THE LINGUISTIC ANALYSER OF A SMART GATEWAY

Dominique Girollet, Bernard Victorri

ELSAP, URA 1234 du CNRS
Université de Caen

Introduction

We describe in this paper the linguistic unit of a smart gateway.

This gateway, named «anté-serveur», has been carried out by a multi-disciplinarian team in an implementation goal : researchers (from the linguistic laboratory ELSAP and the computer science laboratory LAIAC of the University of Caen) and engineers (from TRIEL, Boulogne and CERIA, Caen) in artificial intelligence, information science, and linguistics have been cooperating for two years to build an operational product.

1. The Working Environment of the Linguistic Analyser

1.1. Why User Friendly Database Interfaces ?

The idea of a smart gateway is a challenge in the on-line information systems domain, because it aims at providing users with the efficient user oriented software aids.

Databases are increasing, while their languages and access means are more and more diversifying : now there are thousands of information databanks available for on-line retrieval searching, which are housed on about ten hosts, each of them offering its own command language, in addition to several thousand more local automatic information systems (viewdata services especially) that demand many different directions of use (Benharrat, 1989 ; Chaumier, 1986).

Subsequently users need real natural language interfaces they can have command of even when neophytes.

Manufactured systems do exist in the area of relational database management systems, especially around SQL —Structured Query Language—, like the Swedish paraphrasing SQL system described in Ljungberg, 1991. But these projects deal with a peculiar database information world, where the answer for a request is one-to-one.

Counter to an on-line system. Beside the above mentioned technical barrier, other three outstanding disadvantages arise in automatic searching (for a detailed review, see Le Crosnier, 1990 ; Dachelet, 1990) : the first one regards the search processing, a «par essai et par erreur» one (Le Crosnier, 1990, p.17), which matches blurred (non structured) information with the user's specific information need. To fomulate the query lays an other issue down, since information retrieval systems are generally based upon set logic ; but boolean connectives operate far from linguistic ones, so that their handling implies training as well as practice. The last obstacle comes from the boolean implicit assumption of equal weightness of each term of the question : because the boolean formula divides the database in two discrete subsets (keys absent or present), the system obeys an all-or-nothing logic and only yields a given number of records. Without any information about how to reformulate the first query to carry on a second more useful search, supposing the user receives too many records or none at all. Namely an information retrieval system has to overcome natural language and feed-back barriers.

Some work is being performed in this sense —state-of-the-art in Salton, 1988—, though remaining at lower levels of spelling, phonema, or morpheme or using rough boolean logic : for instance the French information retrieval system SPIRIT (Systex, France) compounds probabilistic and linguistic processings for indexing and inquiry, while such information management softwares as DYNATEXT (Les Logiciels Guepard'NC, Canada) or LOGOTEL (Chemdata, France) tackle lexical analysis and boolean querying (for a comprehensive survey of French and international industrial supplies, see Abbou, 1989 and EC2, 1991).

1.2 Smartness for a Gateway

The novelty of our anté-serveur lies in securing shrewd automatic database querying. It is based upon two kinds of expertise, an information one and a linguistic one, for the purpose of solving the aforesaid problems of databases selection, request understanding and search proceeding on facilities.

Concerning its technical capabilities, the gateway chooses adequate databases and hosts (the user fills up the field form DOMAIN and receives a list of bases and a list of suggestions he may prefer whenever he doesn't agree with the first set forth names), affords their simultaneous interrogation and makes available the whole range of records management services (their display, their printout, their uploading and the storing of all data needful for payment).

Concerning its linguistic and A. I. capabilities, the gateway analyses the user's request : for each natural French-language query generation of several boolean formulations and translation into the language proper to the selected database (s). It also assists search strategy : on the one hand return of the database (s) answers, sorted according to their proximity degree with the question and accompanied with the indication for each formulation of the gained number of documents ; on the other hand presentation of a list of tracks to continue the research, obtained from a frequential sorting upon the descriptors in the accurate bases.

The anté-serveur thus favours interactivity in information search sessions : its intelligence results from its ability to cope with information retrieval « (...) as a Communication Process.», taking into account the variability of language, which is linked to indexing as well as to inquiry process (Blair, 1990, p.188). Its effectiveness, as we will below expose, crucially depends on the linguistic analysis of the first user's request.

1.3. The Anté-serveur Architecture

A general outline of the anté-serveur is given by figure 1 :

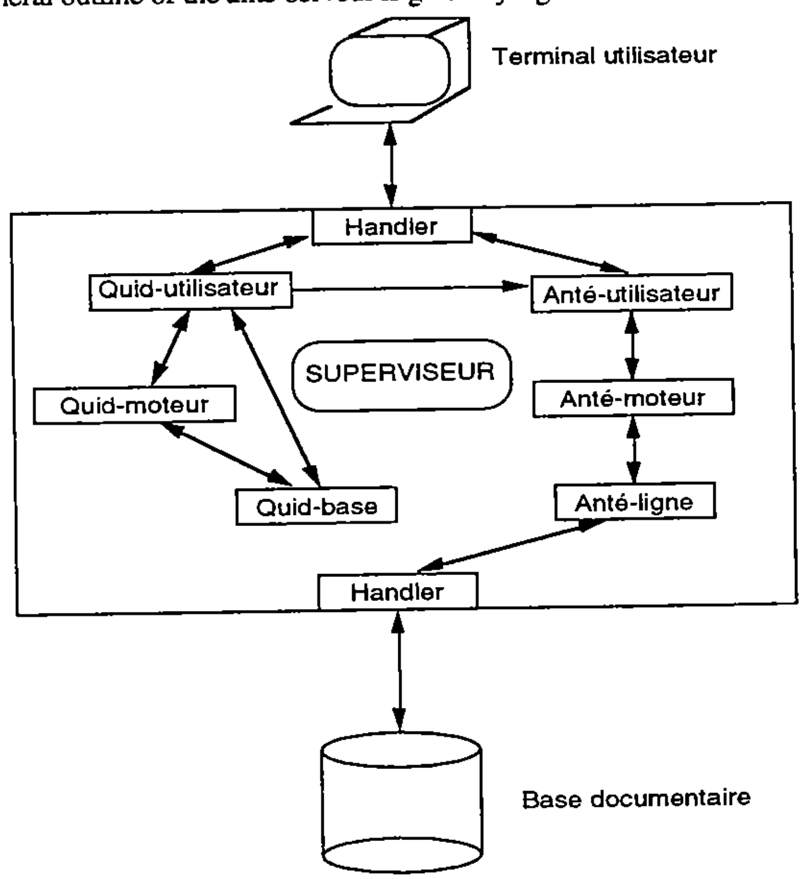

Fig. 1 : Anté-serveur architecture

Six units make it up, all of them were programmed in C++, using LEX and YACC tools in a UNIX environment. The system runs on two kinds of hardware —PC-AT and SUN— and three kinds of user interfaces —SUN work stations, Videotext and smart terminals— (for a technical panorama of the anté-serveur architecture, the reader will refer to Victorri, 1992).

This unit-architecture (six UNIX processes, each specialized in one task and independent of the others as shown in figure 1, one controlling superviseur and two connection handlers) fulfils the flexibility exigency of such a gateway : thanks to its autonomy, every module admits adpatations to specific configurations (new host, new data link, and so forth) without modifying the remainder of the software.

2. Purpose of the Linguistic Analyser

2.1. The Analyser as a Key Unit

The linguistic analyser plays a major part in requests processing for two reasons at least : first, it allows a free human language expression of user's information need ; second, it is to command the efficiency of the search strategy module by producing a semantico-logical representation from which the role and importance of each term in the query can be evaluated. This knowledge enables the strategy module to succeed in ranking the boolean formulas that can be tried to query the databases in accordance with their relevance scale (1.2.). This mechanism is illustrated in 5.

2.2. Specifications of the Linguistic Unit

. INPUT : A free expression in a specified field form

. OUTPUT : Its semantic and logical representation.

As a benefit from the gateway usability, the user enters his question without any limit upon its expression : neither restricted vocabulary and syntax as in programming or command languages, nor restricted domains (a hundred accessible databases).

The request gets however some pre-organization through its sharing into functional divisions (or field forms) : from the spatial point of view, as they display as early as the data entry stage ; from the thematic point of view, as they symbolize the topical functions the various parts of the question fill. These conceptual areas are for example AUTHOR, LANGUAGE, DATE, and SUBJECT. In the SUBJECT field, the user can thus enter the following request :

(1) *Méthodes de détermination d'éléments traces, notamment de métaux lourds par spectroscopie d'absorption atomique dans les corps gras (beurre ou margarine).*

From this general purpose context, the linguistic parser should return a formal description of the request. A semantic sketch (which term (s) do (es) its topic point out ? wich one (s) its meta-questioning preliminary ?) is enriched with a logical architecture (arranging the latter labelled branches with improved boolean connectives, and other connectives such as EXP to bind several terms into one uniterm).

We shall here enlight the key feature of ANAGRAM, its information retrieval accomodation in a domain-independant scope. Its semantic and logical sketches meet requests understanding and querying process needs : they don't refer to any deep argument or conceptual graphs —which are frequently directed at in natural language parsers, like PLUS (Segond, 1992), or like the case model analyzer for technical texts TANKA (Copeck, 1992)—. They formalize grammatical and boolean functions (below, 4 and 5).

A tree diagram may describe this semantico-logical result. Such a representation is generated for (1) in figure 2 :

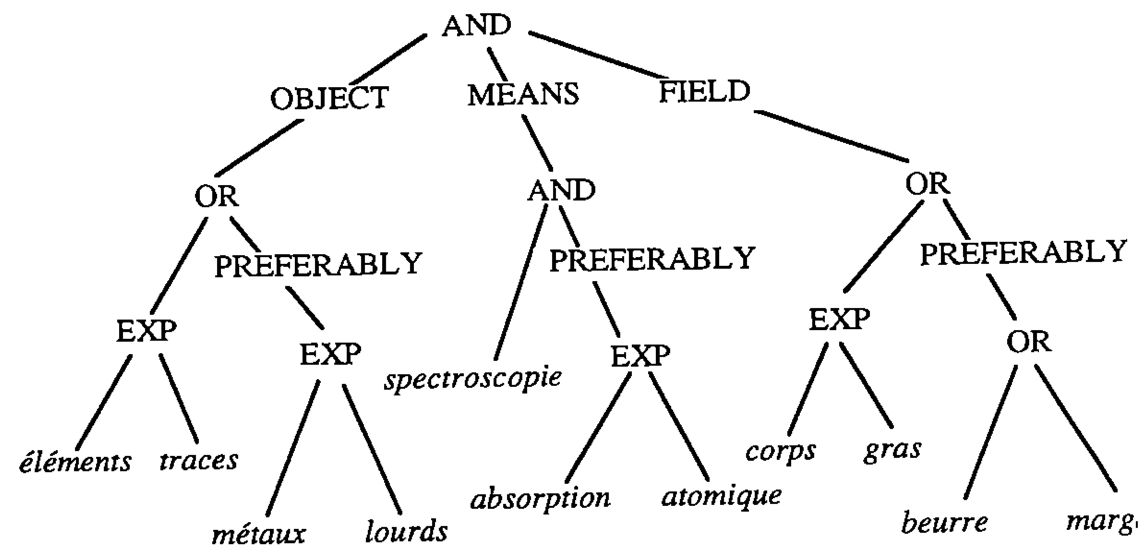

Fig. 2 : Semantico-logical tree

3. The Corpus

Before we delve into the natural language processing, let's have a look on the experimental corpus. Only the very field form SUBJECT interests it, because it receives the most complex sentences.

3.1. Its Gathering

The corpus is made of one basic group of sentences and two enlarged ones.

The basic corpus is a sample of thirty-five genuine recent (1991) requests entered by researchers and students at Caen University Scientific Library. The protocole was the following : they were asked to enter their information need via a terminal, knowing that a human specialist would read their request, conduct the research on fitting databases, and give them back the results.

This corpus was supplemented with :

. seventy formally analogous expressions (headings of tables of contents in books and titles of conferences programmes)

. seventy variants which modify semantic and/or surface syntactic original models to bring out linguistic behavior marks (what does changing determiners involve in the coordination mechanism ?).

Such a technique (complementary sources around a genuine one) bestows check value upon the corpus (no arbitrary extrapolations and focusing on the most typical expressions) and material for investigation.

3.2. Linguistic Description

Items do vary in length and complexity, ranging from the shortest one :

(2) *réseaux neuronaux,*

to extended and intricated ones like (1) or

(3) *études générales sur l'influence du vide ou d'une atmosphère modifiée sur la conservation en emballage du poisson frais ou cru.*

Because all of these sentences ask for information, hence focus on topic, they similarly take on a typical structure pattern :

• rarely explicit meta-questioning formulas, and however in a prevailing form (meta-questioning nouns like *études* or *articles*) + optional adjective + (OBJECT preposition like *sur*) :

(4) *Articles généraux sur l'osmorégulation (osmotic stress) dans les cellules halophytes isolées*

• nominal phrases with various span. Adjonctions generally consist of prepositional phrases (PP) in turn splittable up or not : see (1) and

(5) *Etude des polysaccharides de l'algue Gigartina et leur influence sur la croissance.*

Other two models sometimes occur : relative clauses, such as

(6) *un cours qui récapitule les connaissances de base en chimie cellulaire*,

and particpials, such as

(7) *ensemble des études anglaises ou françaises portant sur la standardisation du ph des laits de fromagerie par des acides organiques notamment acide carbonique*

and

(8) *Système composé d'une banque centrale et d'une banque commerciale*

• generic expressions (no determiner or generic articles) : see (1), (2), and so on

• typographical landmarks (especially comas, brackets and dashes), which reveal important visual cues for the text comprehension, like in (1) and (4).

3.3. Linguistic Issues

No domain specialization, field forms technique and robustness (against errors, unknown words or agrammatical text) entail a stock-saving morphological model : an incomplete dictionary with a set of predictive rules.

Morphological problems hence appear. After detecting an unstored string, the lexicon must classify it : derivational indications (suffixes, roots) are necessary for this task.

Syntax of prepositional complements and coordinated or appositive sequences also makes difficulties.

On the one side, what extent may the parser deduce for prepositional expansions ? Their mobility engenders the two opposite traps of eliminating syntactic ambiguity by declaring for one reading solution, or of creating spurious ambiguity by assuming every solution (even wether not significant). So notice that any attachment option for *dans les corps gras* fits a valid reading of (1), while *dans les cellules halophytes isolées* in (4) accepts none but its attachment to *l'osmorégulation (osmotic stress)*.

On the other side, superficial constructions of conjuncts don't often mate with deep ones ; likewise appositions hide under identical shapes heterogeneous meanings. The parser must recognize them in both cases.

4. Design of the Linguistic Analyser

4.1. Processing Philosophy

Two chief starting-points determined it : the nature of the application aimed at ; the nature of the input. Automatic natural language processing in an information retrieval world directed towards a unit allotement method and a predominantly syntactic-driven parsing, those two conditions here being satisfied of sturdy degree of recognition and strong hypotheses on grammatical characterization of inputs —to compare with a similar approach in a terminological units extraction tool, read Bourigault, 1992—. On-line system frame called for internal formalization suitable to boolean interpretation.

Four components make the parser up ; each of them takes place into a four-stepped linguistic analysis (diagram figure 3) :

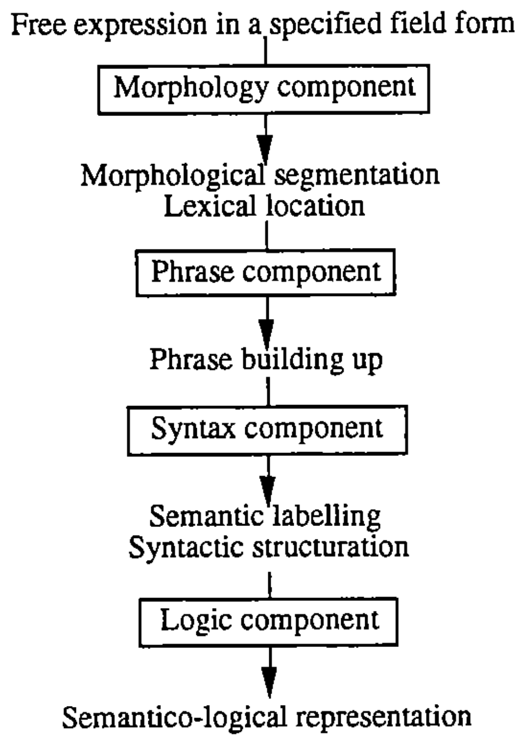

Fig. 3 : Parsing architecture

4.2. Morphology Component

It achieves two tasks (all the more thorny as it has an incomplete glossary in hand) : breaking up an external form in basic units ; feeding these entries with morphological and lexical knowledge.

What then are basic units ? Any punctuation (except hyphen because of compounds) and any character string between blanks or punctuation. As an example, twenty-five units for (1), with two elided *d'* and two items for brackets.

The morphological analyser processes the formal attributes of these units to extract sense of them. It essentially stores function words (over fifty per cent of the two thousand entries) and a selective basis of lexical units ; it attaches to recorded items grammatical (part of speech, nominal and verbal endings, construction) and semantic traits (semantic value of the topical roles played by phrases in the sentence and logical value born by some structural relationships) : «meta-questioning sequence» (META) for *ensemble des études anglaises ou françaises* in (7) ; «poor information content» (VIDE) for *methodes* or *determination* in (1) ; «linking with emphasis on the second element» (PREFERABLY) for *notamment* in (1).

The composition rules that supplement these lists avoid costly lemmatization methods, as well as stochastic part of speech tagging ones (Church, 1988 ; Strzalkowki, 1992). Jointly they solve morphological obstacles (as reported in 3.3.) in an economical manner :

• to treat an unknown unit, the morphology component deduces its grammatical category from its terminal combining form and position, which helps the phrase component to assign these words the right syntactic role (theoretical grounds and an experiment in Ponamalé, 1987 and Caradec et Saada, 1982). For instance in (1) *absorption* rejoins the noun category and *atomique* the adjective one on the basis of their suffixes and surroundings

• as to ambiguous segments, an alternate procedure shows off any potential path, with an exception for conjunctive phrases some transformational rules isolate right now (Gross, 1988 for the linguistic foundations of this method). But it doesn't remove ambiguity, the phrase component will make it one's business to keep the appropriate reading :

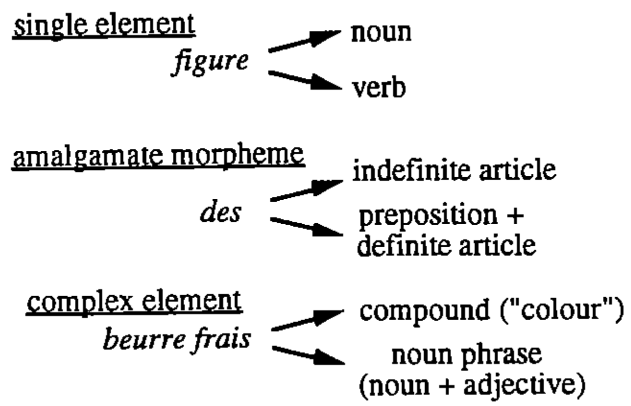

Fig. 4 : Ambiguities

4.3. Phrase Component

This unit intervenes on the previous output by taking a linear lexical basic units string through an immediate constituent group. The sentence description interests the superficial level structure.

The output designs a flat «rake» where nodes are immediate constituents as withdrawn from left to right. In (1) for instance, no order about prepositional phrases anchorage (figure 5) :

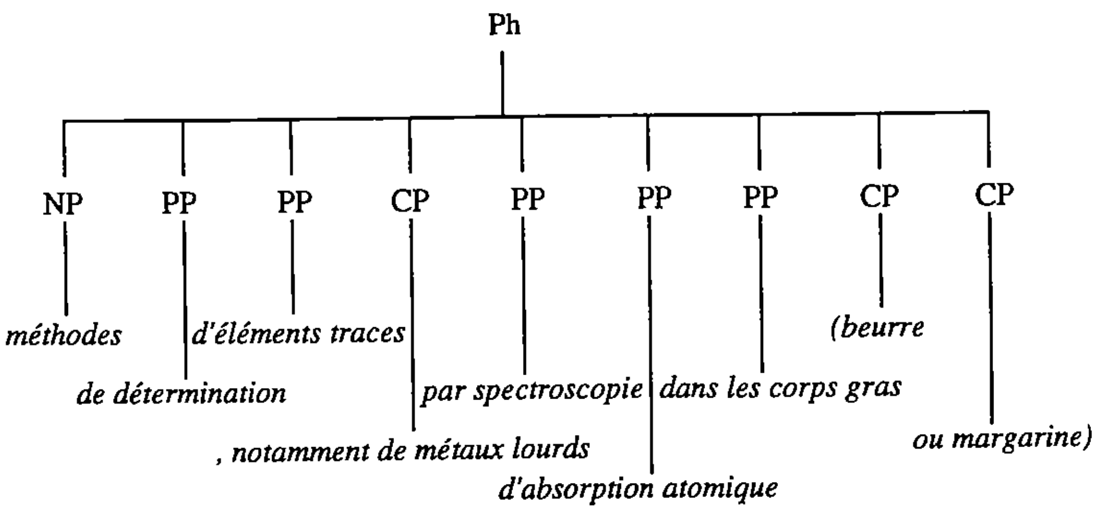

Fig. 5 : Immediate constituent «rake»

In one case however this parsing caters for a multilevel description : a relative clause is dealt with as if a phrase component, accordingly its internal representation is ascribed a specific level (a «rake» in the «rake»).

This phrase grammar sets forth a set of elementary strings (about ten word categories) and a set of ten concatenation rules (states of validation are grammatical and semantic features in lexicon) for combining phrase strings. The simplest phrase consists of a noun phrase ; more complicated phrases are built up from this center string by adjunction —PP— or conjunction —coordinate phrase CP, including apposition—.

It is expressed in a declarative top-down formalism, using a head driven and predictive analytical strategy : at each step, according to the nature of the currently processed word and the current hypothesis, a series of hypotheses is selected from the grammar and each of them is endeavored on the rest of the sentence. If no hypothesis can be found for the encountered word, the current hypothesis is scrapped. In fact, due to the restricted goal of this parsing phase (immediate constituent groups, most of them controlled by specific introducers), very few hypotheses are propounded. This parsing has proved to be very efficient on our corpus.

4.4. Syntax Component

4.4.1. Working Principles

At this stage the analyser leaves the sequential order of a sentence to attack its abstract structure. Two classes of rules are thus managed, semantic ones and structural ones.

Two semantic rules label immediate constituents (phrase rake leaves) with five semantic classes. The latter are settled on morphology and syntax, so that each immediate constituent (save CPs, ulteriorly logically evaluated) is ascribed a topical role :

• lexical family of nouns for NPs —META noun *études* assign META class to its NP *études générales* in *(3)*, *méthodes* and *détermination* convey their VIDE quality to the segment *méthodes de détermination* in *(1)*—.

• lexical family preposition for PPs —beside OBJECT prepositions as quoted in 3.2., FIELD ones and AGENT ones like respectively *dans* and *par*—.

Twenty structural rules (context-free broad coverage grammar, declarative mode, predictive parsing strategy, format «action if conditions») map out this semantic labelling with syntactic restrictions to connect some phrases with others, then to model the phrase rake into a hierarchical tree. As an example, figure 6 shows the representation of (1) after these actions :

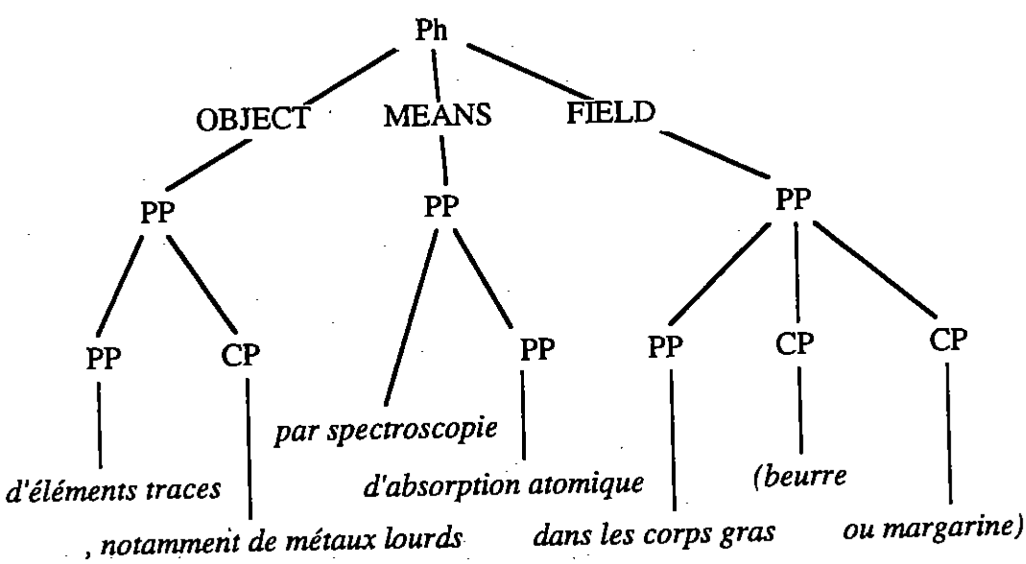

Fig. 6 : Syntactic tree format

4.4.2. Dependency Net

Three specific subsets of instructions contribute to set this syntactic dependency net up :

• the one which decides on governors for PPs. Since no world knowledge is available, it works syntax —the previously quoted TANKA device chooses for the same grounds « (...) syntax-based heuristics (...).» in the early steps of its process (Copeck, 1992, p.1009)—.Semantic labels in ANAGRAM help, equally syntactic constructions values contained in the dictionary. On the above diagram (figure 6) *par spectroscopie* occupies the upper level because *par* brings a MEANS on and because MEANS is defined as governed by a nominalized main verb (*déterminer* from *détermination*)

• the one which handles CPs. Theoretical studies (see for instance Antoine, 1962 ; Grünig, 1977 ; Ruppli, 1988) converge to emphasize the bipartite nature of the coordinative mechanism, which holds :

- a linking function. It undergoes precise spanning and attachment rules. Particularly symetric internal structure for every conjoint —as (Vergne, 1990) exploits with «isomorphism» calculus, and as (Copeck, 1992) does by checking grammatical categories in conjoined clauses— and text organization boundaries, either lexical units or punctuation and lay-out —Adam and Revaz, 1989 ; Fayol, 1989—.

- a semantic function. It commands two types of relations between conjoints, anaphoric one and detemining one. Anaphor system establishes pronominalization or ellipsis in one conjoint and corresponds to an identifying apposition (like in (4)) ; determining system specifies that joint with subordinate values (qualification, predication, emphasis, and so on).

The module manages CPs with the heuristic procedure of a meta-grammar that acts on already built constituent phrases (coordination automatic processing specifications detailed in Dami and Lallich, 1991 and in Matsumoto, 1991). Parsing is thus broken down into three operations : demarkation upon grammatical indicators, reconstruction upon isomorphism or isofunctionality patterns, lastly fastening a stated CP on the first NP on its left upon deep matching (consistent forms and prepositions). The last criterium explains why in (5) anaphoric plural *leur* prevents the CP *et leur influence sur la croissance* from rejoining singular left PP *de l'algue Gigartina*. Interpretation of the determining value has been executed by previous semantic labelling, to be used by the logic component (see underneath)

• the one wich cuts out «poor information content» items. The NP *méthodes* and the PP *de détermination* branches thereby don't appear on the diagram figure 5, as well as META items : the reason for these suppressions will be clear in the next section.

5. Logic Component

As we noted earlier (2.2.), the ultimate result of a free formulation processing is a semantico-logical description of the whole request. Such a boolean-semantic expression has to comply with information system language while it must direct further search sessions.

5.1. Boolean Transfer

The first requirement is answered with a transfer process : syntactic relations are converted into a boolean equation using logical connectives AND, OR, NOT, and a few others manipulated by information databases —like the previously pointed out EXP, which regroups several words into one uniterm— ; the syntactic tree format is consequently reorganized into a logic tree format where nodes are assigned a logical value, usually beginning with an AND root (as exemplified in figure 2).

5.2. Logical Characterization

The second requisite raises the question of the gap existing between logical connectives (they handle words) and linguistic connectives (they handle concepts).

The linguistic parser clears it up in two ways.

On the one hand, it labels syntactic dependencies with relevance weights. This ranking of syntactic arks allows the strategy module to grant priority to some branches upon others (as declared in 1.2.) : for instance if not enough answers are obtained with the whole boolean formula, the strategy module will favour OBJECT before FIELD and FIELD before MEANS. Using these clues, less constrained boolean formulas will be tried to retrieve partially relevant documents from databases. It is the same idea that leads to suppress some items (VIDE and META ones — cf 4. 4.) from the initial syntactic tree.

On the other hand, it refines the signification of boolean connectives with a logical characterization, through the use of five logical values peculiar to language links. For instance, the semantic value of *notamment* in (1) (PREFERABLY — see 4.2. and 4.4.) will be used to label the second branch of the corresponding node with a dedicated marker (PREFERABLY, as indicated in figure 2). If too many answers are obtained from databases, the strategy unit will work this information out with a view to classify answers : the first responses referred to will be the ones satisfying the second part of the OR boolean expression. As a matter of fact, even a standard OR may induce a sorting process when it comes from a CO-PRESENCE label like on the appositive string *(osmotic stress)* in (4) : the answers satisfying an AND equation are given to the user in the first place, instead of the OR genuine one.

Such a combination of weighting and logical marking supplies the user with a reasonable number (neither too few nor too many, two stumbling blocks unfolded in 1.1.) of best answers for his request.

Conclusion

Our results verify how much appropriate it is to associate theory and pragmatism in the field of smart machine requests processing (some researchers end in the same belief with supporting operational realizations, like David Evans from Computational Linguistics Search Committee at Carnegie Mellon University, or Smeaton and Sheridan from Dublin City University, 1991).

The interface has been undergoing iterative processes of evaluation and improvements for eighteen months and now runs as a fully operational product : its customer, the Languedoc-Roussillon Region Board with University Calculation Center —CNUSC— reports a 95 per cent average correct rate of information requests parsing.

Our analyser takes advantage of the minimalist linguistic option for which natural language processing doesn't need linguistic knowledge in bulk, but a few conclusive linguistic information : with an incomplete dictionary, with spare lexical and semantic attributes, with no world knowledge at all, a free expression in a specified field form is structurally, semantically and logically understood with the aim of piloting the information retrieval searching.

The parser actually exploits syntactic and semantic values of grammatical units (precisely the usually named «empty words» in information systems literature), as well from a theoretical standpoint as from an applicative one. This approach furnishes clues enough to guarantee most of the time robust linguistic processing to lead smart search strategy.

References

ABBOU A. (1989) : *Répertoire des produits et services de traitement automatique de la langue française*, DAICADIF, Paris

ADAM J.-M., REVAZ F. (1989) : «Aspects de la structuration du texte descriptif : les marqueurs d'énumération et de reformulation» in *Langue française n° 81*, Larousse, Paris

ANTOINE G. (1962) : *La coordination en français*, 2 tomes, seconde édition, d'Artrey, Paris

BENHARRAT A. (1989) : «Les stratégies de l'offre dans l'industrie des banques de données» in *Documentaliste 26(2)*, Paris

BLAIR D. C. (1990) : *Language and Representation in Information Retrieval*, Elsevier, Amsterdam

BOURIGAULT D. (1992) : «Surface grammatical analysis for the extraction of terminological noun phrases» in *COLING-92 Conference Proceedings* Vol. 3, Nantes

CARADEC R., SAADA G. (1982) : «Définition de la classe syntaxique d'une forme lexicale à partir de sa terminaison graphique» in *Linguisticae Investigationes IV:2*, John Benjamins B. V., Amsterdam

CHAUMIER J. (1986) : *Systèmes d'information : marchés et technologies*, Entreprise Moderne d'Edition, Paris

CHURCH K. W. (1988) : «A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text» in *Proceedings of the Second Conference on Applied Natural Language Processing*

COPECK T., DELISLE S., SZPAKOWICZ S. (1992) : «Parsing and Case Analysis in TANKA» in *COLING-92 Conference Proceedings* Vol. 3, Nantes

DACHELET R. (1990) : «Etat de l'art de la recherche en informatique documentaire : la représentation des documents et l'accès à l'information» in *Le Document Electronique*, cours INRIA, Paris

DAMI S., LALLICH-BOIDIN G. (1991) : «An expert-system for French analysis within a multi-mode dialogue to be connected» in *RIAO 91 Conference Proceedings* Vol. 1, Barcelone

EC2 (1991) : *Génie linguistique 91 Conference Proceedings* Vol. 1 & Vol. 2, Versailles

FAYOL M. (1989) : «Une approche psycholinguistique de la ponctuation : étude en production et en compréhension» in *Langue française n° 81*, Larousse, Paris

GROSS G. (1988) : «Réflexions sur la notion de locution conjonctive» in *Langue française n° 77*, Larousse, Paris

GRÜNIG B. N. (1977) : «Bilan sur le statut de la coordination» in *DRLAV n° 15*, Paris

LE CROSNIER H. (1990) : *Systèmes d'accès à des ressources documentaires - Vers des anté-serveurs intelligents*, thèse de doctorat, Université de Marseille

LJUNGBERG J. (1991) : «Paraphrasing SQL to Natural Language» in *RIAO 91 Conference Proceedings* Vol. 2, Barcelone

MATSUMOTO Y. (1991) : «Handling Coordination in Logic-based Concurrent Parser» in *Natural Language Understanding and Logic Programming III Proceedings*, Elsevier, Amsterdam

PONAMALE M. (1987) : *Compréhension automatique de mots inconnus à partir de leur morphologie*, Mémoire de l'IIE, CNAM, Paris

RUPPLI M. (1988) : *La coordination en français moderne. Théorie à l'épreuve d'un corpus*, thèse de doctorat, Université Paris III

SALTON G. (1988) : *Automatic Text Processing : the transformation, analysis, and retrieval of information by computer*, Addison-Wesley

SEGOND F., JENSEN K. (1992) : «An Integrated Syntactic and Semantic System for Natural Language Understanding» in *COLING-92 Conference Proceedings* Vol. 3, Nantes

SMEATON A. F., SHERIDAN P. (1991) : «Using morpho-syntactic language analysis in phrase matching» in *RIAO 91 Conference Proceedings* Vol. 1, Barcelone

STRZALKOWSKI T. (1992) : «TTP : a Fast and Robust Parser for Natural Language» in *COLING-92 Conference Proceedings* Vol. 1, Nantes

VERGNE J. (1990) : «A parser without a dictionary as a tool for research into French syntax» in *COLING 90 Conference Proceedings*, Helsinki

VICTORRI B., THOMAZO L., BOYREAU G., MADELAINE J., COULOMB J. F., HENRIOT D., LE CROSNIER H., GIROLLET D. (1992) : «L'anté-serveur : une interface intelligente avec l'univers documentaire» in *Informatique 92 Conference Proceedings*, Paris