THE LIMSI ARISE SYSTEM FOR TRAIN TRAVEL INFORMATION*

L. Lamel, S. Rosset, J.L. Gauvain, S. Bennacef
Spoken Language Processing Group
LIMSI-CNRS, 91403 Orsay, France
{lamel,rosset,gauvain,bennacef}@limsi.fr

ABSTRACT

In the context of the LE-3 ARISE project we have been developing a dialog system for vocal access to rail travel information. The system provides schedule information for the main French intercity connections, as well as, simulated fares and reservations, reductions and services. Our goal is to obtain high dialog success rates with a very open dialog structure, where the user is free to ask any question or to provide any information at any point in time. In order to improve performance with such an open dialog strategy, we make use of implicit confirmation using the callers wording (when possible), and change to a more constrained dialog level when the dialog is not going well. In addition to own assessment, the prototype system undergoes periodic user evaluations carried out by the our partners at the French Railways.

INTRODUCTION

The LE-3 ARISE (Automatic Railway Information Systems for Europe) project aims a developing prototype telephone information services for rail travel information in several European countries. In collaboration with the Vecsys company and with the SNCF (the French Railways) we have developed a prototype telephone service providing timetables, simulated fares and reservations, and information on reductions and services for the main French intercity connections. A prototype French/English service for the high speed trains between Paris and London is also under development.

The system providing information for the main intercity connections is largely based on the spoken language systems developed for the LE-MLAP RAILTEL project[1, 6] and the ESPRIT MASK project[3]. The system architecture is modular, and the system runs on a Unix workstation with a telephone interface. Compared to our RAILTEL system, the main advances in ARISE are in dialog management, confidence measures, an the inclusion of optional spell mode for city/station names, and of a barge-in capability to allow a more natural interaction between the user and the machine.

While there are commonly used measures and methodologies for evaluating speech recognizers, the evaluation of spoken language systems is considerably more complicated due to the interactive nature and the human perception of the performance. It is therefore important to assess not only the individual system components, but the overall system performance using objective and subjective measures.

*This work was partially financed by the LE-3 project 4223 ARISE.

SYSTEM OVERVIEW

There are six main modules in the spoken language system. The real-time, speaker independent, continuous speech recognizer[4] transforms the acoustic signal into the most probable word sequence. The recognition vocabulary contains 1800 words, including about 500 station names. Speaker independence is achieved by using acoustic models which have been trained on speech data from a large number of representative speakers, covering a wide variety of accents and voice qualities. The recognizer uses continuous density HMM with Gaussian mixture for acoustic modeling and *n-gram* backoff language models. Context-dependent phone models are used to account for allophonic variation observed in different contextual environments. The *n*-gram statistics are estimated on the transcriptions of spoken queries. Since the amount of language model training data is small, some grammatical classes (such as cities, days, months, etc) are used to provide more robust estimates of the *n*-gram probabilities. A confidence score is associated with each hypothesized word. If this score is below an empirically determined threshold, the hypothesized word is marked as uncertain. These uncertain words can be ignored by the understanding component or used by the dialog manager to start clarification subdialogs. Station names can be optionally spelled so as to improve recognition performance with a large number of cities as this is critical for the task. In our current implementation the speech recognizer output is the best word sequence with a confidence score, however, the recognizer is also able to provide a word lattice.

The text string output by the recognizer is passed to the natural language understanding component. This component first carries out literal understanding of the recognizer output, and then reinterprets the query in the context of the ongoing dialog. In literal understanding, the semantic analyzer applies a caseframe grammar to determine the meaning of the query, and builds an appropriate semantic frame representation[2]. Keywords are used to select an appropriate case structure for the sentence without attempting to carry out a complete syntactic analysis. The major work in developing this component is in defining the concepts that are meaningful for the task and the appropriate keywords.

The concepts needed to carry out the main ARISE ticketing task concern train times, connections, fares and reservations (including reductions and other constraints). Other concepts are used to handle general information available about reductions and services. The concepts have been determined by analysis of queries in the training corpora.

Contextual understanding consists of reinterpretating the utterance in the context of the ongoing dialog, taking into account common sense and task domain knowledge. The semantic frames resulting from the literal understanding are reinterpreted using default value rules, and qualitative values are transformed into quantitative ones. Semantic frames corresponding to the current utterance are then completed using the *dialog history* in order to take into account all the information previously given by the user, as well as the questions posed by the system.

The dialog manager then either prompts the user to fill in missing information or uses the semantic frame to generate an SQL-like request to the database management system. The caller is required to specify four key items before database access: the departure and arrival stations, the date and approximate time of travel. The day and time can be specified exactly (March 14th) or in a relative manner, such as *next Monday, early morning, late tomorrow afternoon*. Interpretative and history management rules are applied prior to generation of the DBMS request. These rules are used to determine if the query contains new information, and if so, if this information is in contradictory with what the system has previously understood. If a contradiction is detected, the dialog manager may choose to keep the original information, replace it with the new information, or enter into a confirmation or clarification subdialog.

The database retrieval component uses a copy of the static SNCF train information (database RIHO). Post-processing rules, which take into account the dialog history and the content of the most recent query, are used to interpret the returned information prior to presentation to the user. The generation component converts a generation semantic frame into a natural language response, which is played to the user. The form of the natural language response depends on the dialog context, and whether or not the same information was already presented to the user. Our aim is to give a direct response to the caller, highlighting the new information (see Figure 1). Careful attention has been paid to construction of sentences that contain the appropriate information and the generation of natural-sounding utterances[1]. Messages are synthesized by concatenation of variable-sized speech units stored in the form of a dictionary[5]. The resulting synthetic speech is rated as very natural by users.

The ability to interrupt the system (a barge-in capability) is often considered to be important for usability. Adding this capacity required modifications to several modules. Firstly, recording and speech recognition must be active at all times, even when the system is synthesizing a response. Software-based echo cancellation, applied to the recorded signal using the known synthesized signal is used to detect if the caller is speaking. If speech is detected, synthesis is stopped. There are also dialog situations in which barge-in is disabled to ensure that the caller hears the entire message.

DIALOG STRATEGY

The main objectives of the dialog strategy are:
1) To never let the user get lost. The user must always be informed of what the system has understood. This is of particular importance as most users will be unfamiliar with talking to a machine.
2) To answer directly to user questions. The system responses should be as accurate as possible and provide immediate feedback of what was understood.
3) To give to the user the opportunity, at each step, to correct the system. This capability is needed to be able to correct for recognition errors, but also the user may correct what s/he said or have a change of mind.
4) To avoid misunderstanding. Even though users are able to correct the system at any moment, we have observed that they tend to not do so. It is therefore important to minimize recognition errors, as users can not be expected to correct the system, our motivation for rejecting unreliable hypotheses.

The dialog has been modified in several ways from our RAILTEL system[1]. The responses given by the system are much shorter, so as to keep the caller in tighter contact with the system, and to make for a more natural dialog. In the RAILTEL system information for up to 3 trains was return. The ARISE system proposes only one train, letting the user ask for another train (the preceding/following one, an earlier/later train, the first/last train). Constraint relaxation is used in retrieving timetable information in order to provide a more cooperative dialogue and response. For example, if no train satisfies the user's request, the system relaxes constraints on the departure time so as to find the closest train before or after the specified time.

Closely related to the changes in dialog managment, the generation component has been modified to produce texts which directly integrate the information given in the user's request. These responses aim to provide only the information that is new and/or of high relevance. This immediate feedback, as illustrated by the dialog in Figure 1 allows the user to know what the system has understood.

A two-level dialog strategy has been implemented, in which a mixed-initiative dialog where the user can provide any information at any point in time is combined with a system-directed dialog in case a problem is detected in obtaining departure and arrival station names and the date of travel. When the second level, or constrained dialog is active, the speech recognizer makes use of a dialog-state dependent language model. Implicit confirmation plays an important role in keeping the user aware of what the system

S: *What information do you want?*
U: I want a train to Paris.
S: *So,* ***you want to go to Paris****, where are you leaving from?*
U: from Bordeaux
S: ***From Bordeaux to Paris****, what day are you traveling?*
U: tomorrow morning
S: ***In the morning****, there is a direct train from Bordeaux to Paris at 10:22 arriving at 13:40, on Sunday* ***21st of September****. Is this ok?*
U: and the next train?
S: ***The next train*** *leaves at 10:42 and arrives at 13:45.*
U: uh what is the price of a first class ticket?
S: ***The price of a first class ticket*** *from Bordeaux to Paris is 549 francs.*
U: Ok, I'd like to reserve on this train.
S: *Do you want a smoking or non-smoking seat?*
U: non smoking, please
S: *You have a reservation in first class,* ***non-smoking****, on the 10:42 train direct from Bordeaux to Paris, arriving at 13:45, Sunday Sept. 21st. Do you want any other information?*
U: no thank you
S: *Goodbye, the SNCF wishes you a nice trip.*

Figure 1: Example dialog except illustrating feedback of what the system has understood (Translated from French). The relevant part of the system response is shown in bold.

S: *You are leaving from Paris, what is your destination?*
U: no (recognition error: user said Dinan)
S: *I'm sorry, I did not understand. Indicate your destination, for example Lille.*
U: Di- (user hesitation results in word fragment)
S: *Give your arrival city and spell it if you want, for example Lille L I L L E* **Constrained subdialog**
U: Dinan D I N A N
S: *From Paris to Dinan, what day are you leaving?*

Figure 2: Dialog excerpt illustrating a constrained subdialog for the arrival city. (Translated from French.)

has understood. A constrained dialog can be initiated by the system if the user does not respond to the system prompt for one of the four basic items (timeout), or in cases where the information received by the system is contradictory with what was previously understood. Such constrained dialogs apply only to the departure and arrival cities, and the travel date. For example, if the system understood a change in the departure or arrival city, one of the following strategies is used depending upon the state of the dialog: the system may choose to ignore the information; it can ask for an explicit confirmation of the new city; or it can ask the user to repeat the information. If the caller changes one of these items during the confirmation request, implicit confirmation is used in the following prompt. The directiveness of the prompt increases if the user does not supply the requested information. An example of a constrainted dialog for the arrival city name is given in Figure 2, where the optional spell mode is used.

PERFORMANCE EVALUATION

Callers are recruited on an ongoing basis to provide data for system development and evaluation. In 1998 we have recorded over 2400 calls, with a total of 37k queries. In addition to these calls, the prototype system has undergone 3 rounds of evaluation carried by the SNCF to assess usability and performance. For the SNCF tests, subjects were recruited by a hostess at a Parisian train station. The subjects were asked to test a new, experimental service, and were given a gift certificate for their participation. Each subject called the system three times, carrying out an open scenario that s/he wrote down prior to each call. Subjects completed a short questionnaire after each call and a longer one after the third one. Despite the differences in recruitment, the general characteristics of these calls (in terms of dialog success, overall call duration, the number of exchanges, vocabulary, types of requests and typical problems) are essentially the same as we observed on subects we recruit via advertisements in local newspapers.

The dialog error in obtaining timetable information was 16% on 58 calls recorded during a two-day period in June.[1] Once a train is selected callers are asked if they would like

[1] The calls from June 3rd are not used here, due to experimental problems, such as the subject speaking with the experimentor, or interference due to simultaneous recording. Results are given for calls from June 4 and 5, 1998.

to reserve a train. Reservations, which require specifying the class of travel, seating preference and reduction, had a failure rate of 11%. A higher error rate (30%) was obtained for diverse questions, due in part to functionality limitations. Since knowing when a dialog has finished is a difficult task, we analyzed how the dialogs ended. 12% of the dialogs ended without a closing formality (ie. the caller hung up) without saying goodbye. Such abrupt endings can occur when a caller got the desired information, or because the user was frustrated.

These results are substantially better than the user trials carried out by the SNCF in Nov97 before most of the modifications presented in this paper were implemented. On 80 calls the timetable information failure rate was 47.5% and the reservation failure rate was 35.7%, and over half of the calls were terminated without a closing formality. The June calls are longer, averaging 15 exchanges (167 seconds), compared to 10 exchanges in November (114 seconds). Although more performant, the two-level dialog has increased the length of the dialog. The overall satisfaction level improved from of 5.9 to 12.7 (out of 20) for the SNCF subjects.

An analysis of the use of barge-in was carried out on the 58 calls of the June98 SNCF test. The callers were aware that they could interrupt the system if they so desired. Users interrupted the system in 72% (42) of the calls, speaking during 122 of 958 system responses (13%). When barge-in was observed during a call, it was used on average to interrupt 3 system responses. Barge-in was observed in a variety of contexts, but was most often used to respond to questions before they were finished. For example, when the system is uncertain about a station name, the caller is prompted to say and optionally spell the city name. (*Give you departure city and spell it if you like. For example, Paris, P A R I S.*) Almost 40% of the interruptions followed this type of prompt. In almost 25% of the cases, barge-in seemed to be inadvertant. The caller was seeminly engrossed in their thoughts, talking to the system and unaware that the system was responding. In contrast to our expectations, barge-in was only rarely used (6% of the cases) to correct the system, and usually to change the date of travel.

DISCUSSION

Enabling efficient, yet user-friendly interaction for access to stored information by telephone is quite difficult. Most existing services are directive, restricting what the caller can ask at any given point in the dialog, and limiting the form of the request. Some laboratory prototypes allow a more open, user-initiated dialog, but performance is generally lower than what can be obtained with more restricted dialog stuctures.

Our goal is to obtain high dialog success rates with a very open structure, where the user is free to ask any question or to provide any information at any point in time. In order to improve performance with such an open dialog strategy, immediate feedback is given to let the caller know what the system has understood. This implicit confirmation makes use of the callers wording when possible. Explicit confirmation may be used when the system is uncertain or has understood contradictory information. The scores associated with each hypothesized word enable the understanding modules to ignore uncertain items, that could be misrecognitions. Although such rejection may lead to a longer dialog, since some correct words are ignored, the overall dialog success rate is improved.

Our preliminary observations of the barge-in capability, judged to be very important for usability, indicate that it is not heavily used, and is not used in the manner we had anticipated (to correct misrecognized items). This may be partially due to the experimental conditions, as callers do not really need the information they are asking for, and therefore may not notice (or care about) the errors.

An important issue that was highlighted during the SNCF user trials is that users do not distinguish the functionalities of the service from the system responses. Although the system was able to detect some out-of-functionality requests, and responded that it was unable to handle these, such responses are not satisfactory for users. For example, if the user tries to reserve for several people, system informs him/her that it is unable to reserve for more than one person at a time. While this is logical and correct from the spoken language system developer's point of view, the caller may not be satisfied with this response. So although we may have a successful dialog, we have an unhappy caller.

The results of our assessment indicate that the overall performance has been improved both in terms of success rate and the average dialog duration. The subjective assessment of the subjects has also improved, with most subjects expressing interest in using such a service.

ACKNOWLEDGMENTS

We thank the SNCF for providing the information database RIHO for use in the ARISE project and for their contribution to assessment by carrying out the user trials. We also thank the Vecsys company for their contributions to the generation and synthesis modules.

REFERENCES

[1] S.K. Bennacef et al., "Dialog in the RAILTEL Telephone-Based System," *ICSLP'96*, Philadelphia, Oct. 1996.

[2] S.K. Bennacef et al., " A Spoken Language System For Information Retrieval," *ICSLP'94*, Yokohama, Oct. 1994.

[3] J.L. Gauvain et al., "Spoken Language component of the MASK Kiosk" in K. Varghese, S. Pfleger(Eds.) "Human Comfort and security of information systems", Springer-Verlag, 1997.

[4] J.L. Gauvain et al., "Speaker-Independent Continuous Speech Dictation," *Speech Communication*, **15**, pp. 21-37, Sept. 1994.

[5] L.F. Lamel et al., "Generation and Synthesis of Broadcast Messages," *Proc. ESCA-NATO Workshop on Applications of Speech Technology*, Lautrach, Germany, Sept. 1993.

[6] L. Lamel et al., "The LIMSI RailTel System: Field trials of a Telephone Service for Rail Travel Information," *Speech Communication* **23**, pp. 67-82, Oct. 1997.