotetico, Rivista di grammatica generativa 1(2).
Roberts C.(1990), Modal Subordination, Anaphora and Distributivity, Garland, New York.
Svenonius, Peter. 2001. Locality, Phases and the Cycle, Manuscript, University of Tromsø, Draft 19.07.01.
Tesnière, Lucien (1959), Éléments de syntaxe structurale, Paris, Klincksieck.


Wiebe J.(1993), Tracking Point of View in Narrative, Memoranda in CCS-261, CRL, New Mexico State University.


