DESIGN AND EVALUATION OF SPOKEN DIALOG SYSTEMS

C. A. Kamm & M. A. Walker
Speech & Image Processing Services Research Lab
AT&T Labs - Research
Florham Park, NJ

Abstract -

Interactive spoken dialog systems extend the range of automated telecommunication services beyond simple limited-choice form-filling applications to goal-directed tasks covering richer, more complex domains. Creating effective and efficient dialog systems requires not only accurate and robust speech recognition and language modeling, but also iterative, principled design of the user interface (UI), coupled with system evaluation with real users. This paper focusses on the design and evaluation of spoken dialog systems, first discussing how general UI principles that are well-understood for graphical user interfaces are also relevant for spoken language interfaces, and then describing the PARADISE framework[14] for evaluating spoken dialog systems.

INTRODUCTION

Spoken dialog systems that enable effective and natural interfaces to automated services, including telecommunication control, information access, and business transactions, are gaining increasing presence, at least in prototype and trial environments [1, 4, 13, 7]. The success of such systems depends on more than the availability of high performance speech recognition, adequate language modelling, appropriate semantic representations for efficiently interacting with back-end knowledge sources, and their integration into a complete real-time system. Significant effort must also be expended in the design of the user interface for these services, in part because of the fragility of the component technologies, and also because of the wide variability in expectations and expertise of users of these systems. The creation of a successful user interface is typically an iterative process, with cycles of design, implementation, experimentation with users, and evaluation, followed by redesign and implementation of improvements, based on the results of the user tests. A critical aspect of this iterative process is how to evaluate the performance of a spoken dialog system. However, the field of spoken dialog systems has been hampered by lack of a consistent overarching framework for evaluating dialog systems. A wide variety of measures has been used, including global measures like task success as well as more local measures based on a single turn

0-7803-3698-4/97/$10.00 © 1997 IEEE.

or interchange in the dialog [2, 3, 8, 10]. To address this lack of consistency, we recently proposed PARADISE (PARAdigm for DIalogue System Evaluation), a general integrative framework for evaluating spoken dialog agents[14]. Thus, the purpose of this paper is twofold: first, to describe user interface (UI) principles that have driven our design of anthropomorphic voice-enabled personal agents for telephony control and information retrieval [4], and second, to present an overview of the PARADISE framework for evaluating spoken dialog agents.

USER INTERFACE DESIGN PRINCIPLES

Some of the basic design principles that are well-understood in graphical user interfaces (GUI) are equally applicable to spoken language interfaces (SLI) [5]. Schneiderman [9] identified three key design principles for GUIs:

- *continuous representation* of the objects and actions of interest; that is, keeping the graphical objects present on the screen so that it is both obvious and intuitive to the user what can be done next.
- *rapid, incremental, reversible operations* whose impact on the object of interest is *immediately* visible, ensuring that every action has an immediate and unmistakable response that can be reversed if desired.
- physical actions or labeled button presses instead of complex syntax using natural language text commands.

These principles reflect, to some extent, an understanding of the limitations of human cognitive processing [15]. Humans have limited working memory capacity, and so cannot retain large amounts of information simultaneously. In GUIs, continuous representation addresses the user's difficulty remembering the available options, and having simple physical actions (e.g., mouse clicks) associated with labeled button presses alleviates the user from having to remember details of command syntax. Similarly, rapid operations with immediately visible impact help maintain the user's attention on the task, facilitating task completion. Spoken-language interfaces must address the same human cognitive limitations, and the transient, serial nature of audio signals imposes additional requirements for SLIs. Without a visual display, different strategies are required to instantiate design principles analogous to continuous representation and immediate impact.

Simulating a continuous representation of all available options at any point in a dialog maybe be virtually impossible (e.g., in dialog systems where the set of possible responses is very large) or highly undesirable (because the time required to list the options might be unacceptably long). An alternate strategy for making the system's capabilities easily apparent to the user is providing prompts in a "question - pause - options" format, where the "options" serve as a reminder of what the user can say at any point in the dialog. Figure 1 shows an example of this strategy, in the context of a personal agent

Agent:	Annie here, what can I do for you?
User:	(says nothing within two-seconds)
Agent:	You can say "call", followed by a name or number, "Get my messages", or "Get me the employee directory". For more options, say "Help me out".

Figure 1: Continuous Representation

for telephony control. In this example, continuous representation is provided when the system agent detects that the dialog is not progressing as expected. That is, when the user does not respond to the system within two seconds, the assumption is that the user may not know what to do, and the agent explains how to perform the most frequently used functions, as well as how to find out about other functions. A second way to provide a sense of continuous representation in audio-only interfaces is by providing consistency across features in a multi-featured application. This consistency can be achieved using a subset of vocabulary that is always available and that serves the same function regardless of where the user is in the feature space or the dialog. For example, the phrase "help me out" provides context sensitive help throughout the personal agent application, and the phrase "cancel" can be used at anytime to undo the preceding action, instantiating the principle of reversibility.

The principle of immediate, observable impact is often provided in spoken dialog systems as implicit confirmation, as show in Figure 2. Here, from the agent's response, the user is informed that a) his/her request was understood and b) the agent is acting to fulfill the request. In order to achieve the design principle of immediacy in spoken dialog interfaces, the system must have real-time, low latency response, and must also allow the user to "barge-in" while the system is speaking. Effective audio interfaces listen even while they are presenting prompts and stop speaking when they are interrupted by the user, following human conversational etiquette. This principle provides a sense of continuity and natural flow to the dialog. The principle of incrementality

Agent:	Annie here, what can I do for you?
User:	Call Lyn Walker.
Agent:	Calling Lyn Walker.

Figure 2: Immediate Impact

is shown in Figure 3, from a dialog in an Air Travel Information System application [7]. Here, the user provides the originating airport, but has not given sufficient information for the agent to disambiguate the destination, so the agent asks a clarifying question to determine the destination.

User:	I want to go from Boston to San Francisco.
Agent:	San Francisco is served by two airports, San Francisco International and Oakland International. Which airport do you want?
User:	San Francisco International.
Agent:	When do you want to travel from Boston to San Francisco International Airport?

Figure 3: Incrementality

An additional constraint imposed by auditory memory limitations results in the need for summarization and aggregration in audio-only interfaces. Information that is relatively easy to represent effectively with a persistent visual display, e.g., in tabular form, must be condensed for audio-only interfaces, where the options for scanning and browsing information are limited or cumbersome. An example of the use of summarization and aggregation from a system that provides voice access to electronic mail is shown in Figure 4.

Agent:	You have 20 new e-mail messages, with important messages from President Clinton and Vice President Gore.
User:	Do I have any messages about meetings today?
Agent:	You have 3 messages about meetings for today: one from Gore about "NII meeting at 3 pm", one from Shalala about "Budget meeting" and one from your secretary about "Staff Meeting at 8 a.m."
User:	Please read the messages about meetings.

Figure 4: Summarization and Aggregation

While speech-enabled interfaces have no direct analog to Schneiderman's third design principle (i.e., labelled button presses in lieu of complex syntax), the advent of natural language spoken understanding systems where the user is not constrained to remember specific vocabulary or grammar is consistent with the underlying motivation for this design principle.

Thus, effective UI design, be it for graphical, spoken-language or multimodal interfaces, applies general principles that reflect a consideration of the capabilities and limitations of human memory and cognitive and sensory processing.

EVALUATION OF DIALOG SYSTEMS

Even principled UI design cannot guarantee a successful or optimal spoken-language system. Spoken dialog systems are comprised of many interrelated technologies and components, and the success of a dialog agent, from the

user's viewpoint, depends on both whether the agent helps the user accomplish the task at hand and how the task is accomplished. A critical step in the development of spoken dialog systems is not only designing the interface and dialog strategies, but also evaluating how well the system performs. The lack of a general framework for evaluating and comparing spoken dialog systems has resulted in the use of a variety of measures of system "performance" (e.g., inappropriate utterance ratio, turn correction ratio, concept accuracy, implicit recovery, number of turns, elapsed time and others [2, 3, 8, 10]), which in turn have made it difficult to compare the utility of different dialog strategies across tasks and systems.

PARADISE [14] (PARAdigm for DIalog System Evaluation) is a general framework for evaluating spoken dialog agents that integrates and enhances previous work in this area. To support the comparison of different strategies for the same tasks, the framework separates the representation of the application task from the strategies that the dialog agent uses to achieve the task goals. In order to determine the tradeoffs among various factors that contribute to performance, it defines performance quantitatively, as a weighted function of a task-based success measure and behavior-based cost measures, and provides a methodology for determining what the weights should be. This section provides a cursory overview of the PARADISE framework. For a more detailed description of PARADISE, see [14]; for an application of PARADISE to comparative evaluation of a voice-enabled e-mail reading application, see [13].

PARADISE uses a decision theoretic framework [6] to specify a model in which the relative contributions of various disparate factors to a dialog agent's overall performance can be determined. Figure 5 shows a block diagram of the structure of the model. In this framework, the primary objective is to

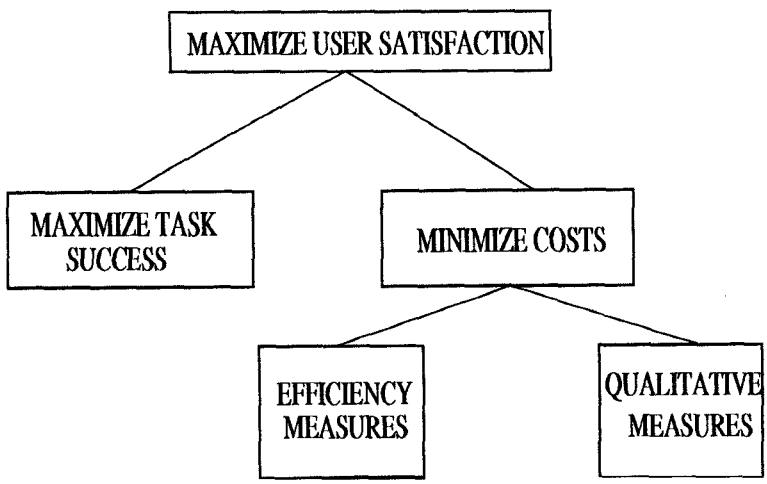

Figure 5: Structure of Objectives for Spoken Dialog Design Problem

maximize an objective related to the usability of the system, which can be measured directly as user satisfaction [10, 8]. The use of user satisfaction as an external validation criterion for system performance is based on the assumption that user satisfaction is predictive of other objectives (e.g., willingness to use or pay for a service) that may not be as easily measured. The model further posits that two factors - task success and dialog costs [12] - are potential contributors to user satisfaction. Dialog costs can be further broken down into two types of factors, objective measures of dialog efficiency and qualitative measures related to subjective perceived costs. In order to estimate a performance function, the model uses linear regression to identify the relative contributions of the success and cost factors for predicting user satisfaction.

The task representation in the PARADISE framework decouples *what* the spoken dialog agent and user accomplish from *how* the task is accomplished using dialog strategies. An attribute value matrix (AVM) is used to represent what is to be achieved - that is, what information elements must be learned by the agent and what information must be conveyed to the user in order to complete the task. For example, in a voice e-mail agent scenario where the user is asked to find out the time and place of a meeting contained in a message from Kim, the task would be represented by the AVM shown in Table 1[13]. To accomplish this task, the user must direct the agent either to read messages from Kim or to read messages about meetings. The agent provides the time and place information. Task success is calculated by examining how

attribute	value
Selection Criteria	Kim ∨ Meeting
Email.att1	10:30
Email.att2	2D516

Table 1: Attribute Value Matrix: Email Scenario Key for Dialogs 1 and 2

well the agent and user achieve the information requirements of the task by the end of the dialog, as measured by the Kappa coefficient[11]. The Kappa coefficient is a measure of agreement between the observed AVM and the ideal AVM (i.e., an AVM where all information elements are correctly obtained) and takes into account the agreement expected by chance.

In addition to being a function of task success, system performance is also a function of a combination of cost measures. Intuitively, cost measures are calculated based on any user or agent dialog behaviors that should be minimized. Cost measures include efficiency measures and qualitative measures. The efficiency measures are directly related to how quickly the dialog was accomplished, whereas qualitative measures reflect the style or feel of the interaction.

Given these definitions of success and costs, performance is then defined as follows:

$$\text{Performance} = (\alpha * \mathcal{N}(\kappa)) + \sum_{i=1}^{n} w_i * \mathcal{N}(c_i)$$

Here α is a weight on Kappa (κ), the cost functions c_i are weighted by w_i, and $\mathcal{N}$ is a Z score normalization function.

Evaluating a dialog system involves having a group of users perform tasks with known "ideal" outcomes (in order to define the AVM key to compare with the observed task outcomes), measuring a wide variety of cost measures for the user-agent dialogs, and measuring user satisfaction with the system during those interactions. Linear regression is then used to determine the subset of factors that is predictive of user satisfaction. Because the factors in the performance equation are normalized, the resultant regression weights indicate the relative importance of the significant factors. For example, in one experiment with the voice e-mail agent, the regression predicted user satisfaction from task success and cost measures of number of user turns in the dialog, number of system turns, elapsed time, number of requests for help, number of time-out prompts, mean recognition accuracy, and the number of speech recognition rejections. The results of the regression analysis demonstrated that mean recognition score and number of user turns were the significant factors, and a second analysis restricted to those factors yielded the performance equation:

$$\text{Performance} = .63 * \mathcal{N}(MeanRecognition) - .32 * \mathcal{N}(UserTurns)$$

accounting for 42% of the variance in user satisfaction.

Our current work involves applying the framework to a variety of dialog agents performing a variety of tasks. We hope these cross-task studies will help develop a general model of the relationship of various factors to user satisfaction. We are also exploring the use of the derived performance function as feedback to the dialog agent to drive its choices of dialog strategies and thereby automatically optimize performance.

SUMMARY

This paper has presented an overview our recent contributions toward creating useful and usable spoken dialog agents. First, we described our contribution to a set of general principles for designing user interfaces to dialog systems that take into account the capabilities and limitations of human cognitive processing. We claim that effective interface design for spoken dialog systems must utilize these principles. Second, we described our recent and ongoing work using the PARADISE evaluation framework. Despite the burgeoning activity in spoken dialog systems, the lack of a consistent integrative framework for evaluation has made it difficult to apply lessons learned from one dialog system to a new application domain. We hope that the broad

application of the principles and framework that we propose will facilitate the identification of general factors influencing user acceptance and will contribute in turn to the development of general principles for building improved spoken dialog systems.

References

[1] S. Bennacef, L. Devillers, S. Rosset, and L. Lamel, "Dialog in the RAILTEL telephone-based system," in *Proc. of the Intl Conf. Spoken Lang. Processing*, 1996, pp. 550–553.

[2] M. Danieli and E. Gerbino, "Metrics for evaluating dialogue strategies in a spoken language system", in *Proc. of the 1995 AAAI Spring Symposium on Empirical Methods in Discourse Interpretation and Generation*, 1995, pp. 34–39.

[3] L. Hirschman and C. Pao, "The cost of errors in a spoken language system", in *Proceedings of the Third European Conference on Speech Communication and Technology*, 1993, pp. 1419–1422.

[4] C. Kamm, S. Narayanan, D. Dutton, and R. Ritenour, "Evaluating Spoken Dialog Systems for Telecommunication Services," in *Proc. EUROSPEECH 97*, 1997, pp. 2203-2206.

[5] C. Kamm, M. A. Walker, and L. Rabiner, "The Role of Speech Processing in Human-Computer Intelligent Communication," *Speech Communication*, in press.

[6] R. Keeney and H. Raiffa, *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*, New York: John Wiley and Sons, 1976.

[7] E. Levin and R. Pieraccini, "CHRONUS, The Next Generation", in *Proceedings of 1995 ARPA Spoken Language Systems Technology Workshop*, 1995.

[8] J. Polifroni, L. Hirschman, S. Seneff, and V. Zue, "Experiments in evaluating interactive spoken language systems", in *Proceedings of the DARPA Speech and NL Workshop*, 1992, pp. 28–33.

[9] B. Schneiderman, *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, Menlo Park, CA: Addison-Wesley, 1986.

[10] E. Shriberg, E. Wade, and P. Price, "Human-machine problem solving using spoken language systems (SLS): Factors affecting performance and user satisfaction", in *Proceedings of the DARPA Speech and NL Workshop*, 1992, pp. 49–54.

[11] S. Siegel, *Nonparametric Statistics for the Behavioral Sciences*, New York: McGraw-Hill, 1956.

[12] M. A. Walker, "The effect of resource limits and task complexity on collaborative planning in dialogue", *Artificial Intelligence Journal*, 1996, vol. 85, pp. 181–243.

[13] M. A. Walker, D. Hindle, J. Fromer, G. Di Fabbrizio, and C. Mestel, "Evaluating Competing Agent Strategies for a Voice Email Agent", *Proc. Eurospeech97*, 1997, pp. 2219-2222.

[14] M. A. Walker, D. J. Litman, C. A. Kamm, and A. Abella, "PARADISE: A Framework for Evaluating Spoken Dialogue Agents," in *Proceedings of ACL/EACL 35th Annual Meeting of the Association for Computational Linguistics*, San Francisco: Morgan Kaufmann, 1997, pp. 271-280.

[15] S. Whittaker and M. A. Walker, "Towards a theory of multimodal interaction," *Proc. AAAI Workshop on Multimodal Interaction*, 1991.