Coping with vague and fuzzy words : a Multi-Expert Natural Language System which overcomes ambiguities

Violaine Prince and **Gérard Sabah**

The Language and Cognition Group,
LIMSI – CNRS, B.P. 133, 91403 ORSAY Cedex, FRANCE, Tel: 33 (1) 69 85 80 03,
E-mail (FNET) : Prince@limsi.fr / Sabah@limsi.fr.

TOPIC AREA : Natural Language Processing and Applications

Key -words : Semantics; Ambiguity Resolution; Linguistic Knowledge Representation

Abstract : the system we present in this contribution is a sub-part of an extensive natural language processing architecture. It is devoted to a particular phenomenon in natural language : the polysemy of words, which produces effects such as vagueness and fuzziness and impedes the normal automatized processing of sentences. We try to show in this paper how our architecture, currently running on a network of SUN stations, relates to preceding works well-known in NLP, and what particular improvement this system for polysemy resolution provides. This system benefits from the dedicated properties of the theoretical model for lexical description. The latter derives its power from the coupling of non monotonic logic (which proved to be very useful for semantic interpretation) and a multi-valued approach for vagueness handling.

Introduction

The aim of this contribution is to present a system which helps the semantic automatic interpretation of a text in natural language, when the problem encountered is generated by the polysemy of one or many words in a given sentence. Polysemy is the phenomenon of recognizing multiple meanings associated to a word, and not being able to firmly decide which meanings have to be rejected. Common effects of polysemy are vagueness (too wide a scope) and fuzziness (unability to strongly prefer a solution). Therefore, semantic interpretation is impeded with a decision problem. Thus, the main reasons for designing an automatized tool for polysemy resolution are to detect or to confirm the validity domain of some meanings attached to a polysemous element, so that a conceptual decision could be more efficiently taken. Secondly (that's not the least) it allows for an enhancement of a natural language processing system robustness.

The requirements of our research have made us design our tool as a brick in a more complex software architecture for natural language processing: CARAMEL [Sabah, 1990]. This choice is motivated by the following reasons:

1 - polysemy effect is only detected when the classical ambiguity resolution - by matching its items with either frames, conceptual graphs or semantic networks - has failed or has proven unsatisfactory. Therefore, our system must be coupled with a traditional parsing module.

2 - Conversely, 'polysemous' elements, when analyzed with our tool, offer the NLU[1] system some pragmatic knowledge, particularly about the local context of the sentence segment or paragraph (this could be a metaphoric local context within a general technical context).This knowledge might prove useful when processing the remaining part of the sentence (or paragraph).

3 - Our system does not impose a solution, but provides a diagnosis about the most probable semantic field in which the NLU system could pick up a concept.

In section 1, we make a quick survey of the problems relative to ambiguity resolution, and the way the CARAMEL architecture has integrated techniques to solve major referential and conceptual ambiguities. In section 2, we detail the architecture of our system as a part of

---

[1]. Natural Language Understanding.

CARAMEL, and the way it tries to link lexical properties to conceptual bases. We last conclude about the present state of our work and give some prospective orientations for our system.

1. Ambiguity resolution

1.1 Different types of ambiguity and corresponding resolution techniques

1.1.1 Major lexical ambiguities encountered by NLU systems

We mainly see two major types of lexical ambiguities.

1- *Referential:* the problem is to link the lexical element processed with a correspondent already existing in the representation of the world, or to indicate the necessity of creating a new instance of an object belonging to the world. This is often the case of ellipsis, anaphora and other types of referential ambiguities.

2- *Conceptual*: a conceptual ambiguity is produced by either homography or polysemy. **Homography** describes the problem of different and independent concepts addressed by the same lexical string (word). Let us remind Cottrell's example of a clear homography [Cottrell 1985]: *the dog's ball* as opposed to *the high school annual ball.* **Polysemy** is more subtle because it addresses concepts which have elements in common (at the extreme, these concepts are uneasily describable with a single word). One may have a **functional polysemy** like string in *Violin belongs to strings* or in *these are twenty-character strings* (that is, polysemy is disambiguated by the functionality associated with the word, which is not its prime function) or **categorial polysemy** like mother in *a mother cell gives birth to two identical daughters* or in *IBM's mother house* as well as in *the Nile mother springs are numerous:* categorial polysemy effects are that of the 'undecidability' about the concept referred because of a 'non-atomic solution' on the reader's behalf; many concepts act as being valid, but not necessarily at the same level of validity. The preference of a concept is more likely to be replaced by the **emergence of a relevant 'conceptual field'** which is less fine-grained than the former.

1.1.2 Techniques associated with lexical ambiguity resolution

The techniques we present are the most popular and we do not claim to provide an extensive list. Let us say that, in our opinion, most of these techniques have in common the combination of inferences of a pragmatic origin with a more or less sophisticated representation of semantic knowledge, to the point where pragmatic knowledge could sometimes be recorded in a semantic network [Sabah, 1978].

1- *Referential lexical ambiguities*: particularly fitting for this problem, we see the research Grosz has initiated about the concepts of Topic and Focus [Grosz, 1977] which is, in our eyes, corroborated by the concept of Interest developed by [Schank, 1979], about the referential ambiguity resolution inferences. In this trend, Sidner (1979) has developed a theory which main point of interest is to offer an algorithm for anaphora resolution. In a further work [Grosz & Sidner 1986], she showed that the same algorithm could apply to conceptual ambiguity resolution. We think that this particularly applies for homography, and even for functional polysemy.

2- *Conceptual lexical ambiguities:* as [Hirst, 1987] pointed out, conceptual ambiguities have to rely heavily on context information in order to choose between different scripts, and even though "in a single script, a word, especially a polysemous one, may still be ambiguous" (op. cit. P.78). Among the numerous methods developed for lexical disambiguation, let us mention Wilks' Preference semantics [Wilks, 1975] which tolerates lexical metaphors to a certain extent, Boguraev's Semantic judgments [Boguraev, 1979] which associates structural with lexical disambiguation, Hayes' work [Hayes, 1977] about finding associations when considering different sources of knowledge mainly represented by frames and case structures. All these methods tend to supply conceptual semantics with 'pragmatic knowledge' as much about the word usage as about commonsense knowledge [Dahlgren, 1989]. In our eyes, these methods are attempts to modify highly conceptual (general) representations with the specificities of lexical usage traditions. They have proven to be efficient in the case of homography and often in the case of what we called functional polysemy, although the ratio of knowledge required grows in great proportion. Nevertheless, [Small, 1980] [Small & Rieger 1982] has shown, that if one wanted to solve totally the phenomenon of lexical ambiguity, one had to provide the system with a complete 'expert' about the word: linguistic knowledge which is associated with the polysemous lexical element is not circumscribed by general mechanisms, whatever precise they are. We think that Small has highlighted a problem proper to what we called 'categorial polysemy' but we do not go as far as he goes in the rejection of generality. Our hypothesis is that although linguistic knowledge must be given its importance, **there is a possible tradeoff**

between conceptual semantics and lexical semantics, provided that one accepts a certain 'degradation' in interpretation. The aim of our system is to show that categorial polysemy could be approached without the cost of a Word Expert, and without dropping conceptual representations which usefulness is still great. Of course, one must then accept that categorial polysemy would not be as precisely interpreted as the other types. But at least, this will spare the NLU system an interruption by failure, or a total misunderstanding.

1.2 The CARAMEL architecture solution for major types

1.2.1 The components of the CARAMEL architecture

Let us briefly remind some of the contents of the CARAMEL architecture before describing the methods it employs for ambiguity resolution. The CARAMEL system is composed of three fundamental elements:

- a **structured memory** containing permanent knowledge and working structures of the system
- a set of **processes**, dedicated to the execution of the various cognitive tasks
- a **supervisor**, whose function is to trigger, to run coherently and to synchronize the processes.

The memory contains a kind of blackboard, which is enhanced with a control mechanism driven by meta-rules. The system may be used in such different applications as: user-friendly interfaces, on-line help in text processors, summaries of texts, intelligent tutoring systems, etc.

The memory

CARAMEL makes use of three kinds of memory:

- a *short term memory*, which receives the results of the perceptual processes.
- a *working memory*, which contains all the structures (eventually provisional) built by the different processes. The text is represented at the various levels differently according to the point of view.
- a *long term memory* which contains all the knowledge of the system. This memory contains for example information about morphology, words, grammar, syntax, semantics and pragmatics, and it is a stable representation of the current state of the world. All this knowledge

is permanent, declarative and expressed in terms of conceptual graphs [Sowa 1984], as this formalism facilitates the communication between processes[2].

The supervisor

Our model takes into account the fact that the various processes cannot be pre-ordered (their order depends on the global task and the particular data to be processed) and that their management has to be based on a **planning** process. As each complex process acts as a supervisor with regards to its subprocesses, the system has a recursive structure.

The supervisor analyses the representations stored in the working memory and deduces the processes that may be triggered. It also takes into account the needs of the active processes and the global task: it has to handle interruptions coming from them. Thus, the system integrates bottom-up and top-down control.

First, a **static planner** builds the sequence of complex processes that can build the needed type of representation (*task-oriented reasoning*). The same type of reasoning produces sub-plans for the complex processes involved.

This first phase is independent of the data to be processed, and, as some aspects are not yet well defined, several processes may be optional (pronoun resolution, error correction…). In a second phase, a **dynamic planner** allows the system to adapt the static plan to the specific data present in the working memories. Whenever a problem arises, the dynamic planning process takes care of it, by determining (based on the inputs and outputs of the available processes) the kind of process capable to solve it.

The existence of these two planning phases allows that a process selected in the static planning phase can break down. When this occurs, a help request is sent to the supervisor which triggers a dynamic planning phase. When the processes selected in this second phase solve the problem, the original process is resumed[3].

---

[2]. At present the implemented data corresponds to a French lexicon of 15 000 entries (about 350 000 conjugated forms), a semantic net of a thousand concepts, a grammar (350 rules) allowing the analysis of complex sentences with prepositional and relative phrases (in French), and pragmatic knowledge about the world (at present, the system knows only a few frames in order to test the validity of the processes).

[3]. Parallelism offers another solution to adapt the static plan to the data. Its implementation, currently under development, will permit a **continuous control** on the processes (Fournier et al 1990): the supervisor will be able to examine the representations that are being constructed and, possibly interrupt a given process in order to wait for the result of another one or to give it some advice.

1.2.2 What ambiguities are solved by CARAMEL

The ‘text comprehension’ task invokes some specialized processes in CARAMEL.

The deterministic parser ANDI [Sabah & Rady 1983] analyses the input and produces a morphological and syntactical representation along with a semantic representation in the form of the associated conceptual graph.

The ‘Character handler’ [Sabah, 1978] [Sabah & Berthelin 1980] creates a referential system with the assumption that the text is a ‘story’ which puts together some characters (closely related to conceptual representations).

The ‘Topic manager’, based upon Grau's work [Grau, 1983], helps relate the semantic representation to the world knowledge and provides a pragmatic-oriented approach of the discourse.

These major processes are associated with others, maybe more restricted in their goals. Among them, let us mention the ‘pronoun solver’ which is directly linked to both of them and whose role is to resolve anaphora. These processes have an important hand in ambiguity resolution.

1- Referential ambiguities: the pronoun solver and mainly the character handler and topic manager have had until now, an efficient behavior for major cases of anaphora and simple ellipsis (intrasentencial). The parallel activation of these processes is considered for solving the cases of intersentencial ellipsis.

2- Conceptual ambiguities: the case of homography is correctly dealt with by the topic manager. Functional polysemy is partly solved: the topic manager is able to provide the proper frame, the problem is that conceptual semantics (mainly conceptual graphs) are not yet sufficiently representative. Nevertheless, some improvements have been suggested by [Sabah & Vilnat 1991] who propose a finer-grained case decomposition which will reduce discrepancies between the reference conceptual graphs and the restriction resulting from the parsing process. Presently, categorial polysemy is not dealt with as such. That is, the lexical side effects of some words are not taken into account, as long as they do not introduce too important a misunderstanding.

2. The GLACE complex process for semantic decision

The aim of the GLACE[4] system is to complete effectively the panel of CARAMEL processes which are invoked for semantic interpretation by offering a tool which avoids a disturbance caused by categorial polysemy. The particularity of GLACE is that it tries to bridge conceptual representations in the CARAMEL long term memory (knowledge bases) with a local lexicon, modelled according to the specificities of linguistic knowledge and polysemy: this local model has to be 'compatible' with a conceptual graph representation, and the system has to be as tolerant as possible to contextual disturbances. We present the system dynamics in § 2.1 and the local lexicon modelling in § 2.2.

2.1 An instanciation of the CARAMEL principles

2.1.1 The GLACE system in its CARAMEL environment

The figure 1 shows the direct environment of the GLACE system seen as a process within the architecture. For semantic interpretation, the supervisor may have the following possibilities. After triggering the ANDI parser, it could initiate a GLACE-like interpretation of the lexical elements in order to delimit the conceptual fields in which other processes could pick up the proper elements for their resolution. The other possibility is to keep GLACE at the end of the chain and not trigger it unless interpretation has failed. Nevertheless, the CARAMEL supervisor also acts dynamically: at any level of the sequence it may drop or invoke an expert process, according to the contents of the working memory. Furthermore, using its parallel capabilities, it is likely to initiate, for instance, a GLACE-like interpretation in parallel with the *character* processing. The results of this interpretation are given back as an advice to the *character* handler, and as semantic features for the parser, stored for further processing.

2.1.2 The GLACE architecture

The GLACE system is in itself an instanciation of CARAMEL which is designed as a recursive architecture. This means that GLACE is modelled according to the same principles. It is composed of a sub-supervisor which manages a set of two main processes and a local lexical

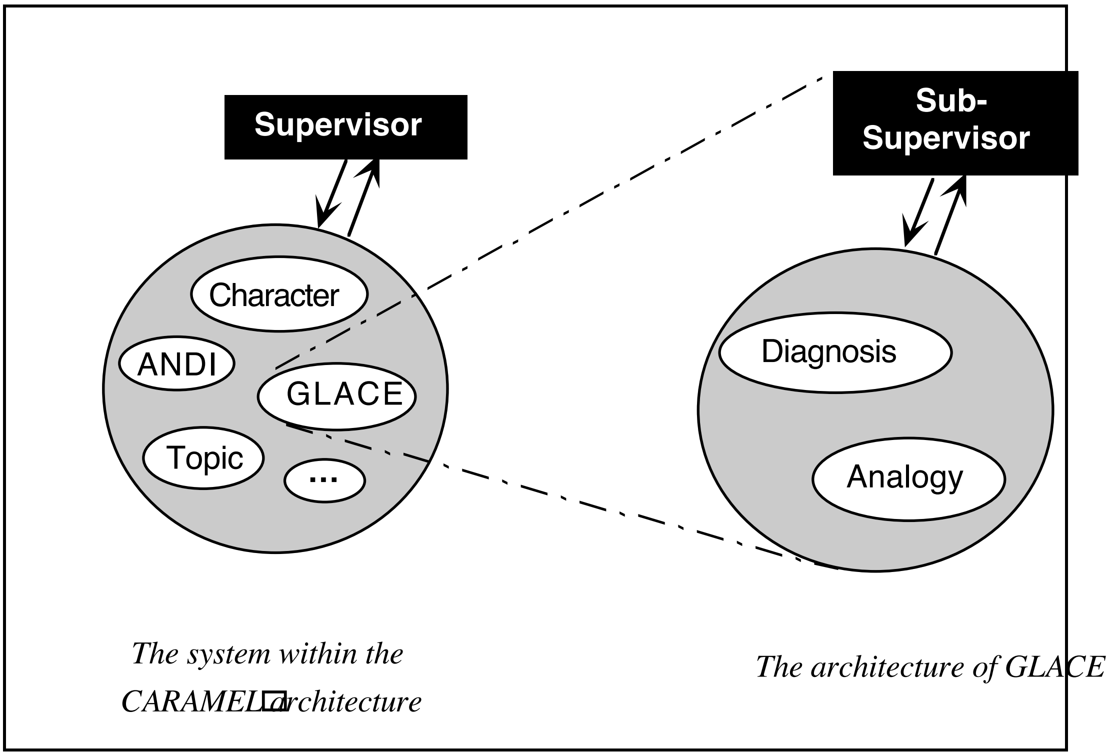
---

[4]. Acronym of "Graphes Lexicaux d'Aide à la Compréhension des Enoncés" : Lexical Graphs that Help Understanding Texts.

base. It is thus named a **complex process**. The two processes are termed as **simple** (elementary) and are the following:

1- the **diagnosis module** is composed of rules of interpretation : in fact, it is a very simple module because it relies on the power of non-monotonic logic circumpscription and default principles that we have implemented as an inference tool associated with our particular lexicon. It is triggered by the sub-supervisor when GLACE is invoked for the task of 'semantic determination'. Its input is the result of the semantic and syntactic analysis provided by the parser, the character handler, the topic manager (what has been transmitted to the GLACE local working memory). It attempts to locate the 'polysemous' word by matching the words in nominal, adjective or verbal position with the words of its knowledge base. If this matching succeeds, it will instanciate the *local pragmatic rules* attached to this word, and collect the result of their application. The result will be its output to sub-supervisor which will transmit to the CARAMEL supervisor.

2- The **analogy module** whose aim is to offer an interpretation if the diagnosis module fails to invoke its pragmatic rules. It relies on structural analogies between lexicon elements to replace a lacking rule with the rule of another element provided that these elements are "alike" enough. *Example*: let us suppose that the polysemous word 'room' has no reasonable interpretation in the sentence: "*he left no room for discussion*" (no associated pragmatic rule premise being instanciated by such a context). Let us also suppose that we know how to interpret the word 'place' in the same context: we have at least a pragmatic rule associated to 'place' which could be triggered by the morpho-syntactic and partially semantic representation resulting from this sentence pre-parsing. Last, suppose that 'room' and 'place' are structurally and pragmatically close (we determined tolerance relations to build this notion of closeness). The analogy module proposes a diagnosis of 'room' by instanciating the 'instanciable' pragmatic rule of 'place' and making it value as much as possible the interpretation of 'room'.We have built up an algorithm which succeeds in interpreting by analogy a case of incomplete information. [Prince & Bally-Ispas 1991]



The system within the CARAMEL architecture

The architecture of GLACE

Figure 1. *CARAMEL and GLACE.*

2.2 Using a particular model for the local lexicon

GLACE benefits from its CARAMEL architecture by having the possibility to rely on a local knowledge base corresponding to a structured long term memory.

As we have pointed out throughout this article, the organization of lexical knowledge associated with categorial polysemy cannot be supplied by conceptual oriented representations.

2.2.1 Main goals of the model, its implementation features and its relationship with other theoretical approaches

We have designed a particular lexical model (detailed in [Prince, 1990] [Prince, 1991]) whose ambitions were the following:

(Goal 1)- to offer a stable 'core of meaning' that we named a *potential*, organized as a graph mixing both conceptual categories and a feature system, with some pre-arranged semantic constraints between features, so that correspondence with conceptual semantics would be rendered easier without impinging upon the calculus of sense. This graph is very simply implemented by means of a structured LISP list.

(Goal 2)- To provide a *dynamic calculus of the sense of a word* in its local context, whose result is a valuated configuration of the word potential which indicates both the validity domain and the saliency of some conceptual categories if so it appears; this part allows some idea to the notion of continuity in semantics proper to categorial polysemy processing [Konfé, 1991], it is compatible with the ‘multiple-readings’ hypothesis [Fodor & Garrett, 1967] and to some extent to the slightly different ‘all-readings’ hypothesis [Swinney, 1976]. This dynamic calculus of sense is triggered in the GLACE system, by matching context information with the premises of some specific rules called *pragmatic rules*. It is continued by applying default and circumpscription rules written within the diagnosis module of GLACE.The process ends when all the features of the graph have been provided with values.

(Goal 3)- To offer pragmatic rules, expressing the *linguistic expertise* about the word usage, associated to the word itself: they are implemented within the same list as the potential. An example of a data structure of our lexicon is given in figure 2. It shows the variety of information recorded within the lexicon as well as its intrinsic simplicity. Pragmatic rules initiate the valuation process corresponding to the calculus of the contextual sense of the word. They are our restricted interpretation of the necessity of proper linguistic knowledge termed by Small in his theory about Word Experts.

```
   (; father structure
    ((transcendance () ()) (hierarchy () ((antecedence ()) (generation ()) (likeness ())))
(affectivity () ((humanity ()) (benevolence ()) (punishment ()) (authority ()))))
   ;integrity rules
   ((imp(generation antecedence)) (opp(benevolence punishment)))
   ;pragmatic rules
   (((position-adjective not-before) ((generation salient) (affectivity ignored)))
    ((context-family) ((humanity salient))))
```

Figure 2. *Example of the data structure of the word "father".*

2.2.2 The formalized model implemented in the local lexicon

Let us name n the representation of an entry w. n is totally characterized by the pair {A,P} and the predicate B_W which we define as following.

<u>Descriptive category/feature representation</u>

We call A the set defined as $\mathbf{A = A_1 \square A_2}$ where:

$A_1 = \{K_1, \ldots, K_n\}$ is the set of conceptual categories associated to the entry w.

$A_2 = \{F_1, \ldots, F_t\}$ is the set of features describing the conceptual categories recorded in A_1.

To create the proper correspondences between features and categories we define a predicate B_W which associates a particular feature to a category in the context of the entry w. $B_w(K_i,F_j)$ = True if K_i □ A_1, F_j □ A_2 and F_j is a feature of K_i relatively to w.

Rules representation for sense calculus and semantic constraints representation

P is the *set of rules* associated with A. **P= P_1 □ P_2 □ P_I** where P_1 is the set of pragmatic rules associated with the entry w, P_2 is the set of semantic constraints associated with w and P_I the inheritance rules set common to all entries.

1- Pragmatic rules:

$P_1 = \{[w_j \varnothing p(a_i,v_i)]\ w_j$ □ $□_w, a_i$ □ A, v_i □ $V\}$

The type of the rules is expressed by the first order predicate calculus implication : $[w_j \varnothing p(a_i,v_i)]$

This expression involves *instanciable conditions* e_j □ $□_w$ and *valuation actions* $p(a_i, v_i)]$, a_i □ A, v_i □ V. These actions are modelled by the predicate p which is defined as following. Let us note V: A ∅ V, V(a) = v, a valuation function, which assigns a value to each element of A (feature or category).

p: AxV ∅ {T,F}

(a,v) ∅ p(a,v) = T If V(a) = v, p(a,v) = F if V(a) ≠ v.

The set V of the possible values associated to the elements of A is defined as a four element set, corresponding to a 'four-values' logic which we consider proper for the calculus of sense determined in our goal number two:

V = {salient, inhibited, accompanying, ignored}

These values reflect the *activation state* of the elements of A for an occurrence of the lexical entry w in a specified context. '*Salient*' means that the element is directly or strongly indicated by context information. '*Inhibited*' means that the feature is contradicted by the context by means of a negation (different from the absence of activation). '*Accompanying*' means that the element is neither directly activated nor contradicted by the context: it can possibly represent the 'implicit' information for the occurrence of the lexical entry and it belongs to the validity domain *by default*. '*Ignored*' means that the element is not relevant in the valuation context and has to be discarded.

2- Semantic Constraints representation

The set P_2 corresponds to pre-formed semantic constraints between features of a same category. $P_2 = \{[p(a_i, v_i) \square B_W(c_i, a_i) \square B_W(c_i, b_i)], c_i \square \square_1, a_i, b_i \square \square_2, v_i, w_i \square V\}$

The allowed semantic constraints between features are: *implication* and *opposition*. (see the example in figure 2) The value by default, not expressed by means of rules in this first set, is *co-existence*. The only possible relation between categories is the value by default.

<u>Definition of implication:</u> (the value of an element is transmitted to another element)

$a_i \varnothing a_k \square \quad p(a_i, v_i) \varnothing p(a_k, v_i)$

<u>Definition of opposition:</u> (a valuated element transmits the opposite of its value to the other element)

$a_i / a_k \square \quad p(a_i, v_i) \varnothing p(a_k, opp(v_i))$

3- Inheritance rules

They address the propagation of valuation among features and categories which have not been yet valuated.They are embedded in the diagnosis module of GLACE. Let us notice that we use here non monotonic default principles for an unusual application : generally, default is used within a reasoning frame. Here we use it as a *value-propagation mean*. The power of default rules is such that the knowledge recorded in GLACE modules shrinks to the minimum. Here these rules are given in the form of production rules, close to their implemented form (in LISP).

Rule 1.$\square x \square A2, \square y \square A_1, \square v_1 \square \{salient, inhibited\}$

$[p(x, v_1) \square B_W(y,x) \square \neg(p(y,v)) \varnothing p(y, salient)]$ (*high value inheritance*)

Rule 2.$\square x \square A_1, \square y \square A_2 [p(x,ignored) \square B_W(x,y) \square p(y,v) \varnothing p(y,ignored)]$ (*circumscription of irrelevance*)

Rule 3. $\square x \square A_1, \square y \square A_1 [p(x, v) \square \neg(=(v,ignored)) \square \neg(p(y,v')) \varnothing p(y, accomp.)]$ (*total default rule*)

Rule 4. $\square x \square A, \square v \square V [p(x,v) \square p(x,accomp.) \varnothing p(x,v)]$ (*circumscription principle*)

2.3 An example

Let us give the example of the verb 'create', which is polysemous. A data structure is created in our lexicon. We give here its semantics by means of the formalized model:

A_1 = {produce, transmit, life} the set of conceptual views ;

A_2 = { physical, abstract, mean, origin, breath, generation, antecedence, alikeness} the set of features ;

B_{create} is defined as : B_{create}(Produce, x) & x □ {physical, abstract};

B_{create}(Transmit, x) & x □ {mean, origin} ;

B_{create}(Life, x) & x □ {breath, generation, alikeness, antecedence}

We define the sets of rules P_1 and P_2 specific to the entry "create".

P_2 = { generation --> alikeness, generation --> antecedence, physical/abstract}

P_1 contains around ten rules. But among them we have the rule :

(agent physical) & (context technical) ---> (physical, salient) & (breath, ignored) & (generation, accomp.) (Rule 08)

This rule has been triggered by the sentence :

The star supernova could create a black dwarf.

"Supernova" has been recognized as "technical" in the semantic network. "Star", in conjonction with it, is the "physical celestial body".The inheritance rules of the model give the following valuation for the configuration of *create* :

p(Produce, salient) & p(physical, salient) & p(abstract, inhibited) & p(Transmit, accomp.) & p (mean, accomp.)
& p(origin, accomp.) & p(Life, accomp.) & p(breath, ignored) & p(generation, accomp.) & p(alikeness,
accomp.) & p(antecedence accomp.)

Figure 3. *Result of the diagnosis module.*

In terms of information to conceptual semantics, GLACE transforms the result of its diagnosis module (as seen in figure 3) by matching conceptual views with concepts in the long term memory of CARAMEL. As one can see in figure 4, it proposes a prefered concept (the salient one) with a *restricted case structure*. But it also proposes **default concepts** (named authorized predicates) with restricted case structures. This means that if the prefered concept were not a best choice in a further processing of the text (i.e. in the next or close sentences) then the system has the opportunity to invoke default choices.

the salient conceptual field is : PRODUCE a PHYSICAL NOT-ANIMATE
authorized features : GENERATION, ALIKENESS, ANTECEDENCE,
authorized predicate : TRANSMISSION
authorized cases : ORIGIN, MEAN.

Figure 4. *The result of GLACE to CARAMEL supervisor*

In terms of our example, the result in figure 4 shows that "black dwarf" can only be interpreted as a physical inanimate object, possibly generated by the agent which is prior to it, and possibly resembling the agent. "Black dwarf" can also be the theme of a predicate associated to the idea of "transmission".

Conclusion

In this article, we have shown that polysemy should be considered differently from homonymy and that few works really tackle that problem, at least from a dedicated point of view. We then remind of the CARAMEL architecture (an application of Distributed Artificial Intelligence to NLU) and how it processes ambiguities. In order to improve its handling of polysemy, we propose GLACE (a new *complex module* in the system).

After having described the sub-modules GLACE controls, we showed its ability to dynamically calculate the sense of a word that is the most contextually appropriate. We also gave the formal representation of the lexicon and definition of the rules that allow such behaviour. We would like to insist upon the fact that, however "complex" the formalized device seems to be , its implementation is oppositely simple.Theoretical complexity was a mean to provide interesting implemention-simplifying issues.

The good results of the present program lead us to envisage its extension over the whole lexicon of CARAMEL and to consider it as the main basis of semantic interpretation. Such a functioning seems to have good psychological justification and will allow to handle and to explain easily such phenomena as metaphorical expressions and semantic flexibility.

REFERENCES

BOGURAEV B.K. 1979 "Automatic Resolution of Linguistic Ambiguities". PhD Dissertation (TR 11), Computer Laboratory, University of Cambridge.

DAHLGREN K., MCDOWELL J., STABLER E.P. 1989 “Knowledge Representation for Commonsense Reasoning with Text “. Computational Linguistics, vol.15, n°3. Pp. 149-170.

FODOR J. & GARRETT M. 1967 " Some syntactic determinants of sentencial complexity "in Perception and Psychophysics, n°2. Pp.289-296.

FOURNIER J.P. , SABAH G. & SAUVAGE C. 1990 "A parallel architecture for Natural Language Understanding Systems". Proceedings of PRICAI-90, Nagoya. Japan.

GRAU B. 1983 " Analyse et Représentation d'un texte d'après le thème du discours". Thèse de doctorat de 3ème cycle. Université Paris VI.

GROSZ B. 1977 The representation and use of focus in a system for understanding dialogs". Proceedings of the Fifth IJCAI. Cambridge,Massachussetts.

GROSZ B.J. & SIDNER C. 1986 "Attention, Intention and the Structure of Discourse". Journal of Computational Linguistics, 12 (3). Pp. 175-204.

HAYES P. 1977 "On semantic nets, frames and associations". Proceedings of the Fifth IJCAI, Cambridge, Massachussetts. Pp. 99-107.

HIRST, G. 1987 Semantic interpretation and the resolution of ambiguity. Cambridge University Press. Cambridge, Massachussets.

KONFE A. 1991 "Contribution au traitement automatique de la polysémie". Thèse de doctorat de l'Université de Caen.

PRINCE V. 1990 "EDGAR: a model and a method for Ambiguity Representation in NLU systems" Proceedings of AINN90. Zurich, Switzerland.Pp.209-213.

PRINCE V. 1991 "GLACE un système d'aide à la compréhension des éléments lexicaux inducteurs d'ambiguités". (*GLACE : a system that helps understanding lexical elements which induce ambiguities*) Proceedings of RFIA-91, Lyon. France.Pp 591-601.

PRINCE V. & BALLY-ISPAS R. 1991 "Un algorithme pour le transfert de règles pragmatiques par analogie d'inférence dans le processus complexe GLACE".*(an algorithm for pragmatic rules transfer by analogical inference in the conplex process GLACE)*.LIMSI Internal report n. 91-17. France.

SABAH G. 1978 "Contribution à la compréhension effective d'un récit". Thèse de doctorat es Sciences. Université Paris VI.

SABAH G. 1990 “ CARAMEL: a flexible model for interaction between cognitive processes underlying natural language understanding”. Proceedings of *COLING-90*. Helsinki.

SABAH G. & VILNAT A. 1991 "Flexible Case Structure Implemented into a Deterministic Parser". International Workshop on Conceptual Graphs. Binghampton.New York.

SCHANK R. 1979 "Interestingness: controlling inferences", in AI n°12, pp.273-298.

SIDNER C. 1979 "Towards a computational theory of definite anaphora comprehension in English discourse". PhD Thesis. MIT, Artificial Intelligence Lab.

SMALL S.L.& RIEGER C.J. 1982 "Parsing and comprehending with word experts (a theory and its realization)". In Lehnert and Ringle. Pp. 89-147.

SOWA J. 1984 Conceptual Structures: processing in mind and machine. Addison-Wesley. Reading,Massachussetts.

SWINNEY D.A & HAKES D.T. 1976 Effects of prior context upon lexical access during sentence comprehension. In Journal of Verbal Learning and Verbal Behavior n'15(6).Pp.681-689.

WILKS Y. A. 1975 "A preferential, pattern-seeking semantics for natural language inference". In Artificial Intelligence n° 6, pp 53-74.