Cooperation and Representation of Syntactic-semantic and Pragmatic Knowledge in a Natural Language Task Oriented Spoken Dialogue System

P. Mousel, J.M. Pierrel et A. Roussanaly
CRIN / INRIA Lorraine
B.P. 239, F-54506 Vandœuvre-lès-Nancy cédex, France

Abstract
Natural language task oriented spoken dialogue systems require highly integrated processing of various knowledge sources: syntax, semantics and pragmatics. This paper presents the DIAL system being developed at present in Nancy. The system's architecture will be unfolded as will be the linguistic models used by the system's components.

1. INTRODUCTION
Today's man-machine communication techniques still seldom use spoken natural language. We think this is due to the fact that until the early eighties the main research efforts were spent on recognition and understanding of single utterances. Effective use of spoken natural language in man-machine communication however requires the entire man-machine dialogue's understanding and managing [9] [19] [20].

Having shown the feasibility of speech recognition with MYRTILLE I and MYRTILLE II [18], we are now developing two different types of dialogue systems:

• the PARTNER system, based on a strongly restricted universe [15];
• the DIAL system, dealing with a more complex universe and cognitive activities.

They use various and manifold knowledge sources and ask for enhanced semantic and pragmatic processing. This cannot be done by simply adding a dialogue managing component to a speech recognition module, interaction between these two levels is too complicated. A high degree of integration is necessary for real dialogue understanding and managing. This leads us to think of a strong cooperation between three components usually dissociated: syntax, semantics and pragmatics.

We chose to present the DIAL system in this paper. After briefly outlining the objectives and architecture of the DIAL system, we shall focus on the information flow within the system. Finally, the syntactic, semantic and pragmatic models used in the system will also be unfolded.

2. DIAL: A NATURAL LANGUAGE TASK ORIENTED SPOKEN DIALOGUE SYSTEM

2.1. Particular Features of Natural Language Task Oriented Spoken Dialogues
The design of natural language task oriented spoken dialogue systems is highly determined by the particular features of this family of dialogues [1] [2] [3] [5] [6] [14]:

• The dialogue's control being shared in this type of communication, menu-driven systems are inappropriate.
• Spoken language introduces nondeterminism and problems due to the communication channel.
• The dialogue is task oriented, i.e. the final objective is the achievement of a task.
• General public applications exclude previous training and require the use of natural language, in fact a sublanguage, which is different from the language as a whole, in mainly lexical, semantic and syntactic structure.
• The user's behaviour is altered by an automatic system, then reducing the vocabulary and the utterances' diversity but increasing the number of exchanges.

We also think that it makes sense to suppose the user is cooperative [10].

2.2. Towards a Distributed System Architecture
The evolution of speech understanding research is shown by the experimental systems that have been developed in several laboratories: KEAL [12], MYRTILLE I [17] and MYRTILLE II [18], ESOPE [11], HWIM [22] and HEARSAY II [8].

We looked at the way these systems take into account the previously mentioned characteristics. Most of them only deal with speech understanding. If they integrate a dialogue component, it has been added during a further phase of the systems' development as for CADI, the dialogue component of the KEAL system [16]. They take into account problems arising with spoken language but they suppose the communication channel to be in a perfect state. Some of them only achieve simple tasks (telephone exchange), others more complex ones (document retrieval, meteorological information center, travel agency). Some of them are designed for specialist users, others for non-trained users. They are all based on various linguistic knowledge sources with a highly centralized control, either because of the existence of a supervisor module or the existence of a central information structure.

These systems show that the use of manifold knowledge sources is absolutely necessary. They also tend to strengthen cooperation between different knowledge sources by more and more heterarchical system architectures. Cooperation is either ensured by centralised processing control (supervisor) or centralized information control (blackboard). Supervisor control imposes a strict order upon the process sequence already at system's design. Blackboard control does not but since component interaction in this context is well defined, it is too general and thus unavailing.
This leads us to define a system's architecture where cooperation is based on distributed control. Components communicate two by two according to the producer-consumer model. Figure 1 shows the architecture of DIAL with:

- APHON, the acoustic-phonetic component that builds a phonetic lattice starting from the acoustic signal corresponding to the speaker's utterance;
- PROSO, the prosodic component that detects pertinent prosodic markers in the acoustic signal;
- LEX, the lexical component that provides a lexical lattice starting from the phonetic lattice;
- SYNSEM, the syntactic-semantic component that produces syntactic-semantic structures starting from the lexical lattice;
- DIALOG, the pragmatic component that answers the speaker's request starting from the syntactic-semantic structures or the lexical lattice.

It also shows the information exchanged by the different components. This information is of two types:

- validation requests, i.e. hypotheses made by a high level component to be validated by a low level component;
- local data, i.e. results of a component's self-triggered processing.

The following sections will be confined to the understanding components of our system, i.e. SYNSEM and DIALOG. Firstly, their interfaces will be presented, then we will detail their internal structure.

2.3. Component Interaction in the DIAL System

2.3.1. SYNSEM Component Interface

SYNSEM produces the syntactic-semantic structure of a speaker's utterance using syntactic and semantic criteria. Therefore it has to exchange information with other components of the system, mainly:

- DIALOG, from which it gets syntactic-semantic hypotheses, i.e. predictions on the structure of the utterance that has to be parsed. It tries to validate these hypotheses and provides the utterance's syntactic-semantic representations to the DIALOG component.
- LEX, to which it sends word hypotheses and from which it gets validated words or well recognized words.
- PROSO, from which it gets syntagm boundaries (up to date, these informations are not used by SYNSEM).

2.3.2. DIALOG Component Interface

DIALOG structures the current dialogue using pragmatic criteria. It therefore exchanges information with other components of the system, mainly:

- SYNSEM, to which it sends syntactic-semantic hypotheses, i.e. predictions on the structure of the utterance that has to be parsed next and from which it gets the utterance's syntactic-semantic representations.
- LEX, to which it sends word hypotheses and from which it gets validated words and recognized words.
- APHON, from which it gets acoustic-phonetic markers of which the utterance's length as well as information about the communication channel status.

2.4. Component Architecture in the DIAL System

2.4.1. SYNSEM Component Architecture

Figure 2 shows the internal structure of the SYNSEM component, mainly its interface with the other components and the knowledge sources it uses. To realize its objective, i.e. build the syntactic-semantic structure of the speaker's utterance, it integrates a number of functions:

- Different parsing techniques have been considered, parsing based on syntactic and semantic knowledge and syntactic-semantic correspondence rules as well as parsing based on semantic knowledge only. Parsing may be triggered by the DIALOG component's hypotheses, i.e. top-down parsing, but also by well recognized words from the LEX component, i.e. bottom-up parsing.
The parsing of an utterance requires management of information of two types: the theory space and the hypothesis space. The theory space is the set of the current utterance's syntactic-semantic representations SYNSEM has built up to a given moment. Initial theories are built with DIALOG's syntactic-semantic hypotheses and well recognized words from LEX. These theories are used by SYNSEM to make word hypotheses that are sent to the lexical level for validation. The validated words allow SYNSEM to progressively complete the theory space. Finally the syntactic-semantic representations are passed over to the DIALOG component. The hypothesis space on the other side is necessary in order to avoid redundancy of word hypotheses.

2.4.2. DIALOG Component Architecture

Figure 3 shows the architecture of the DIALOG component, mainly its interface with the other components and the knowledge sources it uses. Having to structure the current dialogue, it integrates several functions:

- DIALOG makes a contextual interpretation of the utterance's syntactic-semantic representation and the validated words based on contextual interpretation rules with a twofold objective: homogenization of the different representations and solving of the references to utterances preceding the current one.
- Once interpreted, the utterance will trigger a reasoning phase based on a task model. Reasoning will set default values, infer implicit information and manage the task status, i.e. determine missing information and detect inconsistencies as well as the exceeding of the system's limits.
- Thanks to a dialogue model, the dialogue managing phase then provides a reply to the speaker and predictions on the next utterance.
- A heterogeneous data structure, the dialogue history, makes the different processes communicate. It contains a chronological representation of the dialogue, a classification of the utterances and the task status.

3. SYNTACTIC-SEMANTIC KNOWLEDGE

We presented the information SYNSEM and DIALOG exchange as well as the processing they do. Now we will focus on the knowledge they make use of, beginning with SYNSEM.

Although modern linguistics do not clearly distinguish syntax and semantics any more, it seems reasonable to us to have them separated from a purely operational point of view. We call syntax positional constraints and semantics combinatorial constraints valid for a given language. Syntax and semantics being closely related, correspondence rules define these relations.

3.1. Syntax

Syntactic modelization of our application's sublanguage is based on an ATN-like model, the RNP or "Réseaux à Noeuds Procéduraux" (Procedural Node Networks) [18]. Compared to ATNs a considerable advantage of RNPs is the strict separation between knowledge representation and its processing thus facilitating design of adaptable systems. Their descriptive power depends mainly on the nodes' contents, it could reach that of a Turing Machine. Some thirty networks have been defined so far [7].

3.2. Semantics

Semantic knowledge of the DIAL system is based on a case grammar oriented model [7]. The design of an adaptable system also calls for this model to strictly separate knowledge representation and its processing. Semantic knowledge essentially defines case frames for each verb of our application language. However in order to limit the number of rules, the verbs have been regrouped into classes called primitives and thus case frame constraints are defined for each primitive rather than for each verb. We identified some twenty primitives.

3.3. Syntactic-semantic Correspondence Rules

Syntactic and semantic parsing proceed simultaneously. Syntactic parsers try to build up a "syntactic tree" whereas semantic parsers try to build up a "semantic tree" of the utterance. Both structures are strongly related, the semantic structure of an utterance depending on the syntactic one. This dependency is expressed by correspondence rules that, to an entity with a given syntactic function, associate a semantic function [7].

4. PRAGMATIC KNOWLEDGE

DIALOG also uses pragmatic knowledge that will be detailed in this section.

4.1. Interpretation rules

The set of interpretation rules allows the DIALOG component to homogenize the utterance's various representations as well as to solve references thereby facilitating further processing. The formalism used for writing rules is similar to first order predicate calculus and is based on the concept of formula [21]. The interpreter is in fact an inference engine with a working memory, a fact base and a rule base containing the set of interpretation rules thus allowing easier design of an adaptable module. A rule is made of a condition part and an action part, the first specifying facts to be checked in the working memory whereas the second describes actions to be carried out. Possible actions can be summarized as follows: adjunction or suppression of information in the working memory, retrieval of information from the dialogue history and partial building of an interpretation.

4.2. Task Model

The task model contains knowledge related to the task, the most important being:
4.3. Dialogue Model

The dialogue model integrates a descriptive as well as a dynamic point of view. The first considers a dialogue as a hierarchical and functional process based on the concepts of exchange and language acts [13]. The second considers a dialogue as a process obeying to standard scenario [4] using formalisms conciliating declarative and procedural aspects of knowledge. Thus an application is a set of scripts comparable to a classic program.

In the current version of our system, the dialogue model is in fact a procedure where the sequencing is guaranteed by subprogram calls and where predefined actions and access to the dialogue history are provided by a function library.

5. CONCLUSION

DIALOG and SYNSEM at the present time are evaluation prototypes. Communication between components is simulated by interactive means. All functions are not yet implemented. Some of them however have already been tested:

- SYNSEM integrates syntactic top-down parsing, managing of the theory and the hypothesis space and dialogue module. In order to get more significant results, we are now focussing on the knowledge base's extension. The component until now also misses a generator module for natural language replies.

The next step in the system's development will be the implementation of an automatic interface between the utterance understanding and dialogue managing components. We also try to think about tools facilitating the building of this type of systems in order to dispose of a real programming environment for natural language man-machine communication systems.

6. REFERENCES