Relative clause attachment and anaphora: Conflicts in grammar and parser architectures

Rodolfo Delmonte
Università "Ca Foscari", Venezia

Abstract

This paper is concerned with the use of shallow and partial NLP techniques in heavily linguistically demanding tasks such as the one posed by summarization. This approach should not be taken as an alternative way of coping with the same problems by means of a complete system of text understanding and summarization, but as a proposal in line with current NLP research in unrestricted texts that assumes that partial processing can be more suitable and nonetheless useful for better satisfaction of certain requirements. In particular, morphological analysis is a prerequisite in order to better cope with Out of Vocabulary Words(OOW) by means of guessing techniques based on morphological rules; statistical processing is then assumed to 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 can be carried out in order to execute anaphora resolution and a Discourse Model will be built with a reduced ontology and a limited number of relations and properties. Partial semantic interpretation means that not all semantic relations will be detected and encoded appropriately. Nonetheless, what is captured by partial analysis can still be useful to carry out such important tasks as anaphora resolution at discourse level and a rough evaluation of entity relevance in order to better grasp what topic has been the most relevant one.

We will concentrate our attention to parsing, however, and in particular to problems related to challenges coming from attachment of structurally ambiguous constituents such as prepositional phrases (PPs) and relative clauses (RCs).

In a final section, we discuss data from syntactic Treebanks of English – the Penn Treebank – and Italian, the Italian Treebank and the Venice Treebank.

1. Parsing Strategies and Preferences

Differently from what is asserted by global or full paths approaches (see Schubert, 1985; Bear & Hobbs, 1988; Hobbs, Stickel, Appelt, Martin, 1993), 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. Other important approaches are represented by Hindle et al., 1993, who attempt to solve the problem of attachment ambiguity in statistical terms. The important contribution they made, which was not possible in the ‘80s, is constituted by the data on attachment typologies derived from syntactically annotated corpora. We will report similar data on Italian, derived from our corpus of Italian, currently being annotated as a syntactic treebank, a portion of which will be used as sample with comparable length to the English one: the authors worked on a test sample made up of 1000 sentences in which their parser identified ambiguous attachment conditions. In order to simulate automatic disambiguating procedures, we shall use data derived from our shallow parser, which has an output comparable to the Fidditch parser quoted by Hindle et al. and shown in the article, which the final output is recorded in the Treebank, which is the result of the concurrent work done by manual annotators, automatic validation procedures and my final supervision of the resulting constituent structures, visualized by a tree-viewer.

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 analyzer, 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 compatibility 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.

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.

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 top-down depth-first parser which is optimized so as to incorporate all linguistically motivated predictable moves. An already parsed RC 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 RC 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 RC. 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. It builds up a table of wellformed 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 wellformed 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.

Lookahead is used in a number of different ways: it may impose a wait-and-see policy on the top-down strategy or it may prevent following a certain rule path in case the stack does not support the first or even second match:

i. to prevent expanding a certain rule
ii. to prevent backtracking from taking place by delaying retracting symbols from 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. In addition, since in some cases we extend the lookahead mechanism to include two symbols and in one case even three symbols, the possibilities are 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 all our rules satisfy the requirement to have a preterminal in first position in their right-hand side - almost all rules. There are in fact some well-known 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 analyzed 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 the example “The thieves stole the painting in the night.” that we discuss in a section below. The PP is at first analyzed 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, or wellformed substring table) 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.

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 annihilate 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.

2. Relative Clause Attachment and Binding

Relative clause attachment may be triggered by binding requirements imposed by a short anaphor contained within the relative clause itself: in case more than one possible attachment site is available in the previous structure, and the relative clause itself is extraposed, a conflict may arise as to the appropriate s/c-structure which is licensed by grammatical constraints but fails when the binding module tries to satisfy the short anaphora local search for a bindee.

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 given 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.

However we show that anaphoric binding requirements may cause the parser to fail because the structure is inadequate. We propose a solution to this conflict by anticipating, for anaphors only the though, the agreement matching operations between binder and bindee and leaving the coindexation to the following module

2.1 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).

3. Parser

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 well-known problem of ambiguity, which is a pervading problem in real text/life situation, and it is regarded as inseparable benchmark of any serious parser of any language to cope with.

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 behavior of the human processor. In accordance with this criterion, we assume that the implementation should closely mimic 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.

The parser we built is wrought within LFG framework which allows us to use a much richer representation, closer to the semantics. In particular, all levels of Control mechanisms which allow coindexing at different levels of parsing gave us a powerful insight into the way in which the parser should be organized.

Yet the grammar formalism implemented in our system differs from the one suggested by the 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: 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. 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 languages chosen. 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.

Of course there are a number of marked differences in the treatment of specific issues, concerning Romance languages, which are not sufficiently documented in the linguistic literature (see Bresnan, 2000). In particular,

- we introduced an empty subject pronominal - little pro- for tensed propositions, which had different referential properties from big PRO; this had an adverse effect on the way in which c-structure should be organized. We soon realized that it was much more efficient and effective to have a single declarative utterance-clause level where the subject constituent could be either morphologically expressed or Morphologically Unexpressed. In turn MUS or little pros could be computed as variables in case the subject was realized in postverbal position. At the time, LFG posited the existence of a rule for sentence structure which could be rewritten as VP both in case there was no subject, and in case the subject was expressed in postverbal position, an approach that we did not implement;

- we also use functional constituents like CP and IP: CP typically contains Aux-to-Comp and other preposed constituents, adjuncts and others; IP contains negation, clitics, and tensed verbal forms, simple and complex, and expands VPs as complements and postverbal adjuncts;

- each constituent is semantically checked for consistency before continuing parsing; we also check for Uniqueness automatically by variable instantiation. But sometimes, in particular for subject-verb agreement we have to suspend this process to check for the presence of a postverbal NP constituent which might be the subject in place of the one already parsed in preverbal position;

- syntactic constituency is replicated by functional constituency: subject and object are computed as constituents of the annotated c-structure, which rewrite NP - the same for ncomp - this is essential to assign the appropriate annotated grammatical function; this does not apply to VP, a typical LFG functional non-substantial constituent;

- our lexical forms diverge from the ones used in the theoretical framework: we introduced aspectual categories, semantic categories and selectional restrictions in the main lexical entry itself;

- we also have semantic roles already specified in the lexical form and visible at the level of syntactic-semantic parsing;

- rather than generating a c-structure representation to be mapped onto the f-structure, we generate a fully annotated c-structure representation which is then checked for Grammatical Principles Consistency at the level of number/type of arguments and of Adequacy for adjuncts, with a second pass on the output of the parser, on the basis of lexical form of each predicate and semantic consistency crossed checks for adjuncts.

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 are built in order to produce f-structure configuration.

Each major constituents may be associated with different functional values:

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 --> Open or closed Adjuncts according to its selectional properties, occurring anywhere in the sentence according to their semantic nature;

d. PP --> 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' --> 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' -->or 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.

The parser is made up of separate modules:

1. 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;

2. 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;

3. 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.

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 5,000 entries for each language.

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, but see below. 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 Extraposition formalism presented by Pereira(1981;1983). Procedurally speaking, the grammar is implemented using definite clauses. In particular, Extraposition 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.

3.1 Sorting out Language Dependent Differences

In our perspective, we would like to take a very pragmatic and experimental stand on the problem of ambiguity. In the first place, we want to look only at structural ambiguity and build up a comprehensive taxonomy from a syntactic point of view; secondly, we want to spot language dependent ambiguities in order to define the scope of our research appropriately.

A. Omissibility of Complementator
- 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

The list of examples here below includes sentences used in the literature to support one or the other of the two parsing theories, GPT and IIT (see Altman (ed), 1989).

1. Mary put the book on the table
2. Mary put the book on the table in her bag
3. Mary saw the cop with the binoculars
4. Mary saw the cop with the revolver
5. The thief stole the painting in the museum

6. The thief stole the painting in the night
7. John saw Mary in the kitchen
8. John saw Mary from the bathroom

Altmann and Steedman(1989) note that several of the ambiguities present in these sentences and resolved by Minimal Attachment involve contrasts between NP modification and other structures. However, examples 2, 3, and 5 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 example 2, where the pp should be taken as np modifier. If we look at subcategorization requirements of the verb, example 1 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 by FP.

Example 3 contains an instrumental adjunct: when the head preposition "with" is met, the parser will not close the np "the cop" and will continue building an internal pp modifier since preposition "with" heads a compatible np modifier, a comittative. In our dictionary, the verb "see" has one single lexical entry but a list containing two different lexical forms. The first form in the list has a higher number of arguments, *see* <SUB/perceiv, OBJ/theme, PCOMP/locat> where the Pcomp predicates a location of the Object; and a second form, where the Pcomp is absent. In order for a Location pp to be accepted, the head preposition should be adequate, and "with" does not count as such. In examples 3 and 4, 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 comittative - and the analysis should be allowed to continue until the pp is fully analyzed. In other words, SE should verify pp attachment consistency inside the NP constituent. However, this may only happen in case MA is deactivated by FP, after matching with lexical information has failed.

In the following examples (5, 6), argument structure plays no role whatsoever: instrumentals, comittatives, locatives with predicate "steal" are all cases of sentential adjuncts, and only SE can apply. As a matter of fact, "in the museum" might be freely attached lower at NP level as well as higher at Sentence level. This is due to the fact that head preposition "in" constitutes a viable local NP modifier and there are no argument requirements from the main verb predicate. However, "in the night" is not a possible NP modifier and example 6 is a clear case of minimal attachment sentence. On the contrary, in example 5, PP attachment is ambiguous between a np internal modifier and VP level attachment.

In example 7, 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, and this can be achieved by FP which activates MA and makes the locative PP available at VP level.

In example 8, on the contrary, we understand that the location from the 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. This case is treated as the previous one, except that the PP is computed as sentence adjunct rather than as VP complement.

Given that in our framework Quantifier Raising is performed before Anaphoric Binding and will produce new arcs in the graph to represent the scope of quantifiers, this will also undergo failure in order to try a new analysis. This is both time-consuming and unrealistic. A simpler way to solve this problem is to introduce Short Binding as has been defined above. 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 backtraing 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.

4. The Problem of Relative Clause Attachment

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:

9. The doctor called in the son of the pretty nurse who hurt herself

10. 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 (hence RC), a phenomenon very common in English but also in Italian and other languages. The related structures theoretically produced, could be the following ones:

```
9 a.
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]]]]]]]]]
```

```
10 a.
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 10a. will be generated by a parser of English without special provisions. The structure produced in both cases will be 1a. as 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, as the respective f-structures show, the only output available is the one represented by 10b, which wrongly attaches the RC to the closest NP adjacent linearly to the relative pronoun:

10 b.

```
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]]]]]]]]]
```

And the respective f-structures represent the mapping from c-structure preceding the launching of the Binding Module:

```
f-structure index:f3
pred:call_in
lex_form:[np/subj/[human],np/obj/them_nonaff/[human],idio
ms/form/[in]]
voice:active
mood:ind
tense:pastc
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:+
```

```
                mod_a:index:sa3
               cat:[subjective]
               pred:pretty
mods:relative:topic:t_type:relative
          index:sn22
          cat:[human,social]
          pred:who
          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]
       spec:def:+
       tab_ref:[+ref,-pro,+ana,+me,-subj]
tab_ref:[+ref,-pro,-ana,+class]
       tab_ref:[+ref,-pro,-ana,+class]
form:in
```

F-structure for the second example is identical to the first one except for the agreement features of the reflexive anaphoric pronoun ***himself*** which is marked for masculin gender.

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 from taking 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.

5. Short Anaphora

The parser we use 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, since short anaphora requires binding to take place within a local domain, f-command - the corresponding c-command in functional terms, applied to grammatical functions and a graph structure - will impose the same level of containment for both the pronoun and the antecedent. And given 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 since 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, as the two modules obey different Principles and satisfy them separately.

We will now put forward a theoretical proposal 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:

> "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, 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 since 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 this 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 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 given 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 since 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 as all sentences can be adjusted within the parser, sooner or later. Also processing time is not controllable since 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, as the parser has both a lookahead mechanism that tells the parser which rule to access or not at a given choice point.

English is a language with a very impoverished pronominal system which uses the only form of reflexive pronouns it possesses - the composite one as in him-self - for a variety of uses as the following examples show:

a. John hates himself.
b. The boys were afraid that picture of themselves would be on sale.
c. Bill remembered that The Times had printed a picture of himself in the Sunday edition.
d. Max boasted that the queen invited Lucie and himself for a drink.

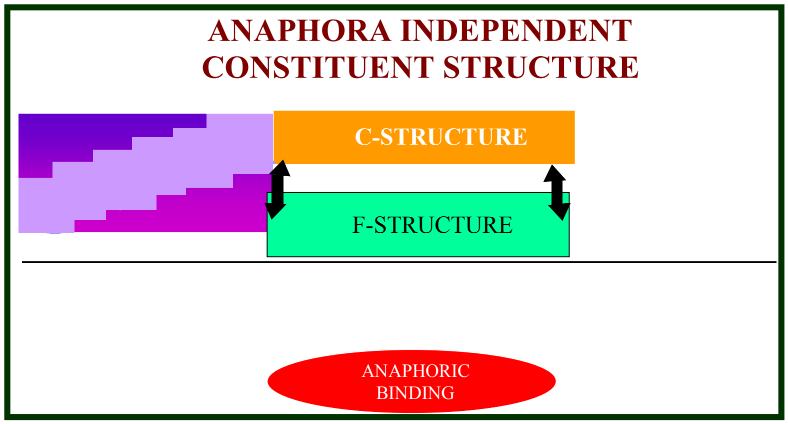

Fig.1 Parser Architecture with Anaphora Independent Constituent Structure

5.1 Italian Reflexive Pronouns

a. Anna odia se stessa.
b. Anna sapeva che la descrizione di se stessa inviata a quella ditta sarebbe stata di grande giovamento.
c. Maria non sapeva che il Corriere della Sera avrebbe parlato di sé.

In other words, whereas English would use the same wordform both in the case of the anaphor included in a SUBJect and OBJect NP of a complement clause, as shown in the b. and c. examples, Italian would not allow it and opt for a simple form "sé" which is however deprived of functional features. In addition, English also allows the same anaphoric wordform as the direct OBJect NP of a complement clause, something which is completely out in Italian. In both cases, however, the condition for the anaphor to be bound out of its minimal nucleus, is that there be no intervening possible binder. The real problem is however constituted by all cases in which the same anaphoric expression is used as logophoric pronoun. And this happens only in appropriate contexts; some examples are shown below:

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 embarassed. [Woolf, 60]

[Philip is starting an affair with Désirée, Zapp’s wife] Whom HE [Philip] was supposed to be fooling, he couldn’t image. Not the twins, surely, because Désirée, in the terrifying way of progressive American parents, believed in treating children like adults and had undoubtedly explained to them the precise nature of her relationship ***with himself***. [Lodge, 170]

In these cases, the subject of consciousness, i.e. the character from whose point of view deals the narration is the only possible antecedent for the anaphor and must be recovered usually from discourse level.

To sum up, the situation is as follows in English:

- English Reflexive Pronouns are also Logophoric Pronouns

- A Parser has to take decisions as to whether a Short Anaphor must be bound in a given Domain, its Minimal Nucleus

- Or else

  - Reject the structure as ungrammatical if no Domain of Point of View is present
  - Accept the structure in case the anaphor can be left free if a Subject of Consciousness is available to bind the anaphor.

As to Domains, we refer to LFG for the definition of the Minimal Nucleus:

(53) Minimal Nucleus Condition
A binding constraint designator ((GF a -) GF') in a nuclear binding constraint is subject to the minimal nucleus condition if
(i) GF and GF' are argument functions and
(ii) if the attribute string a is nonempty
then setting a = xa for some attribute a and possibily empty string of attributes x, the off-path constraint Ø(® SUBJ) holds for every attribute in GF x.

(54) Nuclearity Constraint
A nuclear pronoun is lexically specified
((GF a -) GF' INDEX) = (- INDEX)
subject to the Minimal Nucleus Condition (53).

Finite Domain Condition:
A binding constraint designator ((GF a -) SUBJ) is in the minimal finite domain when, if the attribute string a is nonempty, then setting a = xa for some attribute a and possibly empty string of attributes x, Ø(®TENSE) holds for every attribute in GF x.

NUCLEUS:
Given an f-structure ***f***, the nucleus of ***f*** is the subset of ***f*** consisting of the PRED element and all of the elements whose attributes are functions designated by the PRED. All definitions are taken from Bresnan(2000).

Here below is the Table of l Anaphoric Systems comprising three typologies differing in the combination of the three basic features Subjectivity (sbj), Nuclearity (ncl) and Logoforicity (log). As can be seen, only Icelandic has a type of pronominal specific for logophoric binding.

Table 1. **Anaphoric Binding Systems**

ITALIAN	ENGLISH	ICELANDIC
lui {Øsbj,-ncl,Ølog}	him {Øsbj,-ncl,Ølog}	ham {Øsbj,-ncl,Ølog}
se_stesso{Øsbj,+ncl,Ølog}	himself{Øsbj,+ncl,Ølog}	hamselv{-sbj,+ncl,Ølog}
si{+sbj,+ncl,Ølog}		segselv{+sbj,+ncl,Ølog}
sé/proprio {+sbj,-ncl,Ølog}		seg{+sbj,-ncl,Ølog}
		deim{+sbj,Øncl,+log}

5.2 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 analyze 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 main constituents receive interpretation 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 obviously 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 annihilate all naturally available parsing mechanisms, like for instance lookahead, and use a process of trial-and-error to reconstruct the previously built structure in order not to fall into the same mistake.

Going back to our couple of examples of the Extraposed Relative Clause containing a Short Anaphor, the question would be to prevent Failure since we do not want Constituent Structure Building to be dependent upon the Binding of the Short Anaphor. The only way out of this predicament is that of anticipating in Sentence Grammar some of the Agreement Checking Operations as proposed above. So the Parser would be able to backtrack while in the Grammar and to produce the attachment of the Relative Clause at the right place, in the higher NP headed by the masculine N, "the son". The important result would be that of maintaining the integrity of Syntax as a separate Module which is responsible in "toto" of the processing of constituent structures. The remaining Modules of the Grammar would be fully consistent and would use the information made available in a feeding relation, so that interpretation will follow swiftly.

To integrate this suggestion coming from Implementation problems into the theoretical Framework of LFG or other similar theories, we simply need to integrate GRAMMATICALITY PRINCIPLES as they have been stipulated so far consisting of:

- UNIQUENESS
- COHERENCE
- COMPLETENESS

with the additional restriction:

- BOUND ANAPHORA AGREEMENT

i.e. short anaphors should be checked before leaving sentence grammar, for agreement with their antecedents iff available in their Minimal Nucleus.

6. Treebank Derived Structural Relations

As noted above in the Introduction, an important contribution to the analysis of PP attachment ambiguity resolution procedures is constituted by the data made available in syntactic Treebanks. Work still underway on our Venice Italian Corpus of 1 million occurrences revealed a distribution of syntactic-semantic relations which is very similar to the one reported by Hindle et al. In their recent paper and shows in the Table 2. below.

Table 2. Shallow Parsing & Statistical Approaches
(Data from D.Hindle & M.Roth, Structural Ambiguity and Lexical Relations)

Argument noun	378	39.5?	
Argument verb	104	11.8%	
Light verb	19	2.1%	
Small clause	13	1.5%	
Idiom	19	2.1%	57%
Adjunct noun	91	10.3%	
Adjunct verb	101	11.5%	
Locative indeterminacy	42	4.8%	
Systematic indeterminacy	35	4%	
Other	78	8.8%	39.4%
TOTAL	880	100%	

As the data reported above clearly show, most of the prepositional phrases are constituted by arguments of Noun, rather than of Verb. As to the remaining data, adjuncts are represented approximately by the same amount of cases, 11% of the sample text. At first, we collected all information on prepositions as a whole and then we searched into our Treebank and looked for their relations as encoded in the syntactic constituent structure. Here below, we report data related to prepositions for the whole corpus: notice that in Italian as in English, preposition "of"/di would be used mainly as a Noun argument/modifier PP.

Table 3. Shallow Parsing & Statistical Approaches

Venice Italian Corpus 1 million tokens		
All prepositions – 54 different types or wordforms: 170,000 occurrences		
Argument-like preposition		
DI/of and its amalgams	**78,077 → 46%**	
A/to and its amalgams	**29,191 → 17.2%**	
DA/by-from and its amalgams	**13,354 → 7.9%**	**71.1 %**
Adjunct-like prepositions		
IN and its amalgams -	**21,408 → 12.6%**	
PER and its amagalms -	**12,140 → 7.1%**	
CON and its amalgams -	**5,958 → 3.5%**	**23.2%**

In contrast to English, however, nominal premodifiers do not exist in Italian, and the corresponding Italian Noun-Noun modification or argument relation without preposition would be postnominal. Such cases are not very frequent and constitute less than 1% of Noun- Noun head relations.We then selected 2000 sentences and looked at all prepositional phrases in order to highlight their syntactic and semantic properties, and we found out the following:

- The number of prepositional phrases in Italian texts is four times bigger than the one reported for English Texts, and this might be due to the poor use of nominal modifiers which in Italian can only be post-modifiers, attested from an analysis of the sample text;
- PPs Arguments of Nouns are 53% in Italian and 39% in English, i.e. 14% more in Italian;
- PPs Arguments of Verbs are 15% in Italian and 17% in English – if we sum all argument types and idioms together -, i.e. 2% more in English;
- Adjuncts of Nouns and Verb are 31% in English and 32% in Italian.

Thus, the only real big difference between the two languages can be traced back in the behaviour of PP noun arguments, which in turn can be traced back to a language specific typological difference: the existence of prenominal modifiers in English and not in Italian – or at least, not yet substituted by the use of postnominal modification.

Table 4. Quantitative Syntactic and Semantic Distribution of PPs in VIC

PPs not headed by Da or Di	**3977**	**51%**
Argument of verb	**944**	**23.7%**
Argument of Noun	**1300**	**32.7%**
Adjunct of Noun or Verb	**1733**	**43.6%**
PPs headed by DA	**504**	**6.5%**
Argument of Verb	**164**	**32.5%**
Argument of Noun	**114**	**22.6%**
Adjunct of Noun or Verb	**226**	**44.9%**
PPs headed by DI	**3314**	**42.5%**
Argument of Verb	**72**	**2.17%**
Argument of Noun	**2733**	**82.5%**
Adjunct of Noun or Verb	**509**	**15.4%**
TOTAL	**7795**	**100%**
Arguments of Verb	**1180**	**15%**
Arguments of Noun	**4147**	**53%**
Ambiguous PPs	**2468**	**32%**

7. RC Attachment: Experimental Results from Treebanks

We decided to look at corpus data derived from available treebanks in order to ascertain whether the phonemon we are modeling is actually present in real texts. We also wanted to verify whether the RC extraposition was subject to variation from one language to another. We searched in the available treebanks, PennTreebank for English with 1,000,000 tokens, and the Treebank of Italian we are currently working in for syntactic constituenty XML annotation as well as the Venice Treebank made up of approximately the same number of tokens for a total of 300,000 tokens.

We considered only relative clauses with morphologically expressed complementizer, thus disregarding all reduced relative clauses. As to the distinction between extraposed vs. non-extraposed, we simply looked at the number of brackets – only one - intervening between the constituent label introducing the relative clause in PennTreebank, which is the following (SBAR (WH, and none in the VeniceTreebank. For all remaining cases, we counted an extraposed RC.

We tabulated the results in the Table 3. below where we see that Italian is a language much richer on Relative Clauses than American English. In particular, the amount of relative clauses in the Italian Venice Treebank is 3 times that of the PT. Yet more interesting seems the ratio of Head Adjacent vs. Non Head Adjacent RCs: we see that here, whereas Italian has 1 ambiguous RC every 4 RCs, PT has 1 every 6. We can thus conclude that Italian is much more ambiguous to be parsed than English as far as relative clause attachment is concerned.

Table 5. Treebank Derived Structural Relations for Relative Clauses

	Total No. Tokens	Total No. Sentences	Total No. Rel. Cls.	Head Adjacent		Non Head Adjacent	
PENN Treebank	1 000 000	44808	11559	8906	77.05%	2653	22.9%
SUSANNE Corpus	130 000	7912	1380	1089	78.9%	291	21.1%
VENICE Treebank	300 000	11108	5155	3867	75%	1288	25%

However, the most interesting fact is constituted by the proportion of relative clauses in relation to the total number of sentences: Generic American English in the Susanne Corpus counts 1 relative clause every 5/6 sentences; Specialized American English in the PT goes up to 1 relative clause every 4 sentences. Italian raises the proportion to one relative clause every 2 sentences.

Table 6. Treebank Derived Structural Relations for Relative Clauses

	Total No. Tokens	Total No. Sentences	Total No. Rel. Cls.	Compless Relative Clauses	Ratio Rel. Cls. /Sentence	Ratio Rel. Cls./Tot. Tokens
PENN Treebank	1 000 000	44808	11559	2724	25.8%	1.16%
SUSANNE Corpus	130 000	7912	1380	106	17.44%	0.92%
VENICE Treebank	300 000	11108	5155	-	46.40%	1.72%

8. Conclusions

From what we wrote in the previous sections, it seems clear that attachment problems constitute a serious drawback to the use of shallow parsing techniques in real texts. We experimented both on English and Italian with our shallow and deep parsers and found out that only the use of lexical information as part of the parsing process to accomplish Verb Guidance can assure a quasi-deterministic parse to prevent wrong parses from taking place.

Besides, we also experimented the introduction of shallow attachment criteria, derived from empirical evaluation of treebanks: we treated all di/of headed prepositional phrases as complements/modifiers of the most adjacent nominal head and treated the rest as adjuncts. This choice would save only a percentage of all prepositional phrases and leave the rest to chance. In particular, there would be no principled reasons to tell the shallow parser to attach prepositional phrases higher up more than one constituent boundary. Nor would it be possible to choose in favour of verbargument attachment rather than noun-head attachment. Accomplishing a deterministic policy with a deep parser requires the lexicon to be equipped with all information related to syntactic and semantic selection of arguments and adjuncts. It also requires the parser to be equipped with semantic and aspectual class information to help detect the appropriate attachment of temporal adjuncts.

Shallow parsing does not allow a smooth implementation of control devices which alone will assure predicative complements to be properly bound to their governor. And in general, all semantic processing at structural level is hardly achievable due to the lack of important structural relations to ensure adequate semantic interpretation to take place.

References

Altman, G.T.M.(ed.). 1989. "Parsing and interpretation". *Language and Cognitive Processes* 4, 3/4, Special Issue.

Bianchi, D., Delmonte, R., Pianta, E.. 1992. "GETA_RUN A general text analyzer with reference understanding". In *Proceedings of the 3rd Conference on Applied Natural Language Processing, Systems Demonstrations*, 9-10. Trento, ACL 1992.

Bresnan, J. 1982. "Control and complementation". In *The Mental Representation of Grammatical Relations*, J.Bresnan(ed.), Cambridge, Mass.: MIT Press.

Bresnan, J. (ed). 1982. *The Mental Representation of Grammatical Relations.* Cambridge Mass: MIT Press.

Bresnan, J. 1990. Invited speech, GLOW, Cambridge.

Bresnan, J.; Halvorsen, Per-K., Maling, J. 1985. Logophoricity and bound anaphora, MS, Stanford University.

Bresnan, J., Mchombo, J.M. 1987. "Topic, pronoun and agreement in Chichewa". *Language* 63, 4: 741-782.

Bresnan, J., Kanerva, F. 1989. "Locative inversion in Chichewa: A case study of factorization in grammar". *Linguistic Inquiry* 20, 1: 1-50.

Bresnan, J., Zaenen, A. 1990. "Deep unaccusativity in LFG". *Proceedings Fifth Biennal Conference on Grammatical Relations,* San Diego.

Chomsky, N. 1986. *Barriers*, MIT Press, Cambridge, Mass.

Clifton, C. and Ferreira, F. 1989. "Ambiguity in context". In *Language and Cognitive Processes*, G.Altman (ed), op.cit., 77-104.

Delmonte, R. 1984. "La "syntactic closure" nella teoria della performance". *Quaderni Patavini di Linguistica* 4, Padova, 101-131.

Delmonte, R. 1992. "Linguistic and inferential processing". In *Text Analysis by Computer*, Padova, Unipress.

Delmonte, R., Bianchi, D. 1991. "Binding pronominals with an LFG parser". In *Proceedings of the Second International Workshop on Parsing Technologies*, 59-72. Cancun (Messico), ACL 1991.

Delmonte, R. 2000. "Generating and parsing clitics with GETARUN". In *Proceedings CLIN'99*, 13-27. Utrech.

Delmonte, R. 2000 (to appear). "Parsing preferences and linguistic strategies". In *Proceedings of Workshop Communicating Agents*, 15. IKP, Bonn.

Delmonte, R. 2000. "Parsing with GETARUN, Proc.TALN2000", *7° conférence annuelle sur le TALN*, 133-146. Lausanne.

Frazier, L. 1987a. "Sentence processing". In *Attention and Performance XII*, M.Coltheart(ed), Hillsdale, N.J.: Lawrence Elbaum.

Frazier, L. 1987b. "Theories of sentence processing". In *Modularity in knowledge representation and natural language understanding*, J.Garfield (ed), 291-308. Cambridge Mass.: MIT Press.

Garrod, S., Sanford T. 1988. "Thematic subjecthood and cognitive constraints on discourse structure. *Journal of Pragmatics* 12: 599-534.

Hindle, D. and Roth, M. 1993. "Structural ambiguity and lexical relations". *Computational Linguistics* 19, 1: 103-120.

Kay, M. 1980. "Algorithm schemata and data structures in syntactic processing". *CSL-80*-12, Xerox Corporation, Palo Alto Research Center.

Kennedy, A. et al. 1989. "Parsing complements". In *Language and Cognitive Processes*, G.Altman (ed), op.cit., 51-76.

Pereira, F. and Warren, D. 1980. "Definite clause grammars for language analysis. A survey of the formalism and a comparison with augmented transition networks". *Artificial Intelligence* 13: 231- 278.

Pereira, F. 1981. "Extraposition grammars". *American Journal of Computational Linguistics* 7, 4: 243-256.

Pritchett, B. L. 1992. *Grammatical Competence and Parsing Performance*, , Chicago: The University of Chicago Press.

Schubert, L. K. 1984. "On parsing preferences", *Proceedings. of COLING*, 247-250.

Steedman, M. J. and Altmann G.T.M. 1989. "Ambiguity in context: a Reply". In Altman(ed), op.cit., 105-122.