WEIGHTED INTERACTION OF SYNTAX AND SEMANTICS IN NATURAL LANGUAGE ANALYSIS

Leonardo lesmo and Pietro Torasso

Dipartimesnto di Informatics - Universita' di Torino
Via Valperga Caluso, 37 - 10125 TORINO - ITALY

ABSTRACT

The present paper discusses the extensions to the parsing strategies adopted for FIDO (a Flexible Interface for Database Operations). The parser is able to deal with ill-formed inputs (syntactically ill - formed sentences, fragments, conjunctions, etc.) because of the strict cooperation among syntax and semantics. The syntactic knowledge is represented by means of packets of condition-action rules associated with syntactic categories. The non-determinism is mainly handled by means of rules which restructure the parse tree (called "natural changes") so that the use of backtracking is strongly limited.

In order to deal with difficult cases in which no clear-cut mechanism exists for excluding an interpretation, a weighting mechanism has been added to the parser so that it is possible to explore few different hypotheses in parallel and to choose the best one on the basis of complex interaction among syntax and semantics.

INTRODUCTION

If one considers the evolution of computerized natural language understanding systems (Charniak, 1981), it becomes apparent that the role of syntactic knowledge can vary from being the basis of the process to being completely neglected. In the first case, the conversion form a linear sequence of words to a corresponding structured representation (parse tree) is guided only by the syntactic knowledge, whereas the other knowledge sources (mainly semantics) have the task to translate the parse tree into a meaning representation; within the other approach the understanding process is viewed as a whole and no special role is played by syntactic knowledge (given that such a knowledge is assumed to exist).

As regards purely semantic approaches, they present some problems with respect to the perspicuity of the model. In particular, the structural information (e.g. the fact that in English the adjectives precede the noun necessarily, whilst in Italian they do not) has to be duplicated for the different entities or represented in procedural form within the analysis program. Since we believe that structural information is fundamental in the analysis process, and that its explicit representation increases the understandability (and the modifiability) of the systems, we will take in the following the opposite view, trying to start from syntactic approaches and to justify the increasing role of semantics within them.

The aim of this introduction is to discuss how the semantics can be used to increase the effectiveness of N.L. analysis. In particular, three points will be set forth:

- from the point of view of efficiency of processing, the grammar-based approaches have to use semantic information as soon as possible
- the human ability to understand ill-formed fragments suggests to reduce the predominance of syntactic knowledge and, again, to use more heavily semantic information
- the phenomenon of garden paths shows that two different modes of operation exist: normal and backup. However, purely syntactic approaches fail to account for the phenomenon in a perspicuous way.

If we consider a grammar only from the point of view of expressive power, of course we can, after a thorough analysis of the phenomena occurring in natural language use, hope to find a grammar that characterizes all and only the sequences of words that constitute "acceptable" sentences. The study of the required power of N.L. grammars received considerable attention in the past (for a recent and thorough overview see (Pullum, 1984); some prominent positions are described in (Perrault, 1984)).

However, it has often pointed out that a comprehensive (and useful) N.L. understanding system should also take into account higher-level problems; in particular, it should also provide the researcher with some insights about the relationships existing between syntactic structures and semantic interpretation.

Most classical studies, both within the field of formal languages and within the field of natural languages viewed the semantic interpretation as a process "appended" to the syntactic analysis. It is widely accepted that this way of using semantics is highly inefficient: the number of alternative parses is often so high (especially when prepositional phrases are present), that it is not cost effective to delay the intervention of semantics (Sagalowicz, 1980); on the contrary, it is preferable to use semantics both as a meaning-construction process and as a source of further constraints for the analysis, as soon as possible during the analysis itself (Woods, 1980).

However, some other problems deserve attention. The first of them concerns the idea of "correctness" that, as stated above, is at the basis of the grammar-based approach. It is well known that in most cases, humans are able to understand the 'meaning of sentence fragments that are syntactically ill-formed without any apparent difficulty. It's worth observing that the locution "ill-formed sentences" does not refer: exclusively to sentences that can reasonably be rejected on syntactic grounds. For example, the existence of a conjunction (Huang, 1984) can result in a sentence fragment that is ill-formed, although the entire sentence must be considered correct under any plausible definition of syntactic correctness. Ihe problem of parsing ill-formed inputs has become very popular recently; a number of papers appeared in a special issue of the ACL Journal (AJCL, 1983). It must be noted that the approaches can be roughly categorized in two classes: extensions of grammatical formalisms and semantic-based analyzers. As stated above, we will not discuss here the semantic approaches; as regards the other ones, we can say that an extension of a grammatical formalism lends itself rather well to the relaxation o£ some syntactic constraints (e.g. number agreement), less well to others (e.g. the absence of a required constituent) and meets big difficulties in handling ordering problems (out of order constituents). This is obviously due to the fact that formal grammars have the task of describing "strings" of symbols, i.e. objects where the order is fundamental. It is not sufficiently proved that in natural language sentences, in their aspect of information conveying tools, the order of constituents is as fundamental. This observation leads to a last remark: in languages where the order is not as strict as in English (almost free word-order languages as Italian or Japanese) grammars that are not based on the common concept of rewrite rules (mainly related with case systems) are receiving greater and greater attention (Nitta et al., 1984; Sakamoto et al., 1984).

Another problem that should be mentioned is based on psychological motivations. Although this paper is not intended to present a psychologically valid model of natural language processing, we believe that some well known phenomena cannot be disregarded, because they help in making more clear what should (or should not) happen in a N.L. analyzer. The phenomenon we will consider here Is that of garden paths. It gives a hint about the existence of two processing modes in the interpretation of a sentence: normal analysis and backup. At first sight;., this remark confirms the adoption of: standard, run-deterministic parsing methods, where backtracking is a usual technique - On the other hand, the number of times a normal ATN parser (to consider a well known tool) backtracks is not justified by the relative rarity of garden paths. The efforts in the development of deterministic parsing (Marcus, 1980) tried to characterize the normal processing mode, by stating that PARSIFAL would fail to analyze a sentence in cases where a person would garden path. However, it has been shown (Milne, 1982) that the three-constituent-buffer approach adopted by Marcus does not predict with sufficient accuracy the occurrence of garden paths. Again this can be seen as a failure of approaches based only on syntactic knowledge (a grammar-based one - ATN - and a rule-based one - PARSIFAL) to account for a linguistic phenomenon: the solution should be looked for in a more effective cooperation between syntax and semantics.

It is not possible to close this introduction without qoiting a recent papers, wnich addresses some of the problems mentioned above in a tnorough way. In (Schubert, 1984) the section 2.3 deserves attention. It is entitled "Lack of provision for integration with semantic/pragmatic preference principles". What is shown in the paper is that human reaction to sentences that have exactly the same syntactic structure may vary considerably depending on the semantics of occurring words. Although the analysis is carried on in a way different from ours, the concept of "potential" *us* a means to balance the syntactic and semantic information is similar to our weighting of alternative hypotheses.

Another work closely relates with the present one is reported in (Pazzani, 1984). In that case also, the need of a strict cooperation between syntax and semantics is explicitly acknowledged. On trie otrier hand it seems that the absence of a weighting mechanism could make the LAZY parser fail in some cases wnere no clear-cut choice is possible.

In tne second section of the present, paper we will describe the structure of tne syntactic processor induced in the FIDO system. Although the system (which is fully implemented in FRANZ LISP on a VAX 11/730 computer) does not nandle all trie phenomena discussed in this introduction, tne presentation will allow us to clarify trie basic operating principles, in order to describe in the third section the extensions of the parser we are currently implementing.

SYNTACTIC ANALYSIS if III FIDO

FIDO (a Flexible Interface for Database Operations) is a prototype! 1 system that allows tne user to access in N.L. (Italian) the data stored in a relational database. After a previous approach to building natural language interfaces (Lesmo, Magna ru , 'Torasso 1981*)* , we realized that one of the main concerns nad to be to guarantee the portability of the system; this was achieved by adopting a strongly modular approach. Some efforts have been made to develop efficient methods to store semantic information (Lesmo, Siklossy, Torasso 1983) and to optimize the resulting query, expressed in relational algebra (Lesmo, Siklossy, Torasso 1985). The organization of tne parser was described in (Lesmo, Torasso 1983) and its suitability to the analysis of ill-formeel sentences in (Lesmo, Torasso 1984) ; in particular, the extensions introduced to deal with conjunctions are decribed in (Lesmo & Torasso, 1985). We will overview here the basic design choices.

Ihe syntactic knowledge source is composed of a set of condition-action rules, where the condition examines the current status of the analysis, i.e. the parse tree that has already been built, whereas the action extends in some way the parse tree, hypothesizing the attachment point and the syntactic role oi a new constituent. The parse tree is built according to the head and modifier approach and an example is reported in fig.l.

Six node types have been defined; each node label in fig.l has the form TYPEi: the node labelled XXj is the j-th instance of the type XX that has been built during the analysis. Tne types appearing in the figure are: REL (standing for RELation, normally associated with verbs); REF (REFercnts: nouns and pronouns); CONN (CONNectors, mainly for prepositions; it can happen that the filler of the node is UNMARKED: it means that the corresponding verb case is not marked by a preposition); DET (mainly DETurminers). The other node types are ADJ (ADJectives) and MOD (MODifiers, e.g. adverbs).

The syntactic rules are grouped in packets associated with syntactic categories. When an input word is syntactically ambiguous, different packets are activated and all conditions are tested. If just one of them succeeds, then the action is executed and the analysis goes on deterministically. Otherwise, the status of the analysis is saved to allow for possible backups in a subsequent phase, ana the first action is executed (the different rules are ordered manually). Facilities are provided in the lexicon to handle canned phrases (e.g. "di corsa" - on the run) and compound words (e.g. "dammelo" - give it to me). It must be noted that some conditions require a lookahead (2 words maximum); this is done in order to increase the discriminating power of the conditions and to reduce the number of choice points.

In order to give an idea of the control structure of the analysis, let us see what happens when the first word of the example in fig.l is found (we must stress again tnat FIDO works on commands in Italian: we will go on with English examples in order to increase the readability of the paper. In Italian the most direct translation of "which" is "quale", though its use differs slightly when it is used as a relative pronoun).

There are three different lexical entries for "which", each of which is associated with a different syntactic category: QADJ (interrogative adjective), QPRON (interrogative pronoun), and RELPRON (relative pronoun). The analysis begins with an empty REL node (RELI) as current node. Roughly speaking, all conditions of QADJ rules require that the next word is an adjective or a noun, the QPRON rules apply in the remaining cases, whereas the RELPRON rules can be activated just in case a previous REF node can be used as an attachment point for the relative clause. In our case the QADJ

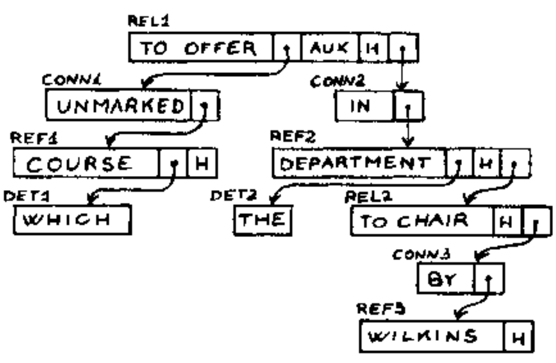

Fig.1 - Parse tree resulting from the analysis of the sentence: "Which courses are offered in the department chaired by Wilkins?".

interpretation is selected and, among the rules of the packet, the one is chosen that applies when the current node is an empty REL node. That rule builds a CONN node (and fills it with UNMARKED), a REF node (and leaves it empty) and a DET node (and fills it with the current word, i.e. "which"). The resulting structure is shown in fig.2. Then, the controller of the parsing process is awaken; it looks for another word and finds it in the lookahead buffer (it was used to discriminate between QADJ and QPRON). The NOUN packet is activated and a rule is selected, which fills the empty REF (REFI) with "course". We leave at this point the example, assuming that it gave an idea about how he analysis of a sentence is carried on.

Instead, some more specific points have to be made clear. The nodes in the figures have been represented very sketchily. Each node is actually a complex data structure, with various slots and some procedures attached to the prototype. For example, a REL node includes slots as HEAD (the verb), FORM (active vs. passive), TENSE, NUMBER, MOOD (indicative, conjunctive, etc.), ROLES (the case frame) and others. Notice that the slot AUX indicates the presence of an auxiliary verb in the sentence. The actual form of the auxiliary is not reported since it can be inferred by taking into account the values of MOOD, FORM, TENSE, etc. The associated procedures are called RELHEADPROC (operations to be done when the HEAD slot is filled, e.g. computing the tense of the verb), RELAGREEPROC (checking the number agreement with the subject), and RELSEPR0C (checking the acceptability of the actual case frame and beginning to build the semantic interpretation). When a node is operated upon, one or more procedures can be scheduled for execution. They can accept or reject the operations done by the parser (syntactic hypothesis). A simple way to make the parser more robust is to relax some of the constraints embodied in the procedures. For instance, an agreement failure can produce just a warning message, without requesting a reorganization of the parse tree, a reorganization which is always attempted in case of semantic failure. Such relaxation techniques can also be introduced in other formalisms, such as ATN (Kwasny & Sondheimer, 1981). More interesting, the proposed formalism handles easily also ordering errors. In fact, the attachment of cases to verbs and of adjectives to nouns is always allowed (in Italian the adjectives can occur both before and after the noun) and only

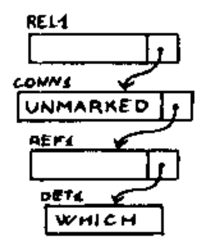

Fig.2 - Partial structure built after the analysis of the first word ("Which") of the sentence in fig.1.

when the node is closed (a node is closed, i.e. it is considered to be complete, when an attachment is proposed to a node above it in the tree) the CHECKORDER procedure verifies that the rules which govern the ordering of constituents are respected. Also in this case, a failure of CHECKORDER results in the issuing of a warning message, without any reorganization of the parse tree.

It has probably been noted the use of the term "reorganization of the tree" in the discussion above. In fact, such modifications, that we call "natural changes" to point out their simplicity and naturalness, are the primary tool for handling non-determinism. The brief presentation of the structure building rules failed probably to make clear one important point: when the action part of a rule is executed, it usually adds a subtree to the current tree; the attachment point of the new subtree is the nearest node of the required type that is above the current node. Of course, this choice is made only on syntactic grounds, so it may happen that it is not acceptable from a semantic point of view. In a standard ATN framework, this problem is solved backtracking: the subnets allowing, for example, PP modifiers include an implicit choice point (in correspondence with the position where a PP could be present or absent) and a semantic failure would involve backing up to such a previous choice point. Although the introduction of some special tools (of the kind of well-formed substring tables or chart parsers) allows the system to avoid the re-analysis of the PP component, some bookkeeping is needed to save the status of the analysis at the choice points. The natural changes mecnanism makes that work useless, in that the choice points are implicitly available in the structure of tne parse tree and can be easily looked for by the modification rules. A further advantage is the high flexibility of the tool: the natural changes are expressed in the form of pattern-action rules (as the standard rules) so that, in principle, an action could restructure the tree in a very complex way. In fact, we use them also to handle some problems related with the analysis of conjunctions (Lesmo & Torasso, 1985) and with some special forms of relative clauses. TVs often happens, the natural changes are actually too powerful; at this time we have not pursued the study of what are the reasonable constraints that must be put on tne operations of the changes. We want to stress, however, that the introduction of the natural changes does not substitute the backup completely: this remark is in agreement with the discussion about the existence of different processing modes in the analysis of N.L. sentences. Although we are not able to state now the correspondence between the use of backup and the occurrence of garden paths, we can notice that the saving of the status is limited (in most cases) to syntactically ambiguous words such that more than one syntactic category is acceptable in the current context: this strongly reduces the number of choice points, as predicted by the garden path phenomenon.

EXTENSIONS TO THE PARSER

Before going on we have to make clear an important point: whereas the parser embodied in FIDO works on Italian sentences, in order to perform the tests that led to the version described in this section, we had to develop a small set of rules for English. The reason why we did this was to have at disposal a wide corpus of thoroughly analyzed examples (i.e. the ones appearing in the referenced papers by Milne and Schubert). This approach to testing has both an advantage and a disadvantage: the adaptability of the parser to a different language is partially demonstrated, but the number of syntactic phenomena that has been taken into account in building the English rules is not very high, so that some ad-hoc solutions could *have* been adopted.

We can now start by seeing what happens when syntactically ambiguous sentences are processed by FIDO (old version). A first example is drawn from Schubert's paper:

(1) John bought the book which 1 had selected for Mary

After the analysis of the first portion of the sentence (as far as the word "selected") the status of the tree is the one of fig.3. Upon encountering the preposition "for", a rule would propose its attachment (in a CONN node) to the node REL2 (i.e. as a verb modifier). The subsequent attachment of a REF node (containing "Mary") to the newly created connector would trigger the semantic check procedures, which give a positive? answer (case frame: TO SELECT; SOBJ: PERSON, OBJ: THING, FDR: PERSON) and allows the system to confirm the proposed analysis.

On the contrary, in the example below:

(2) John bought, the book which I had selected at a lower price

after a sequence of steps analogous to tne one described above (extended to handle the determiner and the adjective), the semantics would reject the syntactic hypothesis. The natural changes would be triggered, the attachment of tne PP to "book" would be tried and again rejected on semantic bases. Finally, the attempt to attach "at a low price" to "buy" succeeds and the analysis is completed.

It is apparent that this process does not work in cases such as:

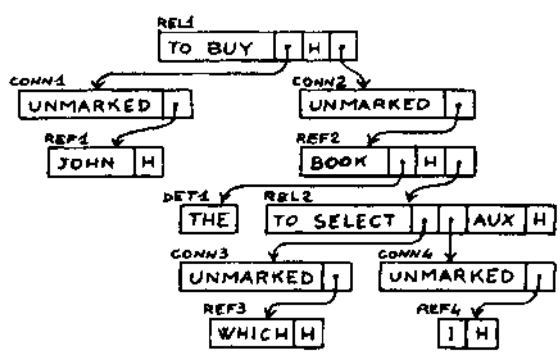

Fig.3 - Analysis of the sentence fragment: "John bought the book which I had selected ...".

(3) John carried the groceries for Mary

The attachment of "for Mary" to "groceries" would be attempted; it cannot be rejected by the semantics, thus the final analysis is the one reported in fig.4 (contrarily to the expected one).

Another example concerns a pair of garden path sentences. In

(4) The building blocks the sun

we obtain the structure reported in fig.5. If the actual input sentence were

(5) The building blocks the sun faded are red

then the analysis would be blocked at "are" (in fact, "faded" would be used as a modifier for "sun"; see (6)) and a backup would make the system build the right structure.

(6) The building blocks the sun faded behind the hills

Again, although the model predicts the correct behavior in the examples above, it fails to characterize the difference between (7) and (8):

(7) The table rocks during the earthquake

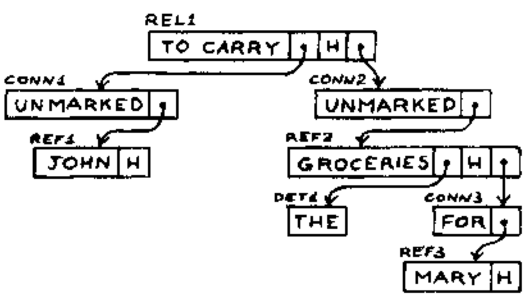

Fig.4 - Analysis of the sentence "John carried the groceries for Mary".

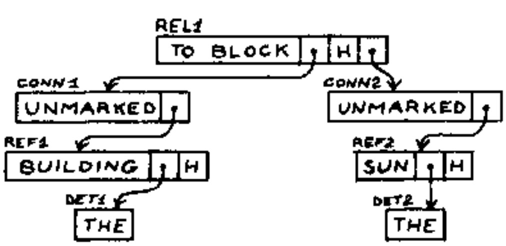

Fig.5 - Analysis of the sentence "The building blocks the sun".

(8) The granite rocks during the earthquake

In these sentences we assume, according to the results of Milne's experiments that (8) makes people garden path, whilst (7) does not. In fact, the only way FIDO has at disposal to express preferences (lexical, in this case) is to take into account the order of the lexical entries. Thus, we are able to say that "rock" is more commonly used in its nominal sense than in its verbal sense, but not to say that (to use Milne's words) "people like granite rocks, but don't like table rocks". In other words, FIDO is not able to express preferences on the basis of the context.

The solution we adopted to overcome the problems mentioned above was to introduce a limited degree of parallelism in the analysis, and to evaluate and update the weights of the different alternatives that are being carried on in parallel, until a definite decision can be taken. Notice that we don't claim that the chosen alternative is always the right one, nor we would like to get such a result: in some cases the system can take the wrong way, as humans do when they garden path.

In implementing such a model, we took advantage of our research on expert systems based on fuzzy production rules (Lesmo, Saitta, Torasso 1985): in fact, the weighting mechanism we adopted is based on possibility theory (Zadeh, 1979). We will not analyze here the advantages of a possibility based approach with respect to approaches based on probability theory or on heuristic methods (the interested reader can see (Lesmo, Saitta, Torasso 1985)), but, as we did in the previous section, we will try to let the reader understand how the weighting mechanism has been merged with the facilities previously available in FIDO. We have only to say that the weight (that, from now on, will be called Confidence Degree - CD) is a value ranging from 0 to 1, where 0 means that the hypothesis cannot be accepted at all and 1 means that the hypothesis is surely the right one.

The basic design choices that must be made concern:
- What alternatives must be considered (how much parallelism should be supported)?
- How long must the different paths be pursued (when should the final decision be taken)?
- What knowledge sources contribute to the weighting (and how can the weights be adjusted)?

It is interesting to note that the structure of the parser described in the previous section gives a hint about the first choice. In fact, two sources of ambiguity are present: more than one rule may be activated at the same time and the attachment point of a constituent is not uniquely determined. Thus, to test the validity of the proposed approach we had to change the basic control structure in only two ways:

1) The different rules whose conditions are satisfied are executed in parallel. Although there is no a-priori upper limit to the amount of parallelism, the constraints embodied in the conditions keep low the number of alternative hypotheses (according to the observation made in the previous section about the reduction of the number of choice points).
2) Instead of waiting for the results of the semantic checks, the natural changes are scheduled as

soon as an attachment is proposed. The natural changes have been modified in a simple way: no detachment is made, but the different alternatives are added to the proposed one (note that this is just a first, low-cost solution: what we are studying now is the possibility to eliminate tiie natural changes mechanism; this can be done if all attachment points can be found at a glance by tne linking procedures).

As regards the second choice (when should the final decision be- taken?) , we decided to distinguish again the sources of ambiguity described above. The reason why different solutions were reasonable stood in the different computational cost of carrying on alternatives: whereas the rule ambiguity seemed to require a real maintenance of different trees, the role (attachment) ambiguity implied only that different links are included in the same tree. In the first case, after trying different alternatives (no lookahead, one-word lookaneacl, just tne lookahead required by tne conditions) we had the pleasant surprise that in most cases the different states had not to be maintained. In fact, the only thing we had to do was to defer the decision about tne rule[1] to apply after the execution of the semantic check procedures: they provide*! the parser with the information about tne semantic preferences that was lacking in the previous version of tne system. As regards the role ambiguity, we let the analysis go on until tne filler of the node which is the root of the attached subtree has been found: this means that we wait until the semantic checks can be done. For example, if sentence (1) were changed into:

(9) *John* bougnt the book tnat 1 selected for the nice blond-haired girl that you know

then the choice would be delayed until the word "girl" is found.

Note that in both cases, though the behavior of the parser is different, the parser pursues different paths until the system allows the semantics to provide it with some evidence about the most reasonable caoice.

Finally, the third problem concerns the knowledge sources involved in the weighting process. Apparently, we had to attach CD's to lexical entries, syntactic rules, and semantic information. On the otiier hand, the semantic information (which in FIDO consists in a semantic net representing the selectional restrictions, see (Lesmo, Siklossy, Torasso 1983)) overcomes the information that could be attached to lexical entries. In fact, the choice is made on the basis of the possibility of attachment of a pair of "concepts": this provides the system with more detailed information than the possibility of occurrence of a single interpretation. That is, if the system knows that CD (TO ROCK, SUBJ: TABLE) = O.8 and CD (ROCK, MODIF:TABLE) = .1, CD (TO ROCK, SUBJ:GRANITE) = 0.6 and CD (ROCK, MODIF .-GRANITE) = 0.9, it can disregard the fact that CD (TO ROCK ICAT:VERBJ) = 0.7 and CD (ROCK ICAT:NOUN]) = 0.9. The solution we adopted is to associate with tne arcs appearing in the semantic net a CD expressing tHE preference of the system. It is not possible here to discuss the details of the implementation (actually, not all arcs have a CD), because such a discussion would require a description of the semantic net. It must be noted, however, that this solution requires the explicit introduction of all possible semantic connections. This is consistent with a database interface, because tne associations carry the information about the correspondence with the database scnema. In a general N.L. understanding system this is quite expensive and some way to propagate the CD's according to the degree of match with the declared selectional restrictions snould be included in the system.

As regards the syntactic Knowledge source, we attached *CD's* to the structure building rules ano we decided to compute tne CD's of the attachment points on the basis of their distance (number of nodes to traverse) from tne "current" node. In particular, the current node is assigned a CD equal to 1 and, for each node that is traversed to fine an alternative, the CD is decreased by a constant factor (currently 0.1). Apart from this latter, all CD's (both in the net and in the rules) have been assigned manually. This allowed the system to succeed on a wide set of examples; of course, a less heuristic determination of CD values would be useful, but it requires a large research effort per

CONCLUSIONS

As Winograd states in (Winograd, 1983), the research on N.L. understanding is being carried on today within a new paradigm: the computetional paradigm. Its main differences with respect to the previous (generative) one, stands in the "attention to process organization" and the "relevance of non-linguistic knowledge".

It is not the aim of this paper to take into account all the problems that non-linguistic knowledge conveys into N.L. analysis, out to make clear that the in-depth understanding of the respective roles of syntactic and semantic knowledge sources and the clarification of the way they interact to construct the interpretation of natural language sentences is fundamental to building N.L. interpreters, we claim tnat neither syntax nor semantics can be assigned the role of "guide" of the interpretation process, but they must operate on a parity basis. Both of them provide the analyzer with information about trie choices that must be made during the interpretation.

The approach outlined in the paper is just a first attempt to satisfy these principles: many problems must be examined and some substantial changes can be introduced, but we maintain the fundamental role of the "rule" concept in tne construction of N.L. analyzers and the necessity of being able to weight the contributions of different knowledge sources: the interpretation is not a categorical (yes/no) process, but it must be based on the idea of preference (Wilks, 1375) (or subjective evidence).

The paper shows how a rule-based approach has been modified to take into account both syntactic and semantic preferences, we hope? to have given a feeling about the ease with which the required modifications were embodied in the previous system. The available space did not allow us to consider same? obier phenomena that FIDO is able to handle quite easily: they concern the analysis of in-formed sentences. Although many aspects of ill-formedness were already handled by the old version of the system, the introduction of CD's and the modifications of the natural changes are useful also to characterize in a more perspicuous way the analysis of conjunctions: also in this case, the CD's are used to compare the different alternatives regarding the role the second conjunct can assume.

REFERENCES

[1] A.A.V.V.: Special Issue on Ill-Formed Parsing, AJCL 9, no.3-4 (1983).

[2] E.Charniak: Six Topics in Search of a Parser. Proc. 7th IJCA1 , Vancouver (1931),1079-1007.

[3] X.Huang: Dealing with Conjunction in a Machine Translation Environment. Proc. COLING 84, Stanford (1984), 243-246.

[4] S.C.Kwasny, N.K.Sondheimer: Relaxation Teon-riiques for Parsing Grammatically Ill-Formed Input in Natural Language Understanding. AJCL 7 (1981), 99-108.

[5] L.Lesmo, D.Magnani, P.Torasso: A Deterministic .Analyzer for the Interpretation of Natural Language Commands. Proc. 7th IJCAI, Vancouver (1981), 440-442.

[6] L.Lesmo, L.Saitta, P.Torasso: Evidence Combination in Expert Systems. Int. J. of Man-Machine Studios, vo1.22 (1985).

[7] L.Lesmo, L.Siklossy, P.'iorasso: A Two Level Net for Integrating Selectional Restrictions and Semantic Knowledge. Proc. IEEE Int. Conf. on Systems, Man and Cybernetics, India (1983), 14-18.

[8] L.Lesmo, L.Siklossy, P.TOrasso: Semantic and Pragmatic Processing in FIDO: a Flexible "Interface for Database Operations. Information Systems 10, n.2 (1985).

[9] L. Lesmo, P.'iorasso: A Flexible Natural Language Parser based on Two-Level Representation of Syntax. Proc. 1st Conf. ACL Europe, Pisa (1983), 114-121.

[10] L.Lesmo, P.'iorasso: Interpreting Syntactically Ill-Formed Sentences. Proc. COLING 84, Stanford (1984), 534-539.

[11] L.Lesmo, P.Torasso: Analysis of Conjunctions in a Rule-based Parser. Proc. 23rd ACL Meeting, Chicago (1985).

[11-] M.Marcus: A Theory of Syntactic Recognition for Natural Language. MIT Press (1980).

[13] R.W.Milne: Predicting Garden Path Sentences. Cognitive Science 6, 349-373 (198;:.).

[14] Y.Nitta et al.: A Proper Treatment of Syntax and Semantics in Machine Translation. Proc. COLING 84, Stanford (1984), 159-166.

[15] M.J.Pazzani: Conceptual Analysis of Garden-Path Sentences. Proc. COLING 84, Stanford (1984), 486-490.

[16] C.R.Perrault (ed.): Special Issue on Mathematical Properties of Grammatical Formalisms. Computational Linguistics 10 (1984), 165-202.

[17] G.K.Pullum: Syntactic and Semantic Parsability. Proc. COLING 84, Stanford (19b4), 112-122.

[18] Y.Sakamoto et al.: Lexical Features for Japanese Syntactic Analysis in MU-Project-JE. Proc. Coling 84, Stanford (1934), 42-47.

[19] D.Sagalowicz: Mechanical Intelligence: Research and Applications. Final Teen. Report, SRI Int., Menlo Park (December 1980).

[20] L.K.Schubert: On Parsing Preferences. Proc. Coling 84, Stanford (1984), 247-250.

[21] Y.A.Wilks: A Preferential Pattern-Seeking Semantics for Natural Language Inference, Artificial Intelligence 6 (1975), 53-74.

[22] T.Winograd: Language as a Cognitive Process. Vol.I: Syntax, Addision Wesley (1983) .

[23] W.A.Woods: Cascaded ATN Grammars. AJCL 6 (1980), 1-12.

24] L.A.Zadeh: A Theory of Approximate Reasoning. In E.Elcock, D.Michie, L.Mikulich (eds.): Machine Intelligence 9, Ellis Horwood (1979), 149-194.