



## Is Nonmonotonic Grammar A Solution to Natural Language Processing?

Christophe FOUQUERE

LIPN-CNRS  
Av J.-B. Clément 93430 Villetaneuse  
tél: (33 1) 49 40 35 93

### ABSTRACT

Natural Language Processing requires flexibility. This statement has considerably influenced Natural Language systems during the last years. Recent studies (3), (5), (15) have suggested that nonmonotonic logic would be an attractive frame. Associated with a bottom-up parse of sentences, it could help to solve ambiguity, to reduce parsing choices, and find corrections when the parse fails. This paper presents a methodology for describing Natural Language grammars in a nonmonotonic first-order logic theory. The general idea is to specify first the most general rules defining the grammatical symbols, then all the exceptional cases. We will expose how this method may help us to solve difficulties, as competence errors. For that purpose, we will use TMS default logic (2) in our examples. Finally, we will argue that this methodology shifts the problems to the definition of the grammar: nonmonotonicity allows us to express additional informations that are needed in several cases, even if at the same time other questions are raising.

### 1 NONMONOTONIC GRAMMAR

The concept of Nonmonotonic Grammar refers to a grammar which has two kinds of rules: *definition rules* and *exception rules*. Recent studies (3), (5), (15) have suggested that nonmonotonic logic would be an attractive frame. Associated with a bottom-up parse of sentences, it could help to solve ambiguity, to reduce parsing choices, and find corrections when the parse fails. This paper presents a methodology for describing Natural Language grammars in a nonmonotonic first-order logic theory. The general idea is to specify first the most general rules defining the grammatical symbols, then all the exceptional cases. The first ones indicate which symbols a grammatical symbol dominates by default: a *sentence* is, by default, the sequence of a *noun phrase* (which will be the subject of the verb) and a *verb phrase*. Exception rules are helpful for completing description of grammatical symbols. If we can express the (default) fact that a *noun phrase* is a *determiner* followed by a *noun*, there must be an exception rule which generates a *noun phrase* with an *adjective*.

Since Reiter's paper in 1980 (13), various theories (corresponding to particular problems) have arisen: conditional logic (1), autoepistemic logic (10). In this paper, we will focus on Doyle's TMS theory (2) and use it to express a toy grammar of Natural Language. In a first part, we will expose briefly the Truth Maintenance System (TMS Doyle theory). We will then expose how the grammar of Natural Language may be described in a nonmonotonic way, and in a short example how sentences are parsed. We will notice later that some of the previous theories would have been inadequate for our purpose.

### 1.1 Doyle's TMS theory

There is no place here to expose all details of Doyle's theory. The reader is invited to refer to (2) for further details. Doyle, as others researchers in default reasoning, increases first-order logic. He uses a modality *out*, expressed syntactically in the following (and most general) way:

$$\alpha(X) \wedge \text{out}(\beta(X)) \rightarrow \gamma(X)$$

There are two kinds of prerequisites: the in-justifiers (as  $\alpha(X)$ ) which roughly correspond to usual prerequisites, and the out-justifiers (as  $\beta(X)$ ) which convey rule inhibition information. Informally, the intended meaning of the rule is the following:  $\gamma(X)$  is inferred from  $\alpha(X)$  iff ( $\alpha(X)$  is true and  $\beta(X)$  is not known as true:  $\text{out}(\beta(X))$  represents the ignorance of the fact  $\beta(X)$ ). These notions are expressed via a labelling of the facts. Facts may be labelled *in* as well as *out*. Facts labelled *in* will finally be considered as true. So the labelling is done under the following conditions:

- Ground facts are labelled *in*.
- For each rule
  - the conclusions are labelled *out*
  - and at least one of the in-justifiers is labelled *out*or
  - the conclusions are labelled *out*
  - and at least one of the out-justifiers is labelled *in*or
  - the conclusions are labelled *in*
  - and the out-justifiers are all labelled *out*
  - and the in-justifiers are all labelled *in*

A labelling that satisfies the previous conditions is said to be a **well-founded labelling**. Only well-founded labellings will be of interest for our purpose. We will show in the examples below how this can express effectively default choices done during the parse stage.

### 1.2 Natural Language nonmonotonic processing

The partition of prerequisites into in- and out-justifiers allows us to describe the grammar with two levels of knowledge: grammatical symbols are defined with general and exceptional rules. Supplementary rules will help us to logically express inhibition.

We will suppose that grammar is given with context-free rules for sake of clarity, but a context-sensitive grammar or a unification grammar would have been treated in the same way:

Let us have rules like the following one:

$$A_1, A_2, \dots, A_n \rightarrow S$$

where  $A_i$  and  $S$  are grammatical symbols.

Suppose that  $\mathbb{F}_S$  is the set of all rules that have  $S$  as their conclusion. Let us have a partition of each set  $\mathbb{F}_S$  into two sets  $\mathbb{D}_S$  and  $\mathbb{E}_S$ . We then define  $\mathcal{D}_S$  and  $\mathcal{E}_S$  respectively from  $\mathbb{D}_S$  and  $\mathbb{E}_S$ .

Rules in  $\mathbb{D}_S$  (resp.  $\mathbb{E}_S$ ) will be called default rules (resp. exception rules). By extension, the same expressions will be used for  $\mathcal{D}_S$  and  $\mathcal{E}_S$  respectively.

- For each rule in  $\mathbb{D}_S$   $A_1, \dots, A_n \rightarrow S$   
write a rule in  $\mathcal{D}_S$ :  $A_1 \wedge \dots \wedge A_n \wedge \text{out}(R_S) \rightarrow S$
- For each rule in  $\mathbb{E}_S$   $B_1, \dots, B_p \rightarrow S$   
write the rules in  $\mathcal{E}_S$ :  $B_i \rightarrow S$   
 $B_1 \wedge \dots \wedge B_p \wedge \text{out}(R_{B_i}) \rightarrow S$

$R_S$  is the predicate which will inhibit the use of corresponding default rule (specified in  $\mathcal{D}_S$ ), when exception rules have to be used (specified in  $\mathcal{E}_S$ ).

We give now a very simple example of what may be done:

- (1)  $\text{np}(\text{subject}, [X_1, X_2]) \wedge \text{vp}([X_2, X_3])$   
 $\wedge \text{out}(r_{\text{sentence}})$   
 $\rightarrow \text{sentence}([X_1, X_3])$
- (2)  $\text{determiner}([X_1, X_2]) \wedge \text{noun}([X_2, X_3])$   
 $\wedge \text{out}(r_{\text{np}})$   
 $\rightarrow \text{np}(T, [X_1, X_3])$
- (3)  $\text{verb}([X_1, X_2])$   
 $\wedge \text{out}(r_{\text{vp}})$   
 $\rightarrow \text{vp}([X_1, X_2])$

( $x_i$  refer to indices in text, other words are constants or predicates in the logical language. Variables have an upper letter as first character, otherwise they are constants).

The three previous rules are the default ones, that describe (rule 1) a sentence as a noun phrase (subject) followed by a verb phrase, and (rule 2) the noun phrase with a determiner followed by a noun, and finally (rule 3) a verb phrase with a verb.

The rule 2 is not valid when there is an adjective, so the set  $\mathcal{E}_{\text{np}}$  has the two following rules:

- (4)  $\text{adjective}([X_1, X_2]) \rightarrow r_{\text{np}}$
- (5)  $\text{determiner}([X_1, X_2]) \wedge \text{adjective}([X_2, X_3])$   
 $\wedge \text{noun}([X_3, X_4])$   
 $\wedge \text{out}(r_{\text{npadj}}) \rightarrow \text{np}(T, [X_1, X_4]).$

A **nonmonotonic grammar** is the union of sets  $\mathcal{D}_S$  and  $\mathcal{E}_S$  for all grammatical symbols  $s$ .

How is a sentence parsed?

**Definition:** Given a nonmonotonic grammar, a sentence will be said **correct** iff the predicate *sentence* has a labelling *in*, for a well-founded labelling (i.e. that has the property given above), where ground facts are the lexical informations extracted from the sentence.

For instance, the well-founded labelling of the sentence "the baby cries" is determined in the following way:  
(we suppose the lexical informations (ground facts):  
 $\text{determiner}([1, 2]), \text{noun}([2, 3]), \text{verb}([3, 4])$   
are labelled *in*)

the in-justifiers of rule 2 are *in*, so, suppose  $r_{\text{np}}$  is *out*,  $\text{np}(T, [1, 3])$  is labelled *in*,  $\text{verb}([3, 4])$  is labelled *in*, so (rule 3), suppose  $r_{\text{vp}}$  is labelled *out*,  $\text{vp}([3, 4])$  is labelled *in*,  $\text{np}(\text{subject}, [1, 3])$  ( $T$  has the value *subject*) and  $\text{vp}([3, 4])$  are labelled *in*, so (rule 1) suppose  $r_{\text{sentence}}$  is labelled *out*,  $\text{sentence}([1, 4])$  is labelled *in*. Rules 4 and 5 involve that the in-justifiers are labelled *out* in order to have a well-founded labelling.

On the contrary, with the sentence "the young baby cries", the fact  $\text{adjective}([2, 3])$  is labelled *in* so (rule 4)  $r_{\text{np}}$  is labelled *in* and  $\text{np}(T, [1, 2] \cup [3, 4])$  is no more labelled *in* for a well-founded labelling. Rule 5 involves that  $\text{np}(T, [1, 4])$  is labelled *in*, so is  $\text{sentence}([1, 5])$ .

### 1.3 An automatic building method

We give now a means to determine the sets  $\mathbb{D}_S$  and  $\mathbb{E}_S$  that are needed to construct the nonmonotonic grammar. Other methods have been investigated in another paper (see (6)): in particular, a statistical method could be well suited for our purpose. In the third part of this paper, we will study what is really taken into account, and what should still be done with meta-knowledge. In fact, we will argue that our method is a means to include some sort of meta-knowledge (specific one) in the actual construction of the grammar (and not outside the grammar, in an additional system to the parser).

The problem of writing a nonmonotonic grammar is to define for each grammatical category  $s$ , the sets of default rules  $\mathbb{D}_S$  and exception rules  $\mathbb{E}_S$ . In fact, the set  $\mathbb{E}_S$  may be deduced from the set of default rules  $\mathbb{D}_S$ , the complementary of  $\mathbb{F}_S$ . So the problem is to make choices in order to extract from all rules that have the same consequent (a grammatical category  $s$ ), a reduced set of default rules.

• Choosing all rules amounts to do nothing. In this case, we have  $\mathbb{F}_S = \mathbb{D}_S$ , and  $\mathbb{E}_S = \emptyset$ . Because default rules are inhibited by predicates that are conclusions of rules of  $\mathbb{E}_S$  (predicates of the form  $r_s$ , see definition of these sets), no facts (that are constructed with these predicates) will be labelled *in*:

facts are labelled *in* if and only if

- they are ground facts (but  $r_s$  are not lexical categories so they cannot be ground facts),
- or they are conclusions of rules such that in-justifiers are labelled *in* and such that out-justifiers are labelled *out* (i.e. rules of  $\mathbb{E}_S$ , empty set in this case).

Exceptions must be explicitly mentioned in 'default' rules as in the following way:

Let us have the two following rules that define the grammatical category  $s$  (it would be the same for any number of such rules):

- $$A_1, \dots, A_n \rightarrow S$$
- $$B_1, \dots, B_p \rightarrow S$$
- write rules in  $\mathcal{D}_S$ :
- $$A_1 \wedge \dots \wedge A_n \wedge (\neg B_1 \vee \dots \vee \neg B_p) \rightarrow S$$
- $$B_1 \wedge \dots \wedge B_p \wedge (\neg A_1 \vee \dots \vee \neg A_n) \rightarrow S$$
- (assuming  $A_i$  and  $B_i$  are different)

There is no more out-justifiers, and exceptions are explicitly introduced in rules. In a PROLOG paradigm, other formulations would have been possible (simpler ones of course), but our remarks would have been the same:

In the previous way, the relations between rules (via the predicates) are apparent (so we can use them in a correction system, see below) but the parse of sentences is more difficult than it is usually.

In a PROLOG formalism (see (12) for a description of DCG formalism), relations are no more apparent (so the correction process involves knowledges that have to be declared outside the grammar)... but the parse process is efficient.

• Let us propose a method based upon a *partial order relation* among the rules which have the same consequent in order to extract automatically the set of default rules (the set of exception rules is deduced from this one).

**Definition:** (partial order relations on rules) If two rules  $R$  and  $R'$  are in  $\mathbb{F}_S$ , let

- $$R: A_1, \dots, A_n \rightarrow S$$
- $$R': B_1, \dots, B_p \rightarrow S$$

then  $R \ll R'$  iff  $\{A_1, \dots, A_n\} \subset \{B_1, \dots, B_p\}$

**Definition:** (default rules) The default rules of  $\mathbb{F}_S$  will be the minima of  $\mathbb{F}_S$  according to the previous partial order relation.

**Example:** The minima of  $\mathbb{F}_{np}$  will be  $noun \rightarrow np$ ,  $pronoun \rightarrow np$ ,  $proper\_noun \rightarrow np$ . The other rules which have  $np$  as their conclusion, will be treated as exception rules: this is the case when the syntactic structure of  $np$  includes a determiner, adjective(s) or/and relatives.

The previous method is a simple one that allows a translation of a grammar to a nonmonotonic one. It is not obviously linguistically motivated. But, we have to notice that that corresponds to keep on (as default structures) minimal part of allowed syntactic structures (and not necessary the more linguistically motivated). This minimal part may be considered as key words in the sentence from which the other words are parsed as *modifiers*: other words involve the use of exception rules and consequently change results of the parse. If an error occurs in a sentence, the conclusions are reached with the only words concerned by default rules: the method is detailed in the text part of this paper.

## 2 COMPETENCE ERRORS

In this paper, we will focus on the correction of competence errors. But we have to notice that nonmonotonicity has been used for various purposes. Dunin-Kuplicz (3) has used it to reconstruct coreferential structure. The author exhibited the fact that in some languages coreferences are syntactically determined completely or at least preferentially. The author used Lukaszewicz' logic (11) in order to express these rules. Zernik and Brown (15) have previously investigated default reasoning in natural language processing. But they have not brought effective solutions to error problems. We will now indicate how competence errors may be solved in a nonmonotonic grammar.

There is a **competence error** when the correct grammatical rule is not known by the user: the syntactic structure is a special grammatical case for which a special rule is needed. We will then suppose that the user knows the general (i.e. default) rule. It is the case for example in language learning situations: the user learns a language first via general syntactic structures (i.e. default grammatical rules), and so does not know rules that have to be applied in particular situations. If such an error occurs in a sentence during the parse process, the user applies the default rule instead of the exception one: the application of the default rule needs the out-justifiers to be labelled *out*, although they are labelled *in* (inhibition of the default rule when an exception occurs). An example may help the reader to understand where the parse is halted and what must be done in order to correct the sentence.

"J'aime toi." is not a correct (french) sentence (approximately "I love at you", the french correct one is "Je t'aime"). In order to correct it, the system must have the knowledge that a rule is substituted by another in the case of pronoun complement. The nonmonotonic frame enables us to have this knowledge:

- (6)  $vp([X_1, X_2]) \wedge gn(object, [X_2, X_3])$   
 $\wedge out(r_{pronoun})$   
 $\rightarrow sentence([X_1, X_3])$
- (7)  $toi([X_1, X_2])$   
 $\rightarrow gn(T, [X_1, X_2])$   
 $\wedge pronoun(T, [X_1, X_2])$
- (8)  $pronoun(object, [X_1, X_2])$   
 $\rightarrow r_{pronoun}$
- (9)  $pronoun(object, [X_1, X_2]) \wedge vp([X_2, X_3])$   
 $\rightarrow sentence([X_1, X_3])$

During the parse stage,  $r_{pronoun}$  (rule 8) inhibits the use of

rule 6. But the word order in the sentence does not allow the use of rule 9. A correction of the previous sentence could be done in the following way (backward assumption):

Because prerequisites of rule 6 are true, and its consequence has to be true, then we can deduce that (default) rule 6 was applied wrongly. So we assume that the writer wanted to use an object pronoun with the french verb 'aimer'; we then correct the sentence according to rule 9 with an object pronoun 'te' (same number as the wrong pronoun). We obtain the following french sentence: "Je te aime.", and after elision of the vowel e: "Je t'aime."

Now we are able to determine the conditions the default logic has to satisfy for the parser to be efficient and to allow correction. As we noticed in the previous example, backward assumption is necessary to find what parse agrees with the user's sentence, and what sentence with the user's will. For the first remark, the parse must go on even if such an error occurs: the sentence will be parsed if  $sentence([1, n])$  is labelled *in* (where  $n$  is the last index of the sentence). What is the diagnosis of such a phenomena? A rule (namely rule 6 in the previous example) has all its in-justifiers and conclusion labelled *in*, and its out-justifiers too (because of rule 8), so the labelling is no more well-founded! The first stage correction process consists in no more taking into account the out-justifiers: the sentence is (wrongly) considered correct.

The second stage correction process consists in generating the correct sentence according to what the user wanted: rule 9 is the only one which takes into account the pronoun, so the correction is to move the pronoun from after to before the verb.

Backward assumption is correctly solved in TMS. Furthermore, TMS is a simple manner to use default reasoning because of its straightforward inference system.

## 3 IS NONMONOTONIC GRAMMAR A SOLUTION ?

Does this nonmonotonic approach solve completely combinatorial and explanation problems of Natural Language Parsing?

No. But it could be a step toward the solution.

Default reasoning systems allow only to express more information than with first-order logic theory. We have shown in part 2 that these informations are necessary to solve competence errors. Other problems would have been solved in the same way: informations encoded in syntactic default rules increase the expressive power of the parser without adding to the system a specific subsystem, the task of which is to simulate this process.

These problems have been shifted from the parse stage (ambiguity, search of the meaning of an ill-formed sentence) to the grammar definition stage. At the end of part 1, we exposed a method in order to determine default rules automatically. We can say that expressing relations amongst rules inside the grammar is a new research work.

In some correction systems (8), (9), specific correction rules were defined, and the correction system used them when the (normal) parser failed. There were no explanation about these rules, although they succeeded: no explicit relation between grammatical knowledge and correction knowledge. In (4), Fay and Coulon tried to determine different levels of knowledge inside each grammatical rule, (they studied agreement errors). Constraint relaxation was performed automatically: when the parser fails, the least important constraints are omitted; when the parse is complete, these omitted constraints are used to determine which correction has to be done. In this paper, we wanted to extend their approach to syntactic structure correction and propose a general model for doing that.

#### 4 CONCLUSIONS

Besides the correction process, it has to be noticed that linked rules (in contextual discontinuous grammars, see (14) for further details) may be simulated in nonmonotonic grammars (6): links between such rules are done via out-justifiers. We are now investigating Gazdar's grammar formalism (Generalised Phrase Structure Grammar, see (7)). In fact, Gazdar uses 'default rules' in order to express the fact that some features have a default value: in this way, the description of rules is simpler and is straightforward in a nonmonotonic logical frame. Gazdar uses too the 'metarule' approach to describe general linguistic principles: so there is metarule for passive voice, extraposition, and so on. But, some linguistic structures cannot be expressed with the metarules defined in the grammar: the frame of default and exception structures may help to describe these difficulties. The previous remarks show that this (nonmonotonic) approach seems to be convenient for linguistic difficulties too, and not only for a correction process. We think that the development of this research will give interesting solutions.

We so prove that this new approach is an attractive one that enables the grammar writer to include extra-syntactic knowledges, necessary in a lot of processing situations.

Note: This work has been sponsored by French program on Man-Machine Communication: PRC-CHM.

#### REFERENCES

- (1)DELGRANDE J.P.  
An approach to Default Reasoning based on a First-Order Conditional Logic: Revised Report  
Artificial Intelligence, Vol. 36, 88
- (2)DOYLE J.  
A truth maintenance system  
Artificial Intelligence, Vol. 12, 79
- (3)DUNIN-KEPLICZ B.  
Partial Reconstruction of Coreferential Structure of Discourse  
Proc. of ECAF'88, Munich, RFA
- (4)FAY C., COULON D.  
Expertise de correction orthographique par une analyse progressive des phrases  
Proc. of COGNITIVA'87, Paris, France
- (5)FOUQUERE C.  
Un modèle pour la correction de phrases: une Grammaire à Configuration Minimale  
Proc. of ARC International Congress, Toulouse, France, 88
- (6)FOUQUERE C.  
Grammaire Non-monotone et Langage Naturel  
Proc. of AFCET-RFIA, Paris, France, 89
- (7)GAZDAR G.  
Phrase structure grammar  
in P. Jacobson and G. Pullum (eds), The Nature of Syntactic Representation, Reidel, 82
- (8)HAYES P.J., MOURADIAN G.V.  
Flexible parsing  
18<sup>th</sup> Annual Meeting of the ACL, Pensylvanie, USA, 80
- (9)HAYES P.J.  
Entity-oriented parsing  
Proc. of COLING'84, Stanford University, USA
- (10)KONOLIGE K.  
On the Relation between Default and Autoepistemic Logic  
Artificial Intelligence, Vol. 35, 88
- (11)LUKASZEWICZ W.  
Minimisation of Abnormality: A Simple System for Default Reasoning  
Proc. of ECAF'86, Brighton
- (12)PEREIRA F.C., WARREN D.  
Definite Clause Grammars for Language Analysis- A Survey of the Formalism and a Comparison with Augmented Transition Network  
Artificial Intelligence, Vol. 13, n° 1-2, 80
- (13)REITER R.  
A Logic for Default Reasoning  
Artificial Intelligence, Vol. 13, n° 1-2, 80
- (14)St-DIZIER P.  
Contextual Discontinuous Grammars  
Natural Language Understanding and Logic Programming II,  
Dahl & St-Dizier Ed., 88
- (15)ZERNIK U., BROWN A.  
Default Reasoning in Natural Language Processing  
Proc. of COLING'88, Budapest, Hungary