Dialog management in the Waxholm system

Rolf Carlson, Sheri Hunnicutt and Joakim Gustafsson

Department of Speech Communication and Music Acoustics,
KTH, Stockholm, Sweden

ABSTRACT

In this paper we will describe the natural language and dialog component in the Waxholm system. Our parser, STINA, is knowledge based and is designed as a probabilistic language model. The dialog management, also implemented in STINA, is based on grammar rules and lexical semantic features. The parser is running with two different time scales corresponding to the words in each utterance and to the turns in the dialog. Topic selection is accomplished based on probabilities calculated from user initiatives.

1. INTRODUCTION

Our research group at KTH is currently building a generic system in which speech synthesis and speech recognition can be studied in a man-machine dialog framework. The demonstrator application, which we call WAXHOLM, gives information on boat traffic in the Stockholm archipelago [1, 2]. In addition to boat time-tables, the database also contains information about port locations, hotels, camping places, and restaurants in the Stockholm archipelago. The system is presented in a separate contribution to this workshop. In this paper we will describe some of the features in the natural language and dialog component. We will also discuss subject performance and system performance during the collection of the Waxholm database.

2. THE WAXHOLM DATABASE

We have been collecting speech and text data using the Waxholm system. Initially, a "Wizard of Oz" has been replacing the speech recognition module. About 1900 utterances (9200 words) in this database have been used for the experiments reported in this paper. About 700 utterances are simple answers to system questions while the rest, 1200, can be regarded as user initiatives [3].

The database was collected using preliminary versions of each module in the Waxholm system. This procedure has advantages and disadvantages for the content of the database. System limitations will already from the beginning put constraints on the dialog, making it representative for a human-machine interaction. However, since the system was under development during the data collection, it was influenced by the system status at each recording time. After about half of the recording sessions, the system was reasonably stable, and the number of system "misunderstandings" had been reduced.

3. THE STINA PARSER

Our initial work has been focused on a sublanguage grammar, a grammar limited to a particular subject domain -- that of requesting information from a transportation database. Our parser, STINA, is knowledge based and is designed as a probabilistic language model [4]. STINA contains a context-free grammar which is compiled into an augmented transition network (ATN). Probabilities are assigned to each arc after training.

Semantic features are divided into two classes, basic features and function features. Basic features such as BOAT and PORT give a simple description of the semantic property of a word. These features are hierarchically structured. In Figure 1, we give an example of a basic semantic feature tree.

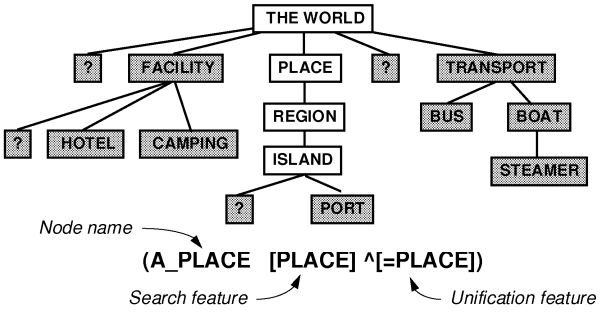

Figure 1. Example of a semantic feature tree.

The second type of semantic features are the function features. These features are not hierarchical. Typically these features are associated with an action. A typical feature is TO_PLACE indicating the destination in an

utterance regarding travel. The function features are also node names in the parser. A verb can have function features set allowing or disallowing a certain type of modifier to be part of a clause. For example, the node DEPARTURE_TIME is disallowed in connection with verbs that imply an arrival time.

This method is also a powerful method to control the analysis of responses to questions from the dialog module. The question "Where do you want to go?" conditions the parser to accept a simple port name or a prepositional phrase including a port name as a possible response from the user. This property of STINA gives the parser some of the advantages of a functional grammar parser.

The rule that Figure 1 depicts uses the feature structure to accept all places, regions, islands and ports. Thus, a unification of the feature PLACE engages all semantic "non-shaded" features in the figure. The whole tree of the lexical entry is moved into the hypothesis including the leaves on the feature tree. A port name will keep its PORT feature even if only the PLACE is noted in the rule. The rules do not have to be more specific than necessary and the domain knowledge can, to some extent, be part of the lexicon rather than the rules.

3.1 Parsing results

The parser has been evaluated in several different ways. Using about 1700 sentences in the Waxholm database as test material, 62 percent give a complete parse, whereas if we restrict the test data to utterances containing user initiatives (about 1200), the result is reduced to 48 percent. This can be explained by the fact that the large number of responses to system questions typically have a very simple syntax. If we exclude extralinguistic sounds such as lip smack, sigh and laughing in the test material based on dialog initiatives by the user, we increase the result to 60 percent complete parses. Sentences with incomplete parses are handled by the robust parsing component and frequently effect the desired system response.

The perplexity on the Waxholm material is about 34 using a trained grammar. If extralinguistic sounds are taken away we get a reduction to about 30. If only utterances with complete parses are considered we get a perplexity of 23.

The parser is relatively fast on our HP 735. It takes about 17 msec to process an utterance. It can be changed to run faster if some of the analysis facilities are taken out; a slightly different approach on constraint evaluation would also make it faster. At the moment, the processing speed of the parser is not an important issue.

4. DIALOG MANAGEMENT

Our objective in the dialog management module is to develop a dialog management module which can handle the type of interaction that can occur in our chosen domain. The system should allow user initiatives, without any specific instructions to the user, complemented by system questions to achieve the user's goal. Based on this aim, two major ideas have been guiding the work. First, the dialog should be described by a grammar. We have chosen to use the same notation and the same software (STINA) to implement the dialog grammar. Second, the dialog should be probabilistic. Topic selection is accomplished based on probabilities calculated from user initiatives. The topic selection based on probabilities in our system has similarities with the effort at AT&T [5].

4.1 Semantic analysis

The semantic analysis is a straightforward process in which the syntactic tree is reduced to a semantic tree, deleting all nodes and branches that contain no semantic information. After this, a special process creates a semantic frame with slots corresponding to attribute-value information taken from the tree. The semantic frame has a feature specification describing which features are used in the frame and which information might have been added to the frame from the dialog history.

4.2 Topic selection

In the following description, we have used the term "topic" to describe what type of information a user is requesting or, in some cases, a special response from the system. The decision about which path to follow in the dialog is based on several factors such as the dialog history and the content of the specific utterance. Each semantic feature found in the syntactic and semantic analysis is considered in the form of a conditional probability to decide on the topic. The probability for each topic is expressed as: p(topic|F), where F is a feature vector including all semantic features used in the utterance. Thus, the BOAT feature can be a strong indication for the TIME_TABLE topic but this can be contradicted by a HOTEL feature.

4.3 Evaluation of topic selection

We have performed a sequence of tests to evaluate the topic selection method. Only utterances indicating a

topic (about 1200) have been included in these tests. The evaluation has been performed using one quarter of the material, about 300 utterances, as test material, and the rest as training material, about 900 utterances. This procedure has been repeated for all quarters and the reported results are the mean values from these four runs. The first result, 12.9% errors in Table I, is based on the unprocessed labeled input transcription. One of the eight possible topics, labeled "no understanding," is trained on a set of constructed utterances that are not possible to understand, even for a human. This topic is then used as a model for the system to give an appropriate "no understanding" system response. It seemed reasonable to exclude the "no understanding" prediction from the result since the system at least does not make an erroneous decision. The accuracy model in word recognition evaluation has the same underlying principle. By excluding 55 utterances, about 5% of the test corpus, predicted to be part of the "no understanding" topic, we reduce the error by about 4%.

In the next experiment, we excluded all extralinguistic sounds, about 700, in the input text. This will increase the number of complete parses with about 10% as discussed earlier. The prediction result was about the same compared to the first experiment.

The final experiment included only those utterances that gave a complete parse in the analysis. The errors were drastically reduced. This means that the utterances with a syntax covered by our grammar also were semantically easier to interpret. On the other hand, we do not yet know if an increased grammatical coverage also will reduce the topic prediction errors.

Table I. Results from the topic prediction experiments.

test material	all material		excluding no understanding	
	%Error	N	%Error	N
woz input	12.9	1209	8.8	1154
no extralinguistic sounds	12.7	1214	8.5	1159
only complete parses	3.1	581	2.9	580

4.4 Dialog rules

Dialog management based on grammar rules and lexical semantic features is implemented in STINA. The STINA parser is running with two different time scales concurrently corresponding to the words in each utterance and to the turns in the dialog. Syntactic nodes and dialog states are processed according to transition networks with probabilities on each arc.

Each predicted dialog topic is explored according to the rules. These rules define which constraints have to be fulfilled and what action should be taken depending on the dialog history. Each dialog node is specified according to node type, node activity, and constraint evaluation. The constraint evaluation is described in terms of features and in terms of the content in the semantic frame. If the frame needs to be expanded with additional information, a system question is synthesized. During recognition of a response to such a question, the grammar is controlled with semantic features in order to allow incomplete sentences. If the response from the subject does not clarify the question, the robust parsing is temporarily disconnected so that specific information can be given to the user about syntactic problems or about unknown word problems. At the same time, a complete sentence is requested giving the dialog manager the possibility of evaluating whether the chosen topic is incorrect.

5. DIALOG ANALYSIS

5.1 Subject performance

A total of 68 subjects participated in the experiment. Each subject was presented with 3 scenarios. A total of 198 scenarios were recorded and analyzed. Each scenario required that the user solved from one to four subtasks. A subtask could be that the subject had to request a timetable, a map or a list of facilities. Each subtask, in turn, required specification of several distinct constraints, such as departure port, destination port and departure day, before the subtask could be solved. The subjects had to provide the system with up to ten such constraints, with a mean of 4.3, in order to solve a complete scenario.

The database contains 265 subtasks and about 84% of these were solved by the subjects. In 75 percent of the cases, 199 out of 265, the subjects had completed a subtask after one to five utterances. The subjects needed about 7 utterances to solve one scenario. After the task was completed several subjects continued to ask questions in order to test the system. About 3 additional utterances were collected this way. In 42 cases a scenario could not be completely solved by a subject, corresponding to an error rate of 21%. In half of these, 21 scenarios, some of the subtasks were solved by the subjects.

The average utterance length was 5.6 words. The average length of the first sentence in each scenario was 8.8

words. The utterance length distribution shows one maximum at two words and one at five words. One reason for this distribution is that many of the utterances were subject answers to system questions. As an example, one type of system question was "Which port would you like to go to/from?". A typical answer to this question was "To/From Stockholm" or "I want to go to/from Stockholm." (The infinitive mark is left out in Swedish).

We can find a few examples of restarts in the database due to hesitations or mistakes on the semantic, grammatical or phonetic level. However, less than 3% of the utterances contain such disfluencies. Some of the restarts are exact repetitions of a word or a phrase. In some cases a preposition, a question word or a content word is changed. We also find repetitions of incorrectly pronounced words. About one fourth of the restarts occur in interrupted words, that is, in words that are not phonetically completed.

5.2 System response analysis

The Waxholm-database contains approximately 1900 dialog turns. After the first 37 sessions, the system went through a major revision. The first phase included approximately 1000 subject utterances. The system responses "I do not understand" and "You have to reformulate" occurred in 35.8 % of the system responses. In the second phase, the dialog manager was updated as well as the scenarios. In this phase, 31 subjects produced 900 utterances. The improved system failed to understand 20.9% of the time, an improvement

of 15%. It should be noted that this system response in some cases also is the correct one.

Most of the questions from the system occurred when the system predicted that the subject wanted a timetable displayed. In these cases, the distinct constraints were evaluated, and if some information was missing, the system took the initiative to ask for this information. The subjects answered the system questions in 95.4% of the cases. Thus, the subjects were quite co-operative and rarely, one percent, used the possibility to change the topic during the system-controlled dialog. In a more realistic environment, using speech recognition as input, the system might misunderstand the user's goal, and topic changes by the subject will become more frequent.

The most serious problems occurred when the system failed to 'understand' an utterance from a subject. The first system response was a simple "I do not understand" utterance. If the system failed to understand once more, the system elaborated more on the problem. First, the subject was informed where it failed to understand, if it was a linguistic problem. Second, the system asked the user to use a complete sentence next time. The following utterance from the subject was used to evaluate whether the system-predicted topic actually agreed with this new utterance or whether the topic should be changed. The system responded 'I don't understand' 575 times corresponding to 268 occasions where consecutive repetitions are counted as one occasion. In 50% of the cases the system recovered after one additional utterance.

6. FINAL REMARKS

The STINA parser handles both the regular grammar analysis and the dialog control in the Waxholm project. We have found this approach to be very profitable since the same notation, semantic feature system and developing tools can be shared. The rule-based and probabilistic approach has made it reasonably easy to implement an experimental dialog management module. We have recently added a graphical interface to the system which presents each network graphically. Both the syntax and the dialog networks can be modeled and edited graphically with this tool.

7. ACKNOWLEDGMENT

This work has been supported by grants from The Swedish National Language Technology Program.

8. REFERENCES

[1] Blomberg, M, Carlson, R, Elenius, K, Granström, B, Gustafson, J, Hunnicutt, S, Lindell, R & Neovius, L (1993). An experimental dialogue system: WAXHOLM, Eurospeech '93, Berlin, 1867-1870

[2] Carlson, R. (1994) Recent developments in the experimental "Waxholm" dialog system, ARPA Human Language Technology Workshop, 8-11 March 1994.

[3] Bertenstam et al. (1995, in press) Spoken dialog data collected in the Waxholm project, STL-QPSR 1, Technical Report, Dept. of Speech Comm., KTH, 1995.

[4] Carlson, R. and Hunnicutt, S. (1995, in press) The natural language component - STINA, STL-QPSR 1, Technical Report, Dept. of Speech Comm., KTH, 1995.

[5] Gorin, A. (1994) Semantic associations, acoustic metrics and adaptive language aquisition, ICSLP, International Conference on Spoken Language Processing, Yokohama, Japan, 79-82.