ISCA Archive
http://www.isca-speech.org/archive

4th European Conference on
Speech Communication and Technology
EUROSPEECH '95
Madrid, Spain, September 18-21, 1995

INTEGRATING PARTIAL SYNTACTICAL ANALYSIS AND PLAN RECOGNITION FOR UNDERSTANDING DB NATURAL LANGUAGE QUERIES

Cristina Delogu°, Andrea Di Carlo°, Rino Falcone*

°Fondazione Ugo Bordoni
Via Baldassarre Castiglione, 59 00142 Roma - Italy
cristina@fub.it

*CNR - Istituto di Psicologia
Viale Marx, 15 00100 Roma - Italy
e-mail: falcone@vaxiac.iac.rm.cnr.it

10.21437/Eurospeech.1995-301

ABSTRACT

In this work we propose a framework for speech understanding in Natural Language DB accessing based on the cooperative behaviour of the system which identifies its goals with user's goals and executes its own plans for satisfying these goals.

The purpose of this work is to verify that in this particular human-machine interaction, by starting from some few recognized keywords, it is possible to reconstruct the user's goal and then to satisfy it. The process of user's goal recognition is realized by a chart-based mechanism for plan recognition. This approach allows us to use the same formalism to represent both the low level (i.e. the syntactical analysis) and the high level (i.e. plan recognition) analyses.

1. INTRODUCTION

We propose a framework for speech understanding in Natural Language DB accessing based on the partial syntactical analysis of the best word sequence provided by the speech recognizer, on the interpretation of local syntactical structures as signs of the linguistic actions performed by the user in order to query the DB, on the syntactical recognition of user's plans, and on cooperative behaviour of the system which identifies its goals with user's goals and executes its own plans for satisfying these goals.

The framework has been experimented on a set of 584 "A" sentences from the ATIS corpus, naturally translated in Italian by a "naive" translator. Sentences of the type A are sentences which contain a complete natural formulation of the query to the DB; that is sentences in a direct interaction without dialogue.

The user-system interaction is a binded interaction both in the domain and in the approach. In fact, the user can realize just one kind of action: the information request. Moreover the only possible requests the system can "understand" should be about the ATIS DB.

The user's plan is unknown to the system which, by matching the hypothesized structures in the input with the plans in its library, tries to individuate it: generally, it is possible that the user knows very well neither the exact content of the DB nor the exact relationships about these information contents.

The results are good considered in the perspective of future development of the system towards user-system dialogue. Further work is foreseen in the framework of plan based dialogue management.

2. DESCRIPTION OF THE SYSTEM

Input to the system is a list of items which can be or not be in the known lexicon: a list of words and non-words. For example the list

[per,favore,vorrei,un,volo,da,dallas,a,boston]
(please, I would like a flight from dallas to boston)

where the items [per,favore] (please) are not in the lexicon.

The lexical analysis consists in recognizing each item as being or not being in the lexicon: for words in the lexicon, the lexical categories ('da' is *sfrom*, 'dallas' is *cityname*) and their localization in the input list ('volo' is the fifth item in the list) are provided. These information are introduced in a data structures called *chart*: a graph with nodes (*vertices*) and arcs (*edges*) [1].

For example, the edge

[6,7, *cityname*,dallas,nil]

is an inactive edge, i.e. a complete interpretation of the included input portion (Figure 1).

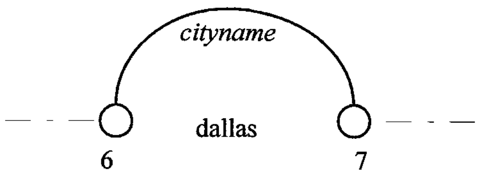


Figure 1

All the set of edges provided by the lexical analysis is the chart at the initialization step.

On the basis of the lexical edges and of a set of context free rewriting rules, the system provides a local syntactical chart parsing which groups words in phrases and locutions; that is, we obtain complex syntactical structures for some portions of the original list. These structures are represented in the chart by inactive edges. For example (Figure 2),

[5,7, *departure_city*, [[*sfrom*,da], [*cityname*,dallas]], nil].

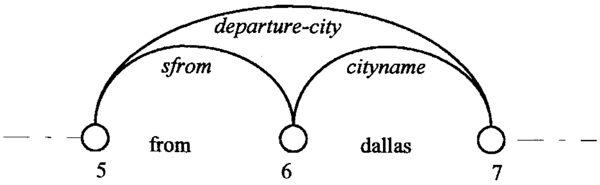


Figure 2

Groups (phrases and locutions) recognized by the local chart parsing are interpreted as linguistic signs of action performed by the user in order to communicate her intentions, her targets and her requests. These actions are introduced in another chart, called the Action Chart, as inactive edges to realize the Chart Plan Recognition [2].

For example, the syntactical edge

[5,7,*departure_city*,[[*sfrom*,da],[*cityname*,dallas]],nil]

introduces, thanks to special activation rules, the inactive edge

[5,7, [*indication_for_departure_city*,dallas], nil,nil]

into the Action Chart. Using a regular syntactical model (*Plan Library*) and through a new stage of chart parsing, these user's actions are grouped in the recognition of user's plans.

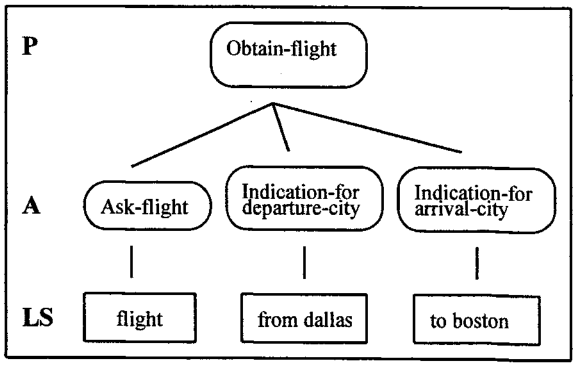
For example, starting from the initialization status of the Action Chart:

[[4,5,[ask_flight],nil,nil],
[5,7,[*indication_for_departure_city*,dallas],nil,nil],
[7,9,[*indication_for_arrival_city*,boston], nil, nil]]

the system can progressively build and introduce the complex edge

[4,9,[*obtain_flight*,Flight_Id,dallas,boston,Airline_Code,Flight_Number,Departure_Time,Arrival_Time], <recog-syntact-struct>,nil]

into the Action Chart, with the meaning that the user wants to obtain some information (such as the Identification and the timing of the flight) about some flight from dallas to boston (see for example Figure 3).



LS = Linguistic Signs
A = Actions
P = Plans

Figure 3

The plan library has been realized starting both from a corpus of sentences and from the relationships among contents in the DB.

Through individuating actions in the input and linking them to the memorized structures (plans), it is possible to establish several correlations among these actions in order to recover some possible omitted actions.

When none plan in the library corresponds with a plan into the user's input there are two main possible reasons:

i) The plan model library does not completely cover the user's possible real plans (this is always possible);

ii) Some keywords during the process from the speech to the lexicon are loosen.

Then, it is possible to think to introduce an activation mechanism such as a parsing mechanism, where it is possible to individuate partial plans, i.e. plans not completely recognized.

The recognized user plan is identified with a plan with the same name and the same argumental structure, in the system plan library able to access the DB and to retuarn the retrieved information. The plans of the library are designed according to certain user's request classes. Each plan is characterized by the presence of a typical request action (*ask_x: ask_flight, ask_company, ask_prize,* etc.) in the decomposition structure. The other actions in the plan allow to give more specificity to the request itself (*indication_x*). In fact, there are also mixed request classes. Given the recognized user's plan, the system is able to individuate a procedure for returning the requested information.

3. EVALUATION

An objective evaluation of the system performance was carried out by comparing system's answers to the "A" ATIS sentences (naturally translated in Italian by a "naive" translator) with the ATIS canonical answers. Sentences of the type A are single unambiguous intelligible utterances without context [3]. Canonical answers are the corrected version of the answer returned under the wizard's control during the ATIS corpus collection.

We categorized the system answers into several types: OK (the same canonical answer), CP (*Correct Plan,* i.e. when the answer differs from the canonical one but the plan produced by the system is the correct plan), WP (*Wrong Plan,* i.e., when the answer differs from the canonical one and the plan produced by the system is wrong), UP (*Uncompleted Plan,* i.e. when the system doesn't give any answer but produces one or more uncompleted plans).

584 "A" sentences were used. Among these, 37 sentences were rejected because they are not in the ATIS domain, and 83 sentences were rejected because they are not in our domain. The domain used in the experiment is a sub-set of the ATIS domain. An example of sentences out of the ATIS domain is "Reserve a ticket on Continental flight 215 for Daphne Jones, for Wednesday, 26 April 1990". An example of sentences out of our domain is "What state is Dallas in?".

3.1 Results

From 584 test sentences 120 were rejected (37+83). The scores of 464 answers are shown in Table 1.

ANSWER TYPES	NUMBER	%
OK	285	61.4
CP	56	12.1
OK+CP	**341**	**73.5**
WP	64	13.8
UP	59	12.7
WP+UP	**123**	**26.5**
tot-answer	464	100

Table 1

In this prototypical phase of the system, we were interested in exploring how large and complex the lexicon for an air travel information service should be. To this purpose, an analysis of the sentences used in the experiment was performed. The results showed that the lexical requirements for an air travel information service are manageable for current speech technology. In fact, from the 3.412 tokens in the 584 experimental sentences, only 380 types have been found[1]. They are subdivided into the domain-independent lexicon (DIL) and the domain-specific lexicon (DSL).

DIL (1.818 tokens, 177 types) is made of all those terms that can be considered as general terms and that are easily transportable from one application domain to another, such as function words, adverbs, modal verbs, comparative or superlative modifier.

DSL (1.594 tokens, 203 types) has been further subdivided into Numbers (133 tokens, 73 types), Proper Nouns (such as airline, airport, and city names) (628 tokens, 30 types), Key-Words (730 tokens, 79 types), and other domain-specific terms (104 tokens, 21 types) different from the K-Words.

4. CONCLUSIONS

The results show that the system obtained a good performance in 73.5% of cases. We think that this is a satisfying result from a point of view of future developments of the system towards user-system dialogue. In this perspective, CP and UP (and in some cases also WP) answers could be recovered during the dialogue: the system always produced completed (even if wrong) or uncompleted plans from which it could retrieve useful information to be used for beginning the dialogue.

Next steps in our work will be some development of a more flexible dialogued interaction.

5. REFERENCES

[1] Stock O., Falcone R., Insinnamo P. "Bidirectional charts: A potential technique for parsing spoken natural language sentences", *Computer Speech and Language*, 3 (3): 219-237, 1989.

[2] Falcone R., Castelfranchi C. *CHAPLIN: A Chart based Plan Recognizer.* Procs. Int. Conf. of Avignon, France, 1993, pp. 31-42.

[3] Hirschman L., Dahl D.A., McKay D.P., Norton L.M., Linebarger M.C. *Beyond Class A: A Proposal for Automatic Evaluation of Discourse.* Procs. DARPA Speech and Language Workshop, Hidden Valley, PA, 1990, pp. 109-113.

[4] Litman D.J., Allen J.F. "A plan recognition model for subdialogues in conversations". *Cognitive Science,* 11, 163-200, 1987.

[1] We intend with *type* a linguistic form, and with *tokens* all the specific occurrences of that linguistic form.