A Three-Tiered Evaluation Approach for Interactive Spoken Dialogue Systems

Kathleen Stibler and James Denny
Lockheed Martin Advanced Technology Laboratories
1 Federal Street, A&E 3W
Camden NJ 08102
{kcomegno, jdenny}@atl.lmco.com

ABSTRACT
We describe a three-tiered approach for evaluation of spoken dialogue systems. The three tiers measure user satisfaction, system support of mission success and component performance. We describe our use of this approach in numerous fielded user studies conducted with the U.S. military.

Keywords
Evaluation, spoken language system, spoken dialogue system

1. INTRODUCTION
Evaluation of spoken language systems is complicated by the need to balance distinct goals. For collaboration with others in the speech technology community, metrics must be generic enough for comparison to analogous systems. For project management and business purposes, metrics must be specific enough to demonstrate end-user utility and improvement over other approaches to a problem.

Since 1998, we have developed a spoken language dialogue technology called Listen-Communicate-Show (LCS) and applied it to demonstration systems for U.S. Marines logistics, U.S. Army test data collection, and commercial travel reservations. Our focus is the transition of spoken dialogue technology to military operations. We support military users in a wide range of tasks under diverse conditions. Therefore, our definition of success for LCS is operational success. It must reflect the real world success of our military users in performing their tasks. In addition, for our systems to be considered successful, they must be widely usable and easy for all users to operate with minimal training. Our evaluation methodology must model these objectives.

With these goals in mind, we have developed a three-tier metric system for evaluating spoken language system effectiveness. The three tiers measure (1) user satisfaction, (2) system support of mission success and (3) component performance.

2. THE THREE-TIERED APPROACH
Our three-tier metric scheme evaluates multiple aspects of LCS system effectiveness. *User satisfaction* is a set of subjective measures that introduces user perceptions into the assessment of the system. *System support of mission success* measures overall system performance with respect to our definition of success. *Component performance* scores the individual system component's role in overall system success.

Collection of user input is essential in evaluation for two reasons. First, it is necessary to consider user perspective during evaluation to achieve a better understanding of user needs. Second, user preference can influence interpretation of success measurements of mission success and component performance. Mission success and component performance are often tradeoffs, with inefficient systems producing higher scores of success. Since some users are willing to overlook efficiency for guaranteed performance while others opt for efficiency, our collection of user input helps determine the relative importance of these aspects.

Mission success is difficult to quantify because it is defined differently by users with different needs. Therefore, it is essential to establish a definition of mission success early in the evaluation process. For our applications, we derive this definition from domain knowledge acquisition with potential users.

It is important to evaluate components individually since component evaluations reveal distinctive component flaws. These flaws can negatively impact mission success because catastrophic failure of a component can prevent the completion of tasks. For example, in the Marine logistics domain, if the system fails to recognize the user signing onto the radio network, it will ignore all subsequent utterances until the user successfully logs on. If the recognition of sign-on completely fails, then no tasks can be completed. In addition, periodic evaluation of component performance focuses attention on difficult problems and possible solutions to these problems [1].

3. EVALUATION METRICS
At the top level of our approach, measurements of overall user satisfaction are derived from a collection of user reactions on a Likert-scaled questionnaire. The questions are associated with eight user satisfaction metrics: ease of use, system response, system understanding, user expertise, task ease, response time, expected behavior and future use. We have categorized our user satisfaction questions in terms of specific metrics as per the PARADISE methodology [5, 2]. These metrics are detailed in Table 1.

Table 1. User Satisfaction metrics

Metric	Description	Example Likert Survey Questions
Ease of Use	User perception of ease of interaction with overall system	The system was easy to use
System Response	Clarity of system response	System responses were clear and easy to understand
System Understanding	System comprehension of the user	The system understood what you said
User Expertise	Shows us how prepared the user felt due to our training	You knew how to interact with the system based on previous experience or training
Task Ease	User ease in performing a given task	It was easy to make a request
Response Time	User's impression of the speed of system's reply	The system responded to you in a timely manner
Expected Behavior	Connection between the user's experience and preconceived notions	The system worked the way that you expected it to
Future Use	Determination of overall acceptance of this type of system in the future	You would use a mature system of this type in the future

The middle tier metrics measure the ability of users to successfully complete their domain tasks in a timely manner. Success, in this case, is defined as completion of a task and segments of the task utilizing the information supplied by the user. A task is considered successful if the system was able to comprehend and process the user's request correctly. It is important to determine if success was achieved and at what cost. The user's ability to make a request in a reasonable amount of time with little repetition is also significant. The mission success metrics fall under nine categories: task completion, task complexity, dialogue complexity, task efficiency, dialogue efficiency, task pace, dialogue pace, user frustration and intervention rate.

For these metrics, we consider the tasks the user is trying to accomplish and the dialogue in which the user has with the system to accomplish those tasks. A session is a continuous period of user interaction with the spoken dialogue system. A session can be examined from two perspectives, task and dialogue, as shown in Figure 1. Segments are atomic operations performed within a task. The success rate of each segment is an important part of the analysis of the system, while the success rate of each task is essential for the comprehensive evaluation of the system. For example, a task of ordering supplies in the Marine logistics domain includes segments of signing onto the radio network, starting the request form, filling in items a through h, submitting the form and signing off the network. Each segment receives an individual score of successfully completion. The Task Completion metric consists of success scores for the overall task and the segments of the task.

Dialogue is the collection of utterances spoken to accomplish the given task. It is necessary to evaluate Dialogue Efficiency to achieve an understanding of how complex the user's dialogue is for the associated task. A turn is one user utterance, a step in accomplishing the task through dialogue. Concepts are atomic bits of information conveyed in a dialogue. For example, if the user's utterance consists of delivery time and delivery location for a particular Marine logistic request, the time and location are the concepts of that turn. These metrics are described in greater detail in Table 2.

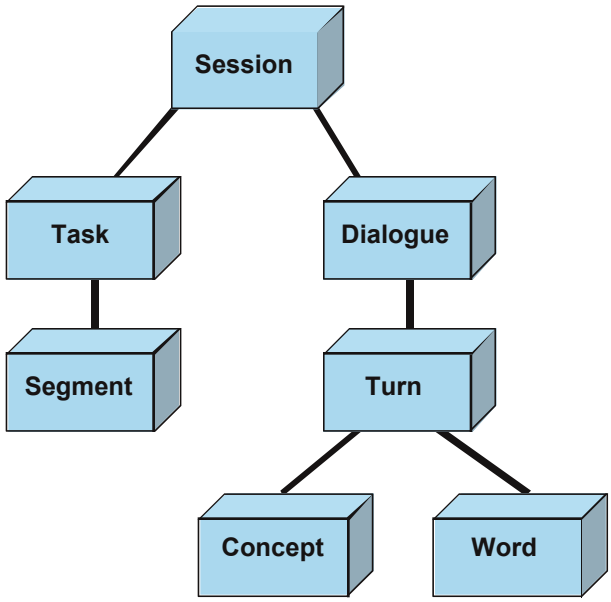

Figure 1. Structural Hierarchy of a Spoken Dialogue System Session

The lowest level tier measures the effectiveness of individual system components along specific dimensions, including component error rates. Overall system level success is determined by how well each component accomplishes its responsibility. This concerns measurements such as word accuracy, utterance accuracy, concept accuracy, component speed, processing errors, and language errors. These measurements aid system developers by emphasizing component weakness. Component Performance metrics also offer explanations for others metrics. For example, bottlenecks within a component may be responsible for slow system response time. Another example is concerned with recognition accuracy. Poor word accuracy may account for low scores of task completion and user satisfaction with the system.

Table 2. Mission metrics

Metric	Description	Measurement
Task Completion	Success rate of a given task	$\frac{\sum \text{correct segments}}{\sum \text{items}}$
Task Complexity	Ideal minimal information required to accomplish a task	$\frac{\sum \text{ideal concepts}}{\text{task}}$
Dialogue Complexity	Ideal amount of interaction with the system necessary to complete a task	$\frac{\sum \text{ideal turns}}{\text{task}}$
Task Efficiency	Amount of extraneous information in dialogue	$\frac{\sum \text{ideal concepts}}{\sum \text{actual concepts}}$
Dialogue Efficiency	Number of extraneous turns in dialogue	$\frac{\sum \text{ideal turns}}{\sum \text{actual turns}}$
Task Pace	Real world time spent entering information into the system to accomplish the task	$\frac{\sum \text{elapsed time}}{\text{task complexity}}$
Dialogue Pace	Actual amount of system interaction spent entering segments of a task	$\frac{\sum \text{turns}}{\text{task complexity}}$
User Frustration	Ratio of repairs and repeats to useful turns	$\frac{\sum (\text{rephrases} + \text{repeats})}{\sum \text{relevant turns}}$
Intervention Rate	How often the user needs help to use the system	$\sum$ (user questions + moderator corrections + system crashes)

Some component performance metrics rely upon measurements from multiple components. For example, Processing Errors combines data transfer errors, logic errors, and agent errors. Those measurements map to the Turn Manager which controls the system's dialogue logic, the Mobile Agents which interface with data sources, and the Hub which coordinates component communication. The metrics are discussed in Table 3.

4. EVALUATION PROCESS

Our LCS systems are built upon MIT's Galaxy II architecture [3]. Galaxy II is a distributed, plug and play component-based architecture in which specialized servers handle specific tasks, such as translating audio data to text, that communicate through a central server (Hub). The LCS system shown in Figure 2 includes servers for speech recording and playback (Audio I/O), speech synthesis (Synthesis), speech recognition (Recognizer), natural language processing (NL), discourse/ response logic (Turn Manager), and an agent server (Mobile Agents) for application/database interaction.

We implement a number of diverse applications and serve a user population that has varying expertise. The combination of these two factors result in a wide range of expectations of system performance by users. We have found that the three-tier system and related evaluation process not only capture those expectations, but also aid in furthering our development.

Our evaluation process begins with conducting a user study, typically in the field. We refer to these studies as Integrated Feasibility Experiments (IFE). Participants involved in the IFEs are trained to use their particular LCS application by a member of our development team. The training usually takes 15 to 30 minutes. The training specifies the purpose of the LCS application in aiding their work, includes a brief description of the LCS architecture, and details the speech commands and

Table 3. Component metrics

Metric	Description	Measurement
Word Accuracy	System recognition per word	NIST String Alignment and Scoring Program
Utterance Accuracy	System recognition per user utterance	$\frac{\sum \text{recognized turns}}{\sum \text{turns}}$
Concept Accuracy*	Semantic understanding of the system	$\frac{\sum \text{recognized concepts}}{\sum \text{concepts}}$
Component Speed	Speed of various components	time per turn
Processing Errors	Percent of turns with low level system error measurements	$\frac{\sum (\text{agent errors} + \text{frame construction errors} + \text{logic errors})}{\sum \text{system turns}}$
Language Errors	Percent of turns with errors in sentence construction, word parsing and spoken output of the system	$\frac{\sum (\text{parse errors} + \text{synthesis errors})}{\sum \text{system turns}}$

*Our use of concept accuracy was inspired by the concept accuracy metric of the PARADISE methodology [5].

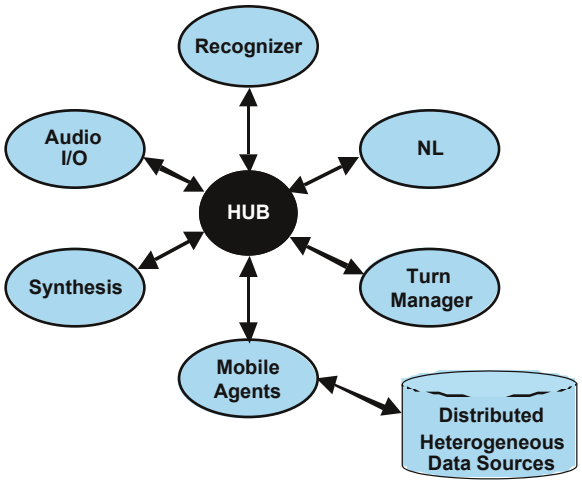

Figure 2. LCS architecture

expected responses through demonstration. After the introductory instruction and demonstration, participants practice interacting with the system.

For each study, we develop a set of scenarios based upon our knowledge of the domain and ask each participant to complete the scenarios as quickly as they can with maximal accuracy and minimal moderator assistance. The study usually consists of approximately five task scenarios of varying difficulty. The scenarios are carried out in fixed order and are given a time limit, generally no longer than 30 minutes. The system logs key events at the Hub, including times and values for the user's speech recording, recognition hypotheses, grammatical parse, resultant query, component speeds, any internal errors, and the system's response. In addition, the moderator notes any assistance or intervention, such as reminding the user of proper usage or fixing an application error. Once the tasks are completed, the user fills out a web-based survey and participates in a brief interview. These determine user satisfaction with the system.

Upon conclusion of a user study, we extract the log files and code the users' recordings through manual transcription. We add diagnostic tags to the log files, noting such events as rephrased utterances and causes of errors and then audit all of the logs for accuracy and consistency. Some of the diagnostic tags that we annotate are number of items and concepts within an utterance, frame construction errors, repeated or rephrased utterances and deficiencies of the training sentence corpus. This is a very time consuming process. Therefore, it is necessary to involve multiple people in this phase of the evaluation. However, one individual is tasked with the final responsibility of examining the annotations for consistency.

A series of scripts and spreadsheets calculate our metrics from the log files. These scripts take the log files as parameters and produce various metric values. While interpreting the metrics values, we may re-examine the log files for an exploration of detail related to particular tasks or events in order to understand any significant and surprising results or trends.

Finally, through a mixture of automated formatting and manual commentary, we create a summary presentation of the user study results. Web pages are generated that contain some of the metrics collected throughout the study.

5. APPROACH VERIFICATION

We have applied our approach in four separate IFEs to date. In each case, our metrics revealed areas for improvement. As these improvements were made, the problems discovered in the next IFE were more subtle and deeply ingrained within the system. Mission success and component metrics aided in the interpretation of user perception and drove future system development. A top-level summary of IFEs, metrics and system improvements is described.

The first IFE was our pilot study, which took place in-house in September 1999. Five subjects with varying military experience were asked to complete three tasks, which were scripted for them. The tier one metrics revealed the users' dissatisfaction with the system responses and the time required in receiving them. These perceptions led to system changes within our Agent and Turn Manager structures that improved the speed of our database agents and more appropriate responses from the LCS system.

The second IFE took place during the Desert Knight 1999 Marine exercise at Twentynine Palms, CA in December 1999. Ten subjects, each an active duty Marine with varying radio operator experience, were given five tasks. This user study offered the subjects the option of following scripts in their tasks. The metrics of tier one showed an increase in overall user satisfaction and revealed the users' difficulty using the system and anticipating its behavior. These concerns influenced future user training and the development of more explicit system responses.

The third IFE occurred during the Marine CAX 6 (Combined Arms Exercise) at Twentynine Palms, CA in April 2000. The seven subjects were active duty Marines, some with minimal radio training. They were required to complete five tasks that had scenario-based, non-scripted dialogues. A combination of tier one, tier two and tier three metrics exposed a deficiency in the speech recognition server, prompting us to increase recognizer training for subsequent IFEs. A recognizer training corpus builder was developed to boost recognition scores.

The most recent IFE was conducted in Gulfport, MS during the August 2000 Millennium Dragon Marine exercise. Six active duty Marines with varied radio experience completed five scenario-based tasks. This time the users expressed concern with system understanding and ease of use through the tier one metrics. The tier three metrics revealed an error in our natural language module, which sometimes had been selecting the incorrect user utterance from recognizer output. This error has since been removed from the system.

The three-tiered approach organizes analysis of the interdependence among metrics. It is useful to study the impact of a metric in one tier against metrics in another tier through principal component analysis. These statistics do not necessarily evidence causality, of course, but they do suggest insightful correlation. This insight exposes the relative significance of various factors' contribution to particular assessments of mission success or user satisfaction.

6. FUTURE ENHANCEMENTS

Although this three-tier evaluation process provides useful metrics, we have identified three improvements that we plan to incorporate into our process: (1) an annotation aide, (2) community standardization, and (3) increased automation. The

annotation aide would allow multiple annotators to review and edit logs independently. With this tool, we could automatically measure and control cross-annotator consistency, currently a labor-intensive chore. Community standardization entails a logging format, an annotation standard, and calculation tools common to the DARPA Communicator project [4], several of which have been developed, but we are still working to incorporate them. The advantage of community standardization is the benefit from tools developed by peer organizations and the ability to compare results. Accomplishing the first two improvements largely leads to the third improvement, increased automation, because most (if not all) aspects from measurement through annotation to calculation then have a controlled format and assistive tools. These planned improvements will make our evaluation process more reliable and less time-consuming while simultaneously making it more controlled and more comparable.

7. CONCLUSION

We have found that structuring evaluation according to the three tiers described above improves the selection of metrics and interpretation of results. While the essence of our approach is domain independent, it does guide the adaptation of metrics to specific applications. First, the three tiers impose a structure that selects certain metrics to constitute a broad pragmatic assessment with minimal data, refining the subject of evaluation. Second, the three tiers organize metrics so that user satisfaction and mission metrics have clear normative semantics (results interpreted as good/bad) and they reveal the impact of low-level metrics (results tied to particular components which may be faulted/lauded). Finally, improvements in selection and interpretation balance satisfaction, effectiveness, and performance, thus imbuing the evaluation process with focus toward utility for practical applications of spoken language dialogue.

8. ACKNOWLEDGEMENT

Thanks to members of the LCS team: Ben Bell, Jody Daniels, Jerry Franke, Ray Hill, Bob Jones, Steve Knott, Dan Miksch, Mike Orr, and Mike Thomas. This research was supported by DARPA contract N66001-98-D-8507 and Naval contract N47406-99-C-7033.

9. REFERENCES

[1] Hirschman, L. and Thompson, H. Survey of the State of the Art in Human Language Technology. Edited by J. Mariani. Chapter 13.1, Overview of Evaluation in Speech and Natural Language Processing. Cambridge University Press ISBN 0-521-592777-1, 1996.

[2] Kamm, C., Walker, M. and Litman, D. Evaluating Spoken Language Systems, American Voice Input/Output Society, AVIOS, 1999.

[3] Seneff, S., Lau, R., and Polifroni, J. Organization, Communication, and Control in the Galaxy-ii Conversational System. Proc. Eurospeech, 1999.

[4] Walker, M. Hirschman, L. and Aberdeen, J. Evaluation for DARPA Communicator Spoken Dialogue Systems. Language Resources and Evaluation Conference, LREC, 2000.

[5] Walker, M. Litman, Kamm, D.C. and Abella, A. PARADISE: A Framework for Evaluating Spoken Dialogue Agents. 35th Annual Meeting of the Association of Computational Linguistics, ACL 97, 1997.