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

6th European Conference on
Speech Communication and Technology
(EUROSPEECH'99)
Budapest, Hungary, September 5-9, 1999

PROCESSING OF ANAPHORIC AND ELLIPTIC SENTENCES IN A SPOKEN DIALOG SYSTEM

Jana Ocelikova, Vaclav Matousek
University of West Bohemia in Pilsen
Univerzitni 22, 306 14 Plzen, Czech Republic
jnetrval | matousek@kiv.zcu.cz

10.21437/Eurospeech.1999-318

ABSTRACT

This paper presents an approach to processing of ambiguous requests in spoken dialogs in information train timetable service systems. The main problem of human-machine interaction is in the fact, that speaking human does not express some essential information, because it results from the receiver's knowledge and the dialog context. Therefore the real understanding system have to combine the incomplete semantic information extracted from the analyzed utterance with the previous data stored in the dialog history path. This is provided in the developed dialog system by the dialog module. Its is based on the representation and storage of facts extracted from the previous steps of dialog (dialog path), and on the creation of simple knowledge bases containing the reasoning rules linking the meaning detected in the analyzed user's utterance with the facts of the dialog path.

INTRODUCTION

The main problem of human-machine interaction can be seen in the presumtion, that both in the dialog participating subjects have approximately equal knowledge and experience in the dialog domain. Therefore, the used dialog model have to take the dialog context into account, using particular knowledge base (*a dialog history*), which is build during the dialog, and simple types of references (anaphora, ellipses) can be processed. The dialog system gains then the features of natural reasoning, of incoherence detection and internal correction, and of anticipation and prediction.

The main components of the developed system prototype are speech input/output interface, acoustic-phonetic recognizer, linguistic processor, and dialog manager (see [5]}. The acoustic-phonetic recognizer, the linguistic processor and a part of the dialog manager are integrated into the subsystem of a spoken dialog interface. The interface of the developed system provides the following three principal functions: the interpretation of user (customer) utterances, the generation of system utterances (answers containing required data and/or clarifying question) and the user-friendly management of a coherent and natural dialog.

1. DIALOG MANAGER

The task oriented subsystem of dialog control and management (*dialog manager*) accepts the semantic representation of the user's utterance and performs its interpretation within the current dialog context.

The dialog model takes context into account, using particular knowledge base (a dialog history), which is build during the dialog. Multiple types of references (anaphora, ellipses,...) may be processed. The system may be capable of reasoning, of incoherence detection and internal correction, and of anticipation and prediction.

At each stage in any possible dialog, the designer attempts to answer question *"What could happen next?"*. This question can be answered at multiple levels. For example: In a classical human dialog the earlier understood information is often modified, specified and can be reversed later. A human has no problem with this situation, he accepts it and the effect of this situation too. But the system "doesn't understand", therefore it must generate a clarifying question. In the dialog interpretation process the semantic interpreta-tions and the dialog model are matched, deciding the subsequent steps of cooperative user interaction in the structured dialog model. Using this partitioned interactional model, dialog management is relatively independent on the language and of the information service domain. By automatic production there is necessary to save dialog's history.

The significant tasks of the dialog control we can summarize to the following list :

- initial greeting,
- generation of a first question,
- output of the question to the text-to-speech component,
- passing control to speech recognition and speech understanding,
- modifying these components according to the current system status,
- processing of the results obtained from speech understanding,
- combining this information with the current system status,
- selection of the next question based on the new system status,
- generation of the exact form of this question,

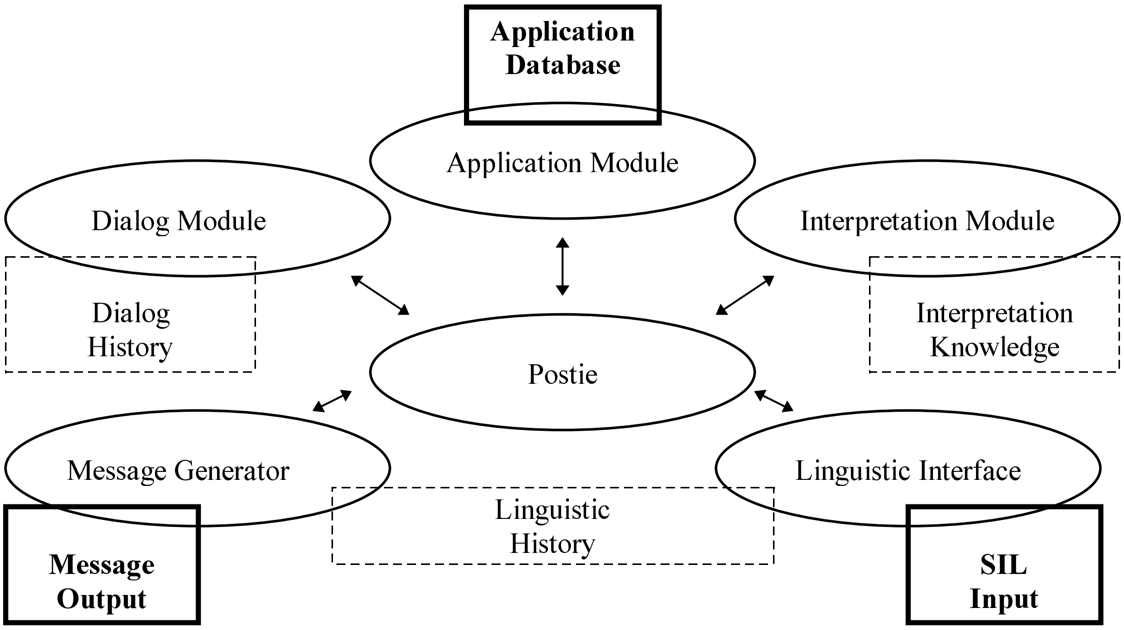


Figure 1: Modular structure of the Dialog Manager

- verification of previously obtained results,
- access to the application database,
- output of the query results, possibly after converting them into a suitable form,
- dealing with special situations during the dialog, like errors from the database, repetition of a question, repetition of results,
- monitoring of the dialog progress.

Dialog manager was designed by the way, that incorporates pragmatic interpretation formation, formulas describing the database relating to the application, interpretation rules of earlier history of the current dialog and their relationships (dialog model), as well as a set of dialog goals representing intentions of the system. The following basic program modules were incorporated in the dialog manager:

- linguistic interface module, providing the communication with the parser,
- interpretation module, setting user utterance interpretation in the current dialog context,
- task module, handling with database query and language adaptation task performance,
- dialog module, planning system utterance formulation within the pragmatic information context and interpretation,
- message planning module, generating the system message and protocols.

The structure of the dialog manager (see Fig. 1) is made full open to enable the system extension by new modules. Their incorporation is performed under control of the dialog control module in which the message passing protocol was used to provide the communication among the modules. With the help of this subsystem we demonstrate the language and application independence of the dialog manager and its possible adaptation to the Czech language and to the required application only by the adaptation or simple modification of message planning and task modules as well. The more detailed description of the developed dialog manager can be found in [5, 6].

2. LANGUAGE AMBIGUITIES

Each system which analyses sentences of natural language has to deal with a problem of sentence *representation.* Every sentence, represented on the input by a string of words, has to be converted into an internal structure. This internal structure has to be well defined and must distinguish all alternative meanings of the original sentence. In other words, a natural language sentence may have several distinct, possible representations, with each representation identifying a different meaning. Consider, for example, the following sentences:

1. *The conductor sold the last ticket to Prague.*
2. *He sold it to a young lady.*

A human would not have a problem understanding the above sequence. A computer system, on the other hand, would not be able to create an internal representation of this sentence, unless several issues were clarified.

2.1 Lexical Ambiguity

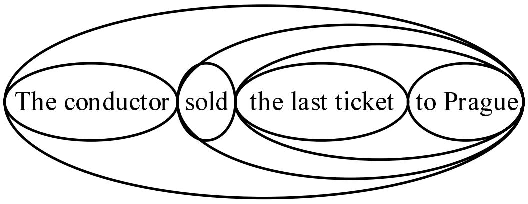
The computer system would, first of all, have to deal with the fact that most of the words have multiple meanings. For example, if we look up the word *ticket* in a dictionary, we find that it can be either a verb or a noun. Determining the correct category poses a problem of *lexical syntactic ambiguity*. Having determined the correct syntactic category, we have to further decide, what semantic concept a word represents. The meaning behind the noun *conductor*, for example, can be either a person who sells tickets on a bus, a person who conducts an orchestra, or a substance that passes heat or electricity. Deciding which of these meanings is correct constitutes the goal of the *lexical semantic ambiguity* resolution.

2.2 Structural Ambiguity

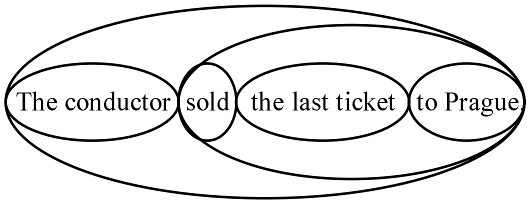
Secondly, the system would to determine the relations between the individual words. Was the ticket in (1) *sold to Prague* or was *it a ticket to Prague* that was sold? The syntactic knowledge defines all possible structural representations, i. e. there are two different corresponding syntactic trees they can be generated in above mentioned cases.

The identification of the relations among the words is based on our knowledge of the concepts they represent.

Without this knowledge, the system would incorrectly take into consideration the inappropriate structure, leaving us with a structural ambiguity of two possible syntactic structures:

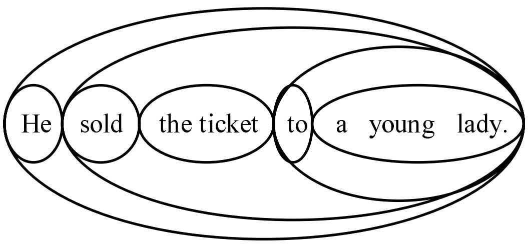


or



2.3 Ambiguity of Reference

Although it seems that the sentence (2) can be quite straightforwardly analyzed into the following structure (the problem of representation is not yet solved):



Every noun phrase is either indefinite or definite. While indefinite noun phrases evoke new discourse entities, the definite noun phrases refer to entities already established in context. If there is more than one entity in the context that the noun phrase can refer to, we face a problem of *referential ambiguity*. The definite noun phrase *the ticket* in (2) refers to the same ticket established in (1). Similarly, the pronoun *he* refers to the concept of *conductor* also established in (1). The situation is not always as simple as in this example, because the number of possible candidates of reference is usually much higher, and/or the decision involves complicated inference with general knowledge.

2.4 Ellipsis

Ellipsis involves the use of sentences that appear ill-formed because they do not form complete sentences. Typically the parts that are missing can be extracted from the previous sentence. An *elliptical* utterance (the input fragment) must correspond in structure to a sub-constituent in the previous sentence (the target fragment). The final syntactic structure of the utterance is constructed by replacing the target fragment in the previous sentence with the input fragment. In this sense, ellipsis resolution is similar to *anaphora* resolution, where we also have to search for appropriate constituents in the previous sentences, and both ellipsis and anaphora resolution are heavily dependent on the context and general knowledge.

2.5 Processing of incomplete and ambiguous sentences

Similarly to the human listener, the computer system needs to apply syntactic, semantic, pragmatic and general knowledge in order to determine the final internal representation of meaning. Unfortunately, this representation cannot be realized solely from the incoming utterance. Rather, it is a result of the utterance itself, the context of the previous utterances, the context in which the communication takes place, and the general knowledge on the listener's, i. e. system's side. Intuitively, we can envisage this representation as a semantic network of concepts that the words represent, connected by the corresponding semantic relations. But, it is only an usual graphical representation to visualize the semantic relations of the sentence. The „real understanding" system have to combine the incomplete semantic information extracted from the analyzed utterance with the previous „data" stored in the dialog history path, (it is provided in the developed dialog system by the special storage module of the *dialog history* - see below), or by using a set of rules (derived from the comprehensive set of training sentences) of an one-purpose rule oriented interpretation knowledge system incorporated into the interpretation module of the dialog manager.

3. IMPLEMENTATION

The processing of sentences of all kinds, including the incomplete sentences, is controlled by the dialog module of the dialog manager (see Fig. 1). The meaning of analyzed sentences is internal represented (and implemented) as a chained hierarchical structure of sentence frames containing at most seven contextual elements. The frames are implemented by using dynamic data structures to obtain the efficient implementation and to save the memory space.

To preserve the dialog history, all analyzed and step by step completed semantic frames are stored to the special temporary storage area of the dialog module. If the last analyzed sentence is classified as an incomplete one, its representing frame is compared with all temporary stored frames of previously analyzed sentences, the frame is „overlapped“ with the most corresponding „elder“ frame and completed for all missing data. If no corresponding frame is found in the temporary storage (in the dialog history), i.e. the analyzed sentence is the anaphoric one, the dialog module hands over the sentence analysis to the interpretation module, which attempts to complete the missing data with the help of a set of especially created and organized domain dependent knowledge bases. If this second analysis phase is unsuccessful too, only then the clarifying question is uttered.

All stored sentence frames representing the saved dialog history are immediately cleared after the ceremony closing the sequence of dialog acts.

4. CONCLUSION

The first version of the „Czech natural speaking“ dialog system was designed and successfully implemented in last 18 months. The functional features of the developed dialog manager involving the program module of the dialog history processing were tested and evaluated by the group of „semi-naive“ users (students and laboratory staff). The results of this evaluation are summarized in the following table:

Number of users	15
Number of dialogs	150
Excellent dialogs	32
Dialogs with troubles *)	61
Unsuccessful dialogs	57
Successful dialogs total	93
%	62.0

*) Dialog with troubles means the dialog with information repeats, corrections, misunderstandings, additions, etc.

Looking back to the designing process we can formulate the following recommendations, how to respect both human linguistic behaviors and speech technology performance and to develop the realiable functioning dialog manager successfully processing user's utterances of all kinds, incl. incomplete and ambiguous sentences:

- carefully formulate the dialog task based on an ergonomic analysis of needs or requirements of potential users;
- analyse an adequate account of all kinds of possible dialogs recorded in real life, making objectives explicit;
- transcribe the dialogs recorded in real life using a standard transcription scheme if possible;
- draw up the rules describing all possible dialog acts;
- design and implement a first version of the dialog module and dialog history storage as well as the knowledge base of the interpretation module involving the general knowledge from the required problem domain;
- carry out various tests with real users, summarize the new knowledge, draw up the new rules and involve these to the knowledge base;
- "tune" the system by iterative modifying, then testing it more;
- repeat the previous two steps until all reasonable requests of real users are satisfied.

The last three design steps of development of the interactive dialog manager have to be performed in close collaboration with a professional organization which have ordered the complete dialog system and takes the system into its services.

REFERENCES

[1] Eckert, W. and Niemann, H. (1994). „Semantic Analysis in a Robust Spoken Dialog System.“ *In: Proceedings of the International Conference on Spoken Language Processing*, Yokohama, Japan, pp. 107-110.

[2] Eckert, W. (1994). „Gesprochener Mensch-Maschine-Dialog.“ Dissertation, Technische Fakultaet der Universitaet Erlangen-Nuernberg, Germany.

[3] Gibbon, D. and Moore, R. and Winski, R. (1997). „Handbook of Standards and Resources for Spoken Language Systems.“ *Mouton de Gruyter Verlag*, Berlin, New York.

[4] Kleckova, J. and Matousek, V. and Netrvalova, J. (1995). „An Automatic Creation of the Language Model for the Spontaneous Czech Speech Recognizer.“ *In: Proceedings of the European Conference EUROSPEECH '95*, Madrid, pp. 1185-1188.

[5] Krutisova, J. and Matousek, V. and Ocelikova, J. (1998). „Development of an One-Purpose Dialog Manager.“ *In: Proc. of the Int. Workshop SPECOM '98*, St. Petersburg, pp. 131-134.

[6] Matousek, V. and Krutisova, J. and Mautner, P. and Ocelikova, J. (1998). „A Dialog Module for the Information Retrieval System.“ *In: Proc. of the AIII Int. Conference on Artificial Intelligence Applications in Industry*, Kosice, pp. 469-478.

[7] Netrvalova, J. (1997). „Representation and Interpretation of Continuous Utterances.“ *In: Proc. of the 2-nd SQEL Workshop on Multi-Lingual Information Retrieval Dialogs*, Plzen, pp. 28-31.

[8] Phillips, M. D. and Bashinski, H. S. and Ammerman, H. L. and Fligg, C. M. (1988). „A Task Analytic Approach to Dialogue Design.“ *In:* Helander, M. (Ed.). „Handbook of Human-Computer Interaction“. *Elsevier Science, North Holland Publ.*, pp. 835-857.

[9] Sacks, H. and Schlegloff, E. and Jefferson, G. (1974). „A simplest systematics for the organization of turn-taking in conversation.“ *In: Language, Vol. 50*, pp. 697-735.

[10] Stetina, J. (1997). „Corpus Based Natural Langu-age Ambiguity Resolution.“ *PhD. Thesis*, CTU Prague.