CONSIDERATIONS IN THE DESIGN AND EVALUATION OF SPOKEN LANGUAGE DIALOG SYSTEMS

Lori Lamel, Sophie Rosset and Jean-Luc Gauvain
Spoken Language Processing Group, LIMSI-CNRS,
B.P. 133, 91403 Orsay cedex, France
lamel,rosset,gauvain@limsi.fr
http://www.limsi.fr/tlp

ABSTRACT

In this paper we summarize our experience at LIMSI in the design, development and evaluation of spoken language dialog systems for information retrieval tasks. This work has been for the most part carried out in the context of several European and international projects. Evaluation plays an integral role in the development of spoken language dialog systems. While there are commonly used measures and methodologies for evaluating speech recognizers, the evaluation of spoken dialog systems is considerably more complicated due to the interactive nature and the human perception of performance. It is therefore important to assess not only the individual system components, but the overall system performance using objective and subjective measures.

1. INTRODUCTION

In our view, spoken language systems should provide a natural, user-friendly interface with the computer, allowing easy access to the stored information. At LIMSI we have experience in developing several spoken language dialog systems for information retrieval tasks [3, 9, 13, 1]. Our activities in this area have been mainly in the context of European projects and a French language action launched by the AUPELF-UREF [4]. The ESPRIT Multimodal Multimedia Service Kiosk (MASK) project developed and tested an innovative, user-friendly prototype information kiosk combining tactile and vocal input [9, 12]. The same basic technology was used to develop and evaluate to a prototype telephone service in the LE-MLAP RAILTEL (Railway Telephone Information Service) project [11] and its follow up project LE-3 ARISE (Automatic Railway Information Systems for Europe) [13]. Both the MASK kiosk and our *Arise* system provide access to rail travel information such as timetables, tickets and reservations, services offered, and fare-related restrictions and supplements. The Vecsys company, a partner in the ARISE project, is currently developing an industrial prototype of a telephone service for the French railways based on the project results. In the context of the AUPELF-UREF action B2, different dialog strategies are being explored for tourist information services (PARIS-SITI) [6]. This action aims to correlate high level measures with low level criteria, so as to deduce a performance measure which can predict user satisfaction from objective measures [4]. At the European level, LIMSI participated in the ESPRIT DISC/DISC2 LTR Concerted Actions which aimed to codify current best practice in spoken dialog systems development and evaluation.

This paper overviews our experience in designing and evaluating SLDSs for information retrieval tasks, and how evaluation is part of the development process.

2. SPOKEN LANGUAGE DIALOG SYSTEMS

We have developed applications in two classes of SLDSs, telephone-based and kiosk-based. Telephone based services are a natural area for spoken dialog systems as the only means of interaction with the machine are via voice and have thus been the focus of many development efforts. Since all interaction with the caller is by speech, dialog design and response generation are very important aspects of the system. Careful consideration must be given to the content and formulation of clear and concise system responses [17, 13]. Information kiosks and multimedia web interfaces are spreading in availability, providing different ranges of services, such as automated ticketing, orientation information, and general tourist services. Audio output (both sound and speech) can be used to direct the users attention or to provide information. Although for most multimedia interfaces the input modalities are limited to a touch screen and a keyboard, there is increasing interest in using speech as an alternative input modality. MASK [9] and PARIS-SITI are kiosk-based systems [4].

Although these applications share many commonalities, there are important differences, primarily concerning dialog strategies and signal capture. By necessity, dialog plays a much more important role in telephone-based services, where in general multiple caller-system turns are required to obtain a satisfactory response. For example, it is preferable to ask the caller to provide additional constraints to limit the possible solutions, then to simply read off a long list of possible solutions satisfying a request. With a multimedia interface it can be more efficient to display all possibilities on the screen, letting the user select amongst them. Concerning signal capture, the telephone signal has reduced bandwidth, and may be affected by varying channel distortions and handset characteristics. For multimedia interfaces a wide-band signal is available, but the microphone is generally far from the talker's mouth, and in order to account for different heights and positions of the expected user population, it may be desirable to use multiple microphones [8]. One obvious solution is to use a handset to control the microphone position, however this has the disadvantage of reducing the user's freedom to use other input modalities. The background acoustic conditions are expected to be noisier for multimedia interfaces, often located in a public place. Mobile telephone speech, particularly in the car or street also pose challenges for current technology. Integrating speech input with other modalities must also be considered.

For both types of applications the capability of the user to interrupt the machine is often considered as crucial for usability. (There may of course be dialog contexts where it is desirable to disable barge-in to ensure that the caller hears the entire message.) For the telephone applications echo cancellation is needed

to remove the echo of the known synthetic speech in order to be able to detect when the caller talks and to recognize the what is said. Evidently barge-in which is based on the recognizer output, and not just speech detection is more efficient and less prone to errors. Simple energy based techniques can be triggered by spurious noises, which can be generated by the user (coughing, throat clearing, touching the microphone) or externally (tapping, door slam, paper rustling). Barge-in with multimedia interfaces requires acoustic echo cancellation, which is a difficult task as the user is generally in the acoustic field and any movement changes the filter characteristics.

3. CORE TECHNOLOGIES FOR SLDS

The main components of a spoken language dialog system are a speech recognizer, a natural language analyzer, and a dialog manager, which controls the information retrieval component including database access and response generation, and a speech synthesizer. Some of the design issues in developing a speech recognizer for an SLDS are discussed in [2, 8, 10]. In general a user can be expected to interact only briefly with the machine, so there is very little data available for model adaptation. Since sufficient training corpora are rarely available, it is often necessary to collect application-specific data, which is needed for accurate modeling at different levels (acoustic, lexical, syntactic and semantic). This data collection represents a significant portion of the SLDS development effort, and finding approaches to minimize this work remains a research challenge.

The nature of interactive dialog imposes several constraints on the system. The primary requirements are for real-time, speaker-independent, continuous recognition of spontaneous speech, where what is really meant by real-time speech recognition is that there is a very short delay from when the user finishes speaking and the system responds. An acceptable latency is on the order of 0.5s. Since speech recognition is being carried out during signal acquisition, dynamic approaches to signal and parameter normalization are required. Most real-time decoders use a frame synchronous search, combined with dynamic pruning, based either on the number of active solutions and/or on the elapsed time. The recognition vocabulary usually has no more than a few thousand words and it is common practice to explicitly model filler words and breath noises, and to use compound words for common word sequences that are subject to strong reduction and coarticulation. Word-class based language models are often used to give a priori information to the speech recognizer and reduce the risk of model inaccuracy due to non-representivity of the training data. The use of dialog context specific language models is another way of adding task-specific knowledge in the recognizer [7].

Different approaches have been taken to interface the speech recognizer with the natural language understanding (NLU) component which extracts the meaning of the spoken query. In most systems a bottom up approach is taken, where the output of the recognizer (is passed to the NLU component. Most understanding components are based on rules, however some stochastically based systems have been reported [16, 14]. The attraction of statistical methods stems from their success in speech recognition, with human intervention being limited to labeling (or correcting labels). Known disadvantages are that stochastic models require large training corpora in order to reliably estimate model parameters. Also, generalizations that can be made relatively easily by humans may not be automatically learned.

The dialog manager is the controller of the entire system as it manages contextual understanding, the dialog history, information retrieval and response generation. The generation component outputs a natural language response based on the dialog state, the caller's query, and the information returned after database access. As more natural SLDSs are developed it is becoming apparent that the dialog manager is a crucial aspect of the system, and design decisions and functionality influence all other system components [17]. Some considerations concern strategies for error detection and correction, and conflict resolution. For example, a confidence measure can be associated with each word in the output, and uncertain words can be rejected by the recognizer or the understanding component, or used by the dialog manager to start a clarification subdialog. Rejection has strong implications for the interaction with the user (there is a risk of annoying the user by asking for a repetition) and on average leads to longer dialogs. However, this may be preferable to making an error, and may be more successful in the long run. Constraint relaxation can be used to provide a more cooperative dialogue and response, when the system is unable to satisfy the user's request. For example, in the case of a train timetable information task, if no train satisfies the user's request, the system can relax constraints on the departure/arrival time in order to find the closest train. In this case it is crucial that the system response is justified by informing the user that the proposed train is the closest match to their request. If not, the user may assume that the system has made a mistake. This example illustrates the close link between dialog management and response generation.

4. DIALOG DESIGN

The dialog manager is the core of a spoken language dialog system, being the window through which users observe the behavior of the system. For many applications it cannot be assumed that the user will be familiar with the system or with speaking to computers. Our experience with MASK and ARISE has led us to propose the following dialog principles:
1) To never let the user get lost. The user must always be informed of what the system has understood. This is particularly important when users are not familiar with speaking 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 may have a change of mind.
4) To avoid misunderstanding. Even if users are able to correct the system at any time, they tend to not do so. It is therefore important to minimize recognition errors, as users can not be expected to correct the system. This suggests using confidence measures and rejection of unreliable hypotheses.

To support a user-friendly, mixed initiative dialog, the system should support negotiation, navigation (that is detection of topic or task changes), and to the extent possible, be able to detect and deal with errors. When the dialog is going well, the user should be able to express him/herself freely, providing information in any order. If the dialog is not progressing, the system should guide the user. Long dialogs indicate that the user is experiencing problem, therefore we try to minimize the number of dialog turns, to rapidly aide the user to obtain their desired information. To support different user needs, a two-level dialog strategy has been implemented, in which a mixed-initiative dialog is combined with a system-directed dialog in case a problem is detected in obtaining important information. When the second level, or constrained dialog is active, the speech recognizer makes use of a dialog-context dependent language model.

Different dialog phases can be identified: acquisition, negotiation, navigation, post-acceptance and metacommunication. During the acquisition phase, the system obtains the information needed to complete the current task. The negotiation phase occurs when the user modifies his/her request as a function of the information returned by the system. Negotiation is particularly useful when there is no database entry satisfying the constraints specified by the user. Navigation refers to when the user changes from one task to another. Navigation also includes the possibility to ask about the functionality of the system and what types of information are available. Once the user accepts the solution proposed by the system, the dialog enters into the post-acceptance phase. In this phase, if the user does not spontaneously ask another question, the system will suggest another task. The new task depends on the current one. If the user does not enter into any of the proposed tasks, the system closes the dialog. Metacommunication concerns the overall dialog flow as well as the detection and treatment of errors. Error detection is of particular importance in an unconstrained dialog, as few constraints can be applied to minimize recognition and literal understanding errors. Either the user or the system can have the impression that something went wrong and needs to be corrected. The system detects a potential error when contradictory information is obtained. The system can choose to ignore the new information, replace the old information with the new information, or enter into a confirmation dialog. A correction initiated by the user can result from a system error (usually a recognition or understanding error) or can reflect a change of mind. Detecting a user correction can be difficult given the openness of the system, as users may not state their correction clearly. In addition, handling of errors is complicated by the fact that a new error can occur while in the process of correcting an earlier one.

5. SLDS EVALUATION

Evaluation plays an integral role in system development, particularly when combined with iterative system development. The collected data can be used first for evaluation purposes, and then as additional training data for subsequent systems. An analysis of the main errors provides input for immediate corrections or longer term modifications. Different types of evaluation can be used, each with their particular strengths and costs. It is advantageous if the evaluation can be carried out automatically, after labeling of the test data. While this type of evaluation can be applied to individual system components, particularly the speech recognizer and the semantic understanding component, it is not currently possible to automatically evaluate the dialog component. A multilevel error analysis can be used to distinguish between errors due to a particular component and those propagating from preceding stages[11]. When experimenting with new user interfaces and dialog strategies, it is often useful to carry out an informal assessment of system performance and capabilities and how these are perceived by users. For example, informal trials were carried out to test the MASK interface. The resulting interface has a self-presentation illustrating the use of the kiosk and explaining the different types of transactions available; an intuitive interface with easy switching between tasks (such as information or ticketing); a facial image of a clerk to let the user know what the system is doing; and a two-level help facility with fixed time-outs. Our experience with this system is that the user interface and the manner in which novice users are able to familiarize themselves with it, has a large influence on the overall transaction success rate [12]. The periodic evaluations of the kiosk highlighted the need for such help facilities.

System: Friday 10/07 leaving Paris at 7am, arriving in Surgères at 10:09 with a change in Poitiers. Do you want to reserve?
Reco: yes
Frame after interpretation: <rep-pos> **OK**
flag-res: 1. **OK**
...
System: Do you have a reduction?
Reco: fw What reductions are available?
Frame after interpretation: <info-reduction> **OK**
flag-concept: info-reduction. **OK**
flag-res: 1. **NOK (C) response OK**
flag-reduc: 1. **OK**
...
System: The possible reductions are: family, child, youth, retiree, couple,... Do you want more information ?
Reco: yes the price with a family reduction
Frame after interpretation: <rep-pos> **OK**
flag-tarif: 1. **OK**
...

Figure 1: Example annotation for task changes.

An important concern is obtaining realistic user trials. These are obviously needed to properly evaluate the prototype or potential service, but can result in negative publicity if carried out prematurely. Often however the system developer does not often have access to the final user, and must resort to recruiting subjects on ongoing basis to provide data for system development and evaluation. In the context of the MASK and ARISE projects, periodic evaluations were carried out by the SNCF [12, 13] (in conjunction with UCL for MASK). The subjects were recruited by a hostess at a Parisian train station or via a polling company for the final ARISE test campaign. Each subject carried out 3 to 5 scenarios, and completed a short questionnaire after each one, which included an estimatation of the completion time. In addition to these evaluations, we carried out in-house evaluations to assess intermediary systems [12, 13].

While there are commonly used measures and methodologies for evaluating speech recognizers, the evaluation of spoken dialog 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. In addition to the commonly used speech recognition word error rate measure, it can be enlightening to measure the error on words that are important for the task. The frame and slot error rates are often used to evaluate the NLU component, and errors arising from ASR and NLU can be measured. The goodness of the dialog is usually assessed manually, based on the system responses. Global evaluation measures concern the entire user interaction, and include both objective and subjective measures, as well as external observations. Some objective measures are the dialog success rate, the average/maximum/minimum number of turns, the total/waiting time, the number of repetitions/corrections/interruptions, and whether or not there was a closing dialog. Subjective user assessment usually addresses qualitative criteria such as the ease of use, perceived speed, and perceived reliability via a questionnaire [11, 12, 13]. In the case of multimodal systems, the effectiveness of speech can be compared with other modalities, such as touch screen or keypad for input and a visual display for output.

In the following we give a few measures that we use for dialog-level evaluation. Figure 1 shows an example of how annotions are added to the completed semantic frame for a portion of a dialog.

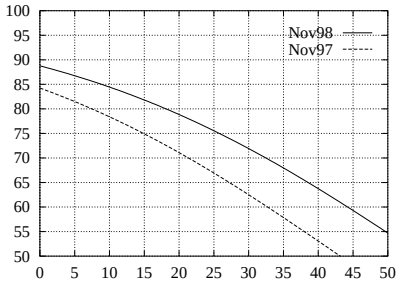

Figure 2: Dialog success as a function of word error rate. Nov97 no error recovery 80 calls), Nov98 with error recovery (189 calls).

These annotations are used to assess the system's capacity to detect task changes [17], by extracting the **OK** and **NOK** marks added to the task change flags. In this example, the **(C)** denotes that the error is due to comprehension.

Figure 2 illustrates the effect of incorporating error recovery mechanisms in the dialog. The curves show the dialog success rate as a function of word error rate for two test sets. For word error rates under 15%, the dialog success rate has been improved by about 5%. For higher word error rates, a larger improvement is seen, at the cost of longer dialogs. Part of the difference is due to the rejection of low scoring words, which tends to lead to longer but more successful dialogs. The dialog success rates are determined by looking at the system responses. Knowing both the correct transcription of the spoken query, the recognizer hypothesis and the semantic frame, we can determine the error source. The dialog and subdialog errors are the ratio of incorrect responses to the total number of system responses to the (sub)dialog. Since knowing when a dialog has finished is a difficult task, we also analyze how the dialogs end. A dialog which ends without a closing formality (ie. the caller hung up early) can occur when a caller got the desired information, or because the user was frustrated.

The ability to interrupt the system (a barge-in capability) is often considered to be important for usability. We analyzed the use of barge-in in our ARISE system. Barge-in was used less often than anticipated in a variety of contexts, in 40% of the cases responding to questions before they were finished. 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.

6. DISCUSSION & PERSPECTIVES

In order to enable efficient, user-friendly interaction with a machine, it is necessary to be able to recognize naturally spoken spontaneous utterances, usually produced while the message is being composed. Spontaneous speech is known to have variations in speaking rate, speech disfluencies and "incorrect" syntactic structures. The SLDS must be able to deal with both the structures of spontaneous speech and recognition errors. Close communication is required for dialog success, thus it is essential that the user be aware of what the system has or has not understood, and be aware of the system functionalities.

Developing and evaluating spoken language dialog systems is complicated by the interactive nature and the human perception of the performance. It is important to assess both the underlying technology and the overall performance, using objective and subjective measures. These are time-consuming processes as subjects must be recruited and much of the analysis must be carried out manually. An unresolved problem is comparing the performance of different systems and dialog strategies, and predicting performance prior to implementation of a new strategy [4, 5, 18]. From the user's viewpoint, the global evaluation measures and subjective opinons are more important than word error and query understanding rates, however these rates do influence the user's perception of the system.

7. ACKNOWLEDGEMENTS

We thank colleagues at LIMSI, the SNCF and at the Vecsys company for their contributions to this work. Our work in spoken language system development and evaluation has benefited from partial support from the following projects: ESPRIT MASK, Language Engineering MLAP RAILTEL and LE-3 ARISE, ESPRIT-LTR Concerted Action DISC, TIDE HOME-AOM, AUPELF-UREF ARC B2.

REFERENCES

[1] J. Shao et al., "An Open System Architecture for Multimedia and Multimodal User Interface," *3rd TIDE Congress*, Helsinki, June, 1998.

[2] N.O. Bernsen, L. Dybkjaer, U. Heid, "Current Practice in the Development and Evaluation of Spoken Language Dialogue Systems," *EuroSpeech'99*, pp. 1147-1150.

[3] H. Bonneau-Maynard et al., "A French Version of the MIT-ATIS System: Portability Issues," *Eurospeech'93*.

[4] H. Bonneau-Maynard, L. Devillers, "A Framework for Evaluating Contextual Understanding," these proceedings.

[5] M. Danielli, E. Gerbino, "Metrics for evaluating strategies in a spoken language system," *AAAI Spring Symposium on Empirical Methods in Discourse Interpretation & Generation*, Stanford, pp. 34-39, 1995.

[6] L. Devillers, H. Bonneau-Maynard , "Evaluation of Dialog Strategies for a Tourist Information Retrieval System," *ICSLP'98*, pp. 1187-1190.

[7] E.W. Drenth, B. Rüber, "Context-dependent probability adaptation in speech understanding," *Computer Speech & Language*, **11**(3), pp. 225-252, July 1997.

[8] J.L. Gauvain, J.J. Gangolf, L. Lamel, "Speech Recognition for an Information Kiosk," *ICSLP'96*, pp. 849–852.

[9] J.L. Gauvain et al., "Spoken Language component of the MASK Kiosk" in *Human Comfort & Security of Information Systems*, K.Varghese, S.Pfleger (Eds.), Springer-Verlag, 1997.

[10] J.R. Glass, T.J. Hazen, I.L. Hetherington, "Real-time Telephone-based Speech Recognition in the Jupiter Domain," *ICASSP-99*, **1**, pp. 61-64.

[11] 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.

[12] L. Lamel et al., "User Evaluation of the MASK Kiosk," *ICSLP'98*, pp. 2875-2878.

[13] L. Lamel et al., "The LIMSI ARISE System," *Speech Communication*, **31**(4), pp. 339-353, Aug. 2000.

[14] E. Levin, R. Pieraccini, "CHRONUS, The Next Generation," *ARPA Spoken Language Systems Technology Workshop*, Austin, pp. 269-272, Jan. 1995.

[15] A. Life et al., "Data Collection for the MASK Kiosk: WOz vs Prototype System," *ICSLP'96*, pp. 1672–1675.

[16] R. Schwartz et al., "Language Understanding using Hidden Understanding Models," *ICSLP'96*, pp. 997-1000.

[17] S. Rosset, S.K. Bennacef, L.F. Lamel, "Design Strategies for Spoken Language Dialog Systems," *Eurospeech'99*.

[18] M. Walker et al, "PARADISE: A general framework for evaluating spoken dialog agents," *EACL'97*, pp. 271-280.