MODIFIED CASEFRAME PARSING FOR SPEECH UNDERSTANDING SYSTEMS*

Maaaiao o**, Claudio Rullent

CSELT - Centro Studi e Laboraton Telecomunicazioni
Via G.Reiaa Romoli, - 10148 Torino - Itsly

ABSTRACT

We suggest for speech understanding systems a esseframe paraing strategy which devistes from 'pure' casefrsme parsing in st lesst two respects: paraing is not exclusively bssed on top-down instantiation of casefrases, and casefrases are Merged before use with syntactic knowledge. Our strategy has been developed to be executed *m&* an inferential process in which a set of Knowledge Sources cooperate through the blackboard. The Knowledge Sources are automatically defined by Merging syntactic and semantic knowledge expressed declsrstivslyi this integration of ayntax rules and caseframes Makes it possible to exploit siMultsneously both syntactic and semantic constrsints.

I. INTRODUCTION

Psrsers for spoken nstursl language and parsers for typed natural language must Meet different requirements. Psrsers for speech work on a set of <u>lexical hypotheses</u> rather than on sequsness of words. An (unpleasant) consequence is that it is usually possible to find sequences of hypotheses not corresponding to the sequence of words actually uttered but which neverthelees constitute complete and correct interpretations. A psrser for speech Must be able to work when some words like srticles and prepositions have not been recognized, and to decide whether two slightly overlapping lexical hypothesee srs competing or not. For all these reasons, paraing a spoken utterance requires inferences as complex as those of empirical, inductive enterprises as expert systems and originates similar problems.

Casefrsme parsing has recently been proposed ([Hayes et si. 1986], [Bristzmsnn, Ehrlich 1986]) as an alternative to the methods developed during the ARPA-SUR project like semantic grammars ([Hayes-Roth 1980]) or networks ([Lowsrre 1976]). Having choaan the blackboard architecture, auitable to cope with the probleme of speech paraing aketched above, we have introduced two modifications to 'pure' caseframe paraing to uae it effectively into such a framework: I. alternating between top-down instantiation of caseframes and bottom-up prediction of casefrsmea requiring a caaefiller already recognized; and Ii. using caaeframes only after they have been merged with syntactic knowledge.

II. A REVIEW OF CURRENT RESEARCH IN CASEFRAME PARSING

The basic notion ofcasseframe is thst of s hesd concept modified by s set of related concepts ([Fillmore 1968], [Tesnisrs 1939]). Each modifier plays s rols (CASE) with respect to the hesd concept. Casefrsme paraing has been proposed long ago for typed nstursl language psrsers ([Hsyes,Csrbonell 1981]) and its applicability to parsers for speech has recently become an object of study ([Hayes et al. 1986], [Brietzmsnn,Ehrlich 1986]). This strategy is promising because the interpretation is anchored to most significant parts of the input, and semantic expectations generated from these more meaningful psrts can be used both to diacriminate between different candidate fillers and to hypothesize the meaning of troublesome fragments. But current caaeframe parsers for speech are still close to parsers developed for typed text, which causes two main problems.

The first problem is thst the parsing strategy they use is top-down: in s first stage all potential caaeframe headers sre activated, then caseframes are 'instantiated' by finding suitable fillers. This strategy would be justified for parsing speech only if the recognizer would assign the beat scores to LHs corresponding to caseframe headers (verbs, common and proper nouns); in this csse the search would always follow the best directions. In a previous version of SUSY we discovered that this assumption is only partially true. In Italian there are in fact words like "quale" ("which", "what") or "dove" ("where") that very often have a better score then caseframe headers end can generate useful bottom-up predictions ("dove", for instance, strongly prefers caseframes in which s LOC esse is required).

Another problem of these systems is the integration of syntax and semantics. In the approach followed by ([Hayes et si. 1986]), caseframes are augmented with syntactic constraints on case fillers and separate phrase structure rules sre developed to combine the resulting caseframes. Another approach ([Brietzmsnn, Ehrlich 1986]) is thst of developing different syntactic interpretations of frsgmsnts, thst sre then compered to find the one thst fits best. Neither of these approaches is completely satisfying. Our idea is thst, both for completeness and clarity, the grammar should be developed independently (using for instance phrase structure rules) and integrated only in a second moment with semantic knowledge (as noted by [Hayes et si. 1986] thsir spprosch gets into trouble when the syntactic complexity of the sentence increases). But we slso think it is necessary to reduce the size of the inferential activity, by applying both syntactic and semantic constraints simultaneously even when aggregating the smaller fragments.

III. THE SUSY SYSTEM

The Speech Understanding SYstem (SUSY) understands and answers continuous speech queries about Italian geography. SUSY receives a lattice of lexical hypotheses produced by an independent word recognition module ([Laface et al. 1987]).

* This reeearch has bean partially supported by the EEC ESPRIT Project n.26 "Advanced Algorithms and Architecture for Signal Processing"•

** Current adress of Massimo Poasio is Project WISBER, Univeraity of Hamburg, Postsl P.O.Box 302762 D-2000 Hamburg 36 Wast Germany.

We have adopted for SUSY the blackboard approach, very effective in such complex problem solving tasks ([Erman et al. 1980]), and developed SUSY's parsing strategy by modifying 'pure' caseframe parsing to overcome the problems previously described and make it work in the blackboard framework. Our 'impure' methodology can be summarized as follows:

- A set of Knowledge Sources (KSs) build and validate hypotheses about utterance fragments(Dedurtion Instances) that are put on the blackboard. Each 01 represents a sentence fragment at some stage of aggregation, and it is supported by a set of lexical hypotheses (the Die are an elaboration of the HWIM idea of islands [Woods 1982]). 01s have a score derived from those of the associated LHs by using the score combining methods described in [Woods 1982].

- Only sentence fragments that can be characterized both syntactically and semantically are aggregated. This way we can exploit syntactic and semantic constraints simultaneously even at the lowest levels of aggregation. Syntactic and semantic knowledge are partitioned into KSs following the principle that each KS owns the syntactic and semantic competence necessary to deal with a particular cla98 of sentence fragments.

- Top-down, expectat ion-based parsing activities are merged with bottom-up, predictive activities. An incomplete DI (goal PI) can originate a search for missing components, causing the activation of the KSs (and of those only) that deal with the classes of fragments that are required; vice versa, complete DIs (i.e. covering completely a time interval) when put on the blackboard activate KSs requiring a DI of their class as a component (for instance, a complete casefiller will activate KSs building caseframes requiring that particular case filler).

- A complete DI d carries on an interpretation of the fragment it represents; this interpretation is used by other KSs to generate interpretations of larger fragments having d as a component. An interpretation is not necessarily a case frame. Also fragments whose interpretation is a casefiller or a definite description are aggregated: these DIs can be shared by several Die to avoid repeating the asme parsing steps more than once.

- Neither syntactic rules nor caseframes are used as such, but compiled to get the KSs. Syntactic and semantic knowledge are, however, expressed declaratively and independently developed. Using this approach we get more flexibility than either completely compiled systems like HARPY or systems that make use of semantic grammars, while maintaining an efficiency and a constraining ability superior to those that can be obtained by using the two forms of knowledge independently and declaratively.

IV. REPRESENTATIONS FOR SYNTAX AND SEMANTICS

For syntactic rules we have used the formalism of Dependency Grammars ([Hays 1964]). There are two advantages in using DGs: structures based on 'governors' and 'dependents' can be easily napped onto caseframes (which is not casual, as both dependency grammar and valency theory trace back to the work by [Tesniere 1939]), and parsing is strongly anchored to input since, for a DC rule to be applied, it is necessary to find a suitable candidate for the governor.

We have adopted Conceptual Graphs (CG) [Sowa 1984] to represent caseframes. The primitives repreaentable in CGs are individuals and relations. Each individual is characterized by a type and an individual marker (imarker for short). Types are organized in a type hierarchy where any two types tl and t2 have a maximal common subtype 1l*t2. An individual 1 conforms to type I if it is an instance of a type tl <= t. A Conceptual Graph is recursively defined a3 either a concept, or a concept connected to other concepts by Conceptual Relations (CR). The Canonical Graph (CAGR), is a special CG associated to types to represent selectional restrictions.

Sowa's theory has the advantage of formal CAGRs unification operations that conserve the selectional restrictions, for the purposes of this paper we are interested in three of these operations: by join, two CAGRs are unified over a 'common' concept (a concept appearing in both CAGRs); by type restriction, the type of a concept can be substituted by any of its subtypes; by referent restriction, a generic referent in a concept can be substituted by an imarker.

Caseframes are represented in SUSY as CAGRs in which one concept, called thp head, represents the head of the caseframe: for instance, the caseframe we use for the predicate LOCATEU-IN-REGION (one of the interpretations of the verb "trovarai", "to be located"***) is represented by the canonical graph

```
[LOCATED-IN-REGION]
      -XAGNT Obligatory)->[MQUNT*PROVINCE+LAKE];
      ->(L0CObligatory)->[REGIQN].
```

Fig. 1 - The caseframe for LOCATEU-IN-RLGION

in which cases are represented as CRs. In this framework both caseframes connection and case filling are reduced to join operations, since simple casefillers are represented as concepts (i.e. conceptual graphs without CRb) and complex casefillers are caseframes themselves.

V. COMBINING CASEFRAMES AND SYNTAX TO DEFINE KNOWLEDGE SOURCES

All KSs share a common body of procedural knowledge (routines to check temporal constraints, functions that compute the score of a DI); they all include knowledge about syntactic and semantic constraints on case fillers, and must return an interpretation, that is always built using joins and restrictions only. The behaviour of a KS is then completely defined by specifying I. the class of DIs it can aggregate, n. its activation conditions, in. the set of constraints and IV. the way caseframes must be instantiated to get the interpretation.

Since only fragments that can be classified both from a semantic and from a syntactic point of view are aggregated, a DI d could be classified twice: with a syntactic category C (syntactic categories are types in the type hierarchy) and with a type T of the domain. The type that is effectively used to classify d is therefore Y = C*T (for instance, the KS above will produce DIs of class VERB*L0CATED-IN-REGI0N). The definition of KS23.1 in Fig. 2 includes a description of the compositional structure of the fragment in terms of semantic types, that is used together with the (isomorph) compositional structures of associated syntactic rules for expectations and predictions purposes: the KS above, for instance, will aggregate sequences whose first element is of class ADV*REGI0N or NOUN^REGION, the second element of class PART-RIFL*JOLLY or BE*XLLY, etc. The type JOLLY is used to classify words whose -

Since in our domain there are restrictions on what entitiB8 can be contained in others (an island can only be located in a sea, for instance) and representing these restrictions in CAGRs would lead to uneceasary complexities, we have introduced different subtypes of LOCATED for each 'containing' entity.

ayntsctic classification only is relevant (and that can eventueily be skipped).

```
(DefKS KS23.1

  {; Compositional Constrsints
  LOCATED-IN-REGION « REGION JOLLY <HEADER> MOUNUPROVINCE+LAKE

  H Associsted Syntactic Rules
  (Dr23 Dr23s 0r23b Dr23c Or23d OR23e)
        ;jVERB a NOUN PART-RIFL <G0VERN0R> NOUN
        ;;VER8 « NOUN BE <Q0VERNOR> NOUN
        ;;VER8 = NOUN PART-RIFL <G0VERNOR> PROPER-NOUN
        t;VERB s ADV PART-RIFL <G0VERNOR> NOUN
        UVERB s AOV BE <G0VERNOR> NOUN
        ;;VERB s AOV PART-RIFL <G0VERN0R> PROPER-NOUN

  ;; Act ivet ion Condition
  G(*>X) s*> ACTION (?X LOCATED)

  ;; Casefraaes Instantiation Rule
  •RIS((LOCATED ! ♦
              AGNT ?Z
              LOC *>Y)) <z»
        RIS(RECION (ACTION ! ♦ ?X ?2))
        RIS(MOUNUPROVINCE*LAKE (ACTION ! • 7W 7Y)))
```

Fig. 2 - KS definition

KS definitions ere the result of e compilstion. They are produced froa dependency rules, ceseframes and mopping rules. For example, the definition of KS23.1 hee been obteined froa the set of aeeociated dependency rules in Fig. 2, *one* of which le shown in Fig. 3.8 with its mapping rule (Fig. 3.b), and froa the caaefraae in Fig.l.

```
s. (Dr23 (VERB   NOUN PART-RIFL <G0VERNOR> NOW)
      VERB (MODO (INDIC)) (TEMPO (PRES)) (PERS (3))
           (NUM ! ?N) (TRANS (VIT))
           (COMPLEMENTO ! (STAT0-LU0G0))
           (RIFL (RIFL))
      NOW (T-COMPL ! (STAT0-LUOG0))
      PART-RIFL NIL
      NOUN (NUM I ?N))

b. (Mr23 ((LOCATED-REGION LOCATED-PROVINCE WASH) *
               LOC (JOLLY) <HEAD€R> AGNT))
```

Fig.3 - A dependency rule with ite Mapping to caeefranes.

The compilation is facilitated by the atructural similarity between dependency rules and caeefranes (both are eeaentielly baaed on the idee of e 'head' Modified in soae wey). The compositionel part of the KS le derived from that of the dependency rule by substituting syntactic types with the velue restrictions on ceee fillers specified in the caaefraaei the position of esses in the fregnent la speciflsd in the napping rule. Only those caaaa that can be rilled uaing infor metion present in the fragment *mre* included; thie pre-eelection evoide searching for information that could not be found. The activation conditions are aeeociated to typaa in the knowledge baee.

Having partitioned the knowledge in independent chunks, when a new ayntactic rule is added it is never neceessry to recompile the entire eet of KSet either a previously defined KS la modified to take into account the new rule, or new KSa are added to the existing eet. In this latter caee, since not all eyntactic/eeaentic combinations are admissible, the number of KSa that are generated is limited. The cese in which s new type is added to the domain has more aarioua consequences: with our currant aat of about 130 dependency rules in the average about 15 new KSa ara defined whan adding a new entity type. These problems ara typi-cal of semantic grammars; in our caaa there is the advantage that it la the system itself that generates the new KSa.

VI. CASEFRANCE INSTANTIATION

The aaaning of a word in the dictionary is a aat of predicates able to activate the appropriate KSai for instance, the meaning of the word "trova" includee the predicate

ACTI0N(TR0VA LOCATED)

that activates KS23.1 of Fig.2. That KS includes a cssframe instantistion rule that produces the inatantiated casefremee constituting the interpretation of the sentence fragments it analyzes. By this approach, information duplication is avoided, ainee a single csseframe is shared by many words, and caaefreams instantiation is more efficient, since the inatantiation rule specifies 'a priori' how the caseframe associated with the header must be joined with those aasociated with the component OIs.

Only two operations ara needed to build the interpretation: referent reatrictiona are used to fill concepts, and Joins are uaed to merge concepts representing cases with canonical graphs representing caeefillere. Repreaenting a canonical graph aa a term and generic referents aa logic vanablee, joins can be effectively implemented aa unification oparationa using a PROLOG interpreter after all components indicated in the composition rule have been found. The instantiation rule of KS23.1 (reproduced in Fig.4.a) together with the CGs carried by DIa of class HOIMT+PROVINCE+LAKE (4.b) and REGION (4.c) producaa the inatantiated caaeframe *in* (4.d). This unification haa a vary low coat, since the interpreter runs in a spacial context in which only the results carried by accepted components (the "ria" predicatea in Fig.4) have been aaaartad.

```
a.   RIS((LOCATED ! *
            AGNT ?Z
            LOC ?Y)) <z»
         RIS(MOUNT+PROVINCE+LAKE (ACTION ! * ?X ?Z))
         RIS(REGION (ACTION ! * ?W ?Y))

b.   RIS(MOUNT+PROVINCE+LAKE
        (ACTION ! * AGNT (MOUNT ! ROSA)))

c.   RIS(REGION (ACTION ! * LOC (REGION ! ?r)))

d.   (LOCATED ! *
            AGNT (MOUNT ! ROSA)
            LOC (REGION ! ?R)))
```

Fig.4 - Caaeframe inatantiation

VII. INTEGRATING TOP-DOWi AND BOTTOM-UP CASEFRAME PARSING

Suay always moves from the best Lexical Hypothesis or Deduction Inetanee. A top-down, expectation baaed activity is started if a caaefraae header is selected} a bottom-up predictive activity is initiated whan a DI representing a caaafiller or a LH repreaenting a case marker are sslscted.

Let us consider the sentence (translated word by word)t

IN QUALE REGIONE SI TROVA IL MONTE ROSA?
(In which region le located mount Rosa?)

The LHs involved in the solution, ordered by scores (the best first) are:

1) QUAL (which)	1.267
2) IN (in)	1.301
3) MONTE (mount)	1.371
4) SI	1.485
5) REGIONE (region)	1.560
6) TROVA (is located)	1.610
7) ROSA (Rosa)	1.629

Fig.5 - The LHs involved in the solution

The process leading to the solution is displayed in Fig.6:

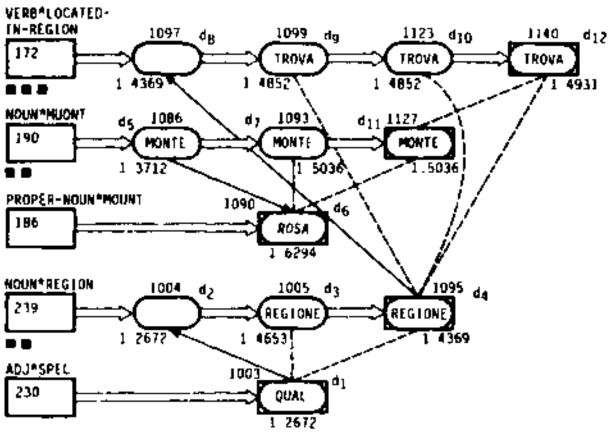

Fig.6 - The deductive process leading to the solution

- The best LH, "qual" is not a caseframe header but an interrogative adjective that acts as a casemarker together with the preposition "in". The KSs whose activation conditions are satisfied by the predicates associated with the word "qual" are triggered. One of these KSs requires no other components and produces the complete Deduction Instance d1 of class ADJ*SPEC, that is put on the blackboard.
- Since d1 has a good score the analysis proceeds along this path, causing a bottom-up activation of all the KSs having a component of class ADJ*SPEC. The activation of one of these, a KS of class NOUN*REGION, generates d2. d3 is generated when a suitable caseframe header is found (the LH "regione"); later, also the casemarker "in" is found, and the complete DI d4 is generated.
- As the score of d4 is worse than that of the LH "monte", this branch of the inferential process is momentarily abandoned. The caseframe header "monte" causes the activation of all KSs instantiating MOUNT type caseframes. The KS of interest requires only an individual referent, that is found (the LH "Rosa"). This process ends in the DI d7 of class NOUN*MOUNT.
- The score of d7 is not very good, due to the bad quality of the LH "Rosa". The deduction path that had produced d4 (with score 1.4369, better than that of d7) is resumed. The KSs requiring a LOC case to be filled by an entity of type REGION are activated; among these, KS23.1. The DI d8 is generated by KS23.1 when d4 is accepted as casefiller; d9 when the header of the caseframe for the type LOCATED-IN-REGION is found (the LH "trova").
- The caseframe for LOCATED-IN-REGION also requires the AGNT case to be filled. A suitable filler is d11 (derived from d7 after the aggregation of the article "il"). The final, complete DI d12 is obtained; note that this last step involves the confluence of two deductive processes that developed independently. The three canonical graphs of KS23.1 are joined to give the final interpretation, the instantiated caseframe of fig. 4d.

VIII. CONCLUSIONS

Two modifications have been proposed to caseframe parsing in order to make it work effectively in a blackboard framework: a more flexible strategy alternating top-down and bottom-up parsing, and the compilation into the same knowledge source of syntactic and semantic knowledge. Due to the integration of syntactic rules in the caseframe approach, the system is able to handle fairly complex sentences with a reasonable efficiency.

This approach has been implemented and tested and we are now improving the compiler and the overall blackboard architecture.

IX. REFERENCES

1 Brietzmann, A., Ehrlich, U., "The role of semantic processing in an automatic speech understanding system", Proc. COLING-86, Bonn, West Germany, 1986

2 Erman, L.D., Hayes-Roth, F., Lesser, V., Raj Reddy, D, "The Hearsay-II Speech Understanding System: Integrating Knowledge to Resolve Uncertainty", Computing Surveys, v.12, n.2, June 1980

3 Fillmore, C.J., "The case for case", in Bach, Harris (Eds.), Universals in Linguistic Theory, Holt, Rinehart, and Winston, New York 1968, 1:90

4 Hayes, P.J., Carbonell, J.G., "Multi-Strategy Construction-Specific Parsing for Flexible Data Base Query and Update", Proc. IJCAI-81, Vancouver, August 1981, 432:439

5 Hayes, P.J., Hauptmann, A.G., Carbonell, J.G. - Tomita, M., "Parsing Spoken Language: a Semantic Caseframe Approach", Proc. COLING-86, Bonn, West Germany, 1986

6 Hayes-Roth, F., "Syntax, semantics, and pragmatics in speech understanding systems", in W.Lea (ed.), Trends in speech Recognition, Prentice-Hall, Englewood Cliffs (NJ), 1980

7 Hays D.G., "Dependency theory: a formalism and some observations", Memorandum RM4087 P.R., The Rand Corporation, 1964

8 Laface, P., Micca, G., Pieraccini, R., "Experimental results on a large lexicon access task", in Proc. ICASSP-87, Dallas, 1987

9 Lowerre, B., The HARPY Speech Recognition System. Computer Science Department, Carnegie-Mellon University, April, 1976

10 Sowa, J.F., Conceptual Structures, Addison-Wesley, Reading (MA), 1984

11 Tesniere, L., Elements de syntaxe structurale, 1st edition, Librairie C.Klincksieck, Paris, 1959

12 Woods, W.A., "Optimal Search Strategies for Speech Understanding Control", Artificial Intelligence, v.18, 1982, 295:326