TRAINS-95 System Evaluation

Teresa Sikorski James F. Allen

The University of Rochester
Computer Science Department
Rochester, New York 14627

TRAINS Technical Note 96-3

July 1996

Abstract

In this paper we describe a recent experiment designed to evaluate the performance of the TRAINS-95 system. The evaluation uses a task-based evaluation methodology appropriate for dialogue systems such as TRAINS-95, where a human and a computer interact and collaborate to solve a given problem. In task-based evaluations, techniques are measured in terms of their affect on task performance measures such as how long it takes to develop a solution using the system, and the quality of the final plan produced.

The evaluation explores the robustness of the TRAINS-95 system in the presence of word recognition errors, the amount of training required to effectively use the system, and user preferences.

This work was supported in part by ONR/ARPA grants N00014-92-J-1512 and N00014-95-1-1088, and NSF grant IRI-9503312.

1 Introduction

TRAINS-95 is the first end-to-end implementation in a long-term effort to develop an intelligent planning assistant that is conversationally proficient in natural language. The initial domain is a train route planner, where a human manager and the system must cooperate to develop and execute plans [Allen *et al.*, 1996; Ferguson *et al.*, 1996]. TRAINS-95 provides a real-time multi-modal interface between the human and computer. In addition to making menu selections and clicking on objects using a mouse, the user is able to engage in an English-language dialogue with the computer using either keyboard or speech input.

The impetus behind the development of TRAINS-95 was the desire to implement and thereby test computational theories of planning, natural language and dialogue, which have long been focal research areas of the Computer Science Department at the University of Rochester. Once the first version of TRAINS-95 was operational in January 1995, the question of how to evaluate the system needed to be addressed. Since this was just a first step in an incremental development, we wanted to develop an evaluation method which would allow us to easily measure the progress made by subsequent versions of the system, and the effect of various design decisions on the performance of the system as a whole. Furthermore, we wanted to be able to use the evaluation results to guide us in system debugging, and to some extent our future research focus.

Standard accuracy models used to evaluate speech recognition and data base query tasks such as ATIS [Hirschman *et al.*, 1993] are not appropriate for a dialogue evaluation. There is no right answer to an utterance in a dialogue. Rather, there are many different possible ways to answer, each of them equally valid. Some may be more efficient along some dimension or another, but there is no single best answer.

There are also a range of technology-based evaluations that could be performed, such as checking whether some agreed upon syntactic structure is produced, checking if referring expressions are correctly analyzed, or possibly checking if the right speech act interpretation is produced. These techniques are useful, but miss the real point. For instance, does it matter if the parse structures produced are faulty in some way if this is later compensated for by the discourse processing and the system responds in a way that best furthers the task the human is performing? The ultimate test of a dialogue system is whether it helps the user in performance of some task. This means that measures like time to completion and the effectiveness of the solution produced are critical. Of course, accuracy measures provide insight into various components of the system and thus will always be useful. But they will never answer the ultimate question of how well the system works. Accepting this argument, however, involves a shift of perspective from viewing the problem as a spoken language understanding problem to a problem that is more closely related to human factors and human computer interfaces.

The experiment described here is our first attempt to perform a task-based evaluation of a dialogue system. For our application, the task-based evaluation was performed in terms

of two parameters: time to task completion and the quality of the solution. Our measure of solution quality was based on whether routes were planned to move trains from an initial configuration to a requested final configuration and, in cases where these goals were achieved, the number of time units required to travel to planned routes.

Our task-based evaluation methodology is domain-independent in that it can be applied to any system in which there are objectively identifiable goal states and solution quality criteria. For most planning applications, these are realistic constraints.

The solution quality criteria in the TRAINS-95 domain is extremely simple, but our method is extensible to future versions of the system where there will be an interplay between costs of various resources, all of which can be ultimately translated to a monetary cost. Once the goal state and solution quality criteria are defined in objective terms, the evaluation can be completely automated. This gives the task-based evaluation method a significant advantage over other techniques [Polifroni *et al.*, 1992; Walter, 1992] where human evaluators must intervene to examine individual responses and assess their correctness. This human intervention is both costly and introduces a subjective element that is unnecessary when task-based evaluation is applicable.

1.1 Evaluation Goals

By performing the experiment described herein, we have attempted to provide a quantitative evaluation of the level of success achieved by the TRAINS-95 implementation. A primary goal of TRAINS-95 was to develop a dialogue system that could exhibit robust behavior despite the presence of word recognition errors. The overall goal of the evaluation was to test this robustness. Other issues addressed in the evaluation include the following:

- the level of training required to effectively use the system
- establishment of an evaluation methodology and a baseline against which to evaluate future versions of the system
- identification of system deficiencies
- observations regarding user input mode preferences

1.2 Hypotheses

Our initial hypotheses were:

- Speech input is more time-efficient than keyboard input for accomplishing routing tasks using the TRAINS-95 system despite the presence of word recognition errors.
- Subjects with minimal training can accomplish routing tasks using the TRAINS-95 system.

- If word recognition accuracy is poor, the time taken to accomplish the task will increase.
- Users of the TRAINS-95 system prefer speech input over keyboard input.

Below we report the results of experiments conducted to test these hypotheses.

2 Experimental Design

In the experiment, input medium and word recognition accuracy were varied to determine their effect on our task performance measures (time to arrive at a solution and quality of the solution).

All subjects were given identical sets of 5 routing tasks to perform, in the same order. Half of the subjects were asked to use speech for the first task, keyboard for the second, speech for the third and keyboard for the fourth. The other half used keyboard for the first task and then alternated. All subjects were given a choice of whether to use speech or keyboard input to accomplish the final task.

Word recognition accuracy was varied by using an error-correcting speech recognition post-processor for half of the subjects (four subjects using keyboard input first, four using speech input first). The speech recognition post-processor has been successful in improving the word recognition accuracy rate by an additional 5%, on average, in the TRAINS-95 domain [Ringger and Allen, 1996].

Figure 1 represents subject groupings based on which input medium they started with and whether their speech recognition output was run through the post-processor before being input to the TRAINS-95 parser. We will consistently refer to subjects by their subject number throughout this document.

	Sphinx-II Only	Post-processor Enabled
Keyboard Input First	Subjects 1-4	Subjects 9-12
Speech Input First	Subjects 5-8	Subjects 13-16

Figure 1: Subject Groupings

2.1 Experimental Environment

Overview

The experiment was performed over the course of a week in November 1995. Each of the sixteen subjects participated in a session with the TRAINS-95 system which lasted approximately 45 minutes.

Hardware and Software Configuration

All sixteen sessions were conducted in the URCS Speech Lab using identical hardware configurations. The software components used in the experiment included:

- A Sphinx-II speech recognizer developed at CMU [Huang *et al.*, 1993], running on a DEC Alpha
- TRAINS-95 version 1.3 including the speech recognition post-processor running on a SPARCstation 2
- TrueTalk, a commercial off-the-shelf speech generator (available from Entropics, Inc.) running on a SparcLX

Subjects, working at a Sun SPARCstation, wore a headset with a microphone to communicate with the speech recognizer. The TRAINS-95 system uses a click-and-hold protocol for speech input. The acoustic data from this speech input was recorded and later transcribed manually. Word recognition accuracy was computed by comparing these transcriptions of what the subject actually said to transcripts of what was input to the parser from the speech recognition module or post-processor.

The TRAINS-95 system communicated with the subjects verbally using the speech generator, visually by highlighting items on the map, and textually, through a text output window and dialog boxes.

An example of the initial configuration of a railway map used by the TRAINS-95 system appears in Figure 2. The actual system uses color displays. As a route is planned, the proposed route is highlighted in a unique color. Cities that the system has understood to be goals or has identified to the user as causing delays are also highlighted.

Subjects

Sixteen subjects for the experiment were recruited from undergraduate computer science courses. None of the subjects had ever used the TRAINS-95 system before, and only five reported having previously used *any* speech recognition system. All subjects were native speakers of American English. Fifteen subjects were male, one was female.

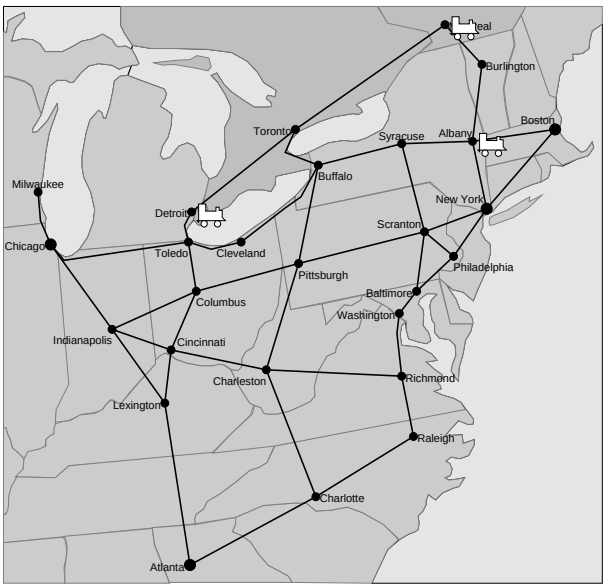

Figure 2: TRAINS-95 Map

2.2 Task Selection

The experiment's routing scenarios were designed with the following restrictions to ensure non-trivial dialogs and avoid drastic complexity variation among tasks:

- Each task entails moving three trains to three cities, with no restriction on which train goes to which city.
- In each scenario, three cities are experiencing delays.
- One of the three routes in each scenario requires more than four time units to travel.

Table 2.2 shows the scenarios used for each of the 5 tasks.

2.3 Procedure

The procedure for the experiment was as follows:

Tutorial

The subject viewed an online tutorial lasting 2.4 minutes. The tutorial, which was developed for the purpose of the experiment, described how to interact with the TRAINS-95 system using speech and keyboard input, and demonstrated typical interactions using each of these media. Although the demonstration gave the subject an indication of how he was

	Start cities	Cities with delays	Goals
Scenario 1	Atlanta	Buffalo	Boston
	Charlotte	Cincinatti	Chicago
	Cleveland	Columbus	Montreal
Scenario 2	Milwaukee	Columbus	Atlanta
	New York	Detroit	Lexington
	Richmond	Indianapolis	Toronto
Scenario 3	Cleveland	Buffalo	Albany
	Milwaukee	New York	Lexington
	Scranton	Syracuse	Raleigh
Scenario 4	Boston	Buffalo	Lexington
	Charlotte	New York	Philadelphia
	Toronto	Pittsburgh	Washington
Scenario 5	Albany	Baltimore	Lexington
	Detroit	Philadelphia	Milwaukee
	Montreal	Scranton	Washington

Table 1: Experiment Scenarios

expected to speak to the system, the subject was given no explicit direction about what could be said. The tutorial encouraged the subject to "speak naturally, as if to another person". While we are aware that human-computer dialogue is significantly different than human-human dialogue, there is evidence that such instructions given to test subjects does significantly reduce unnatural speech syles such as hyperarticulation [Shriberg *et al.*, 1992].

Since it was important for the subject to understand how the quality of his solutions would be judged, the tutorial also explained how to calculate the amount of time a route takes to travel. For simplicity, subjects were told to assume that travel between any two cities displayed on the map takes one time unit. Additional time units are charged if the planned route includes cities experiencing delays or if routes for different trains interfere with eachother. (The system informs users if either of these situations exist.) Finally, to put the subjects at ease, the tutorial emphasized that the TRAINS-95 system was being evaluated rather than the user.

Practice Session

The subject was allowed to practice both speech and keyboard input before actually being given a task to accomplish. At the outset of the practice session, the subject was given a list of practice sentences that was prepared by a TRAINS-95 developer (Appendix A). Only the display and speech input modules were running during the practice session. Although there was no time limit, subjects spent no more than two minutes practicing, and several subjects chose not to practice keyboard input at all. During the practice session, the output from the speech recognition module was displayed in a separate window, showing the subject

how his speech was being "heard" by the system. This gave the subject an opportunity to experiment with different speech rates, enunciations, and microphone positions in order to accomodate the speech recognizer. The subject also learned to avoid common input errors such as beginning to speak before clicking, and releasing before completing an utterance.

Task Execution

At the outset of each task, the subject was handed a 5"x8" index card with the task instructions and a map highlighting the destinations. The index cards specified which input medium was to be used, the destinations of the trains, and some additional information about cities to be avoided. (The exact instructions given to the subjects are provided as Appendix B.) The subject didn't know the initial location of the trains until the map with the initial configuration was displayed on the CRT. The instructions for the first two tasks were simply to plan routes to get the trains to their destinations. Instructions for the three remaining tasks asked the subjects to find efficient routes as quickly as possible. During this phase of the experiment, the subject had no interaction with the experimenter.

Questionnaire

After completing the final task, the subject was given a questionnaire that solicited the subject's impressions about the cause of any difficulty encountered, reasons for selecting speech or keyboard input for the final task, and recommendations for system improvements. The questionnaire and responses are provided in Appendix C.

3 Experiment Results

- Metrics were collected for each subject on a per-task basis. Appendix D contains tables detailing the raw data collected. The figures appearing in this section summarize the data.
- Of the 80 tasks attempted, there were 7 tasks in which the stated goals were not met. (Note that the figures do not include dialogues in which the goals were not accomplished.)
- Of the 16 subjects, 12 selected speech as the input mode for the final task and 4 selected keyboard input.
- Figures 3 and 4 show that the plans generated when speech input was used are of similar quality to those generated when keyboard input was used. However, for each task, the amount of time to develop the plan was significantly lower when speech input was used. Speech input was $28 - 49\%$ faster. This performance of speech input over keyboard input in our experiment is in contrast with experimental results obtained on some previous systems [Rudnicky, 1993]. The experiment results are consistent with our expectations that as speech recognition technology improves and robust techniques for dialogue systems are developed, human-computer interaction using speech

input will become increasingly more efficient in comparison with textual input as is the case with human-human interaction [Cohen and Oviatt, 1994].

Note that in Figures 3 and 4, we give the average for Tasks 1-4, as well as the average overall 5 tasks. The average for Tasks 1-4 may be a more accurate point of comparison since in Task 5 the subjects were using the medium with which they were most comfortable.

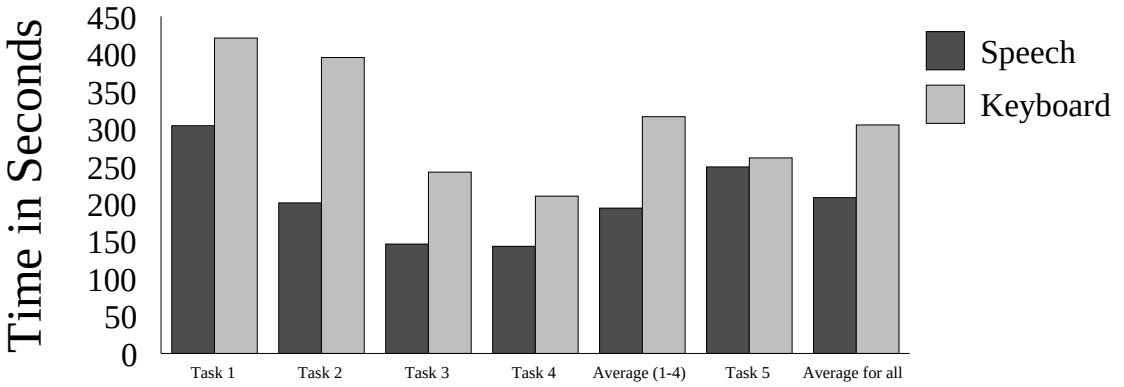

Figure 3: Average Time To Solution

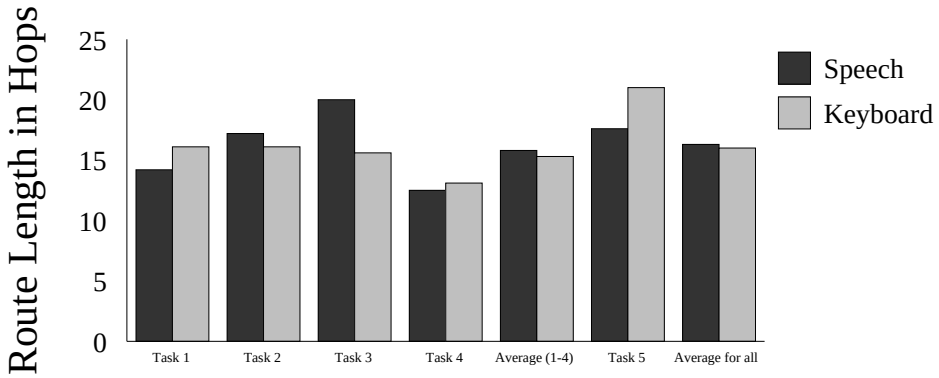

Figure 4: Average Length of Planned Routes

- Figure 5 plots word recognition accuracy versus the time to complete the task for each of the 5 tasks.

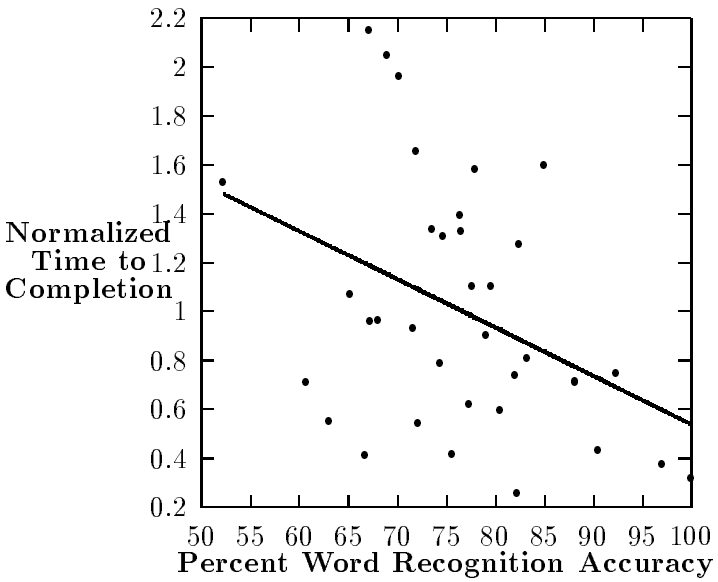

Figure 5: Recognition Accuracy Versus Completion Time

4 Discussion

Of the four subjects selecting keyboard for the final task, one had been unable to complete a task because of a series of speech recognition errors. Another had experienced poor speech recognition and had actually been able to arrive at solutions more quickly using keyboard input. A third had solved the first four tasks in approximately the same amount of time using keyboard input as using speech input. The final subject selecting keyboard input indicated on his questionnaire that he believed he could solve the problem more quickly using the keyboard; however, the subject had solved the two tasks using speech input 19

In each task where the goals were not met, the subject was using speech input. Interestingly, there was no particular task that was troublesome and no particular subject that had difficulty. Seven different subjects had a task where the goals were not met, and each of the five tasks was left unaccomplished at least once.

A review of the transcripts for the unsuccessful attempts revealed that in three cases, the subject misinterpreted the system's actions, and ended the dialogue believing that all the goals had been met. In each of these cases, the misunderstanding occured when the system identified a destination mentioned by the subject by highlighting the city on the map. The subjects took that action as meaning that a route had been planned between the initial location of the train and the destination eventhough no proposed route was highlighted.

Each of the other four unsuccessful attempts resulted from a common sequence of events: after the system proposed an inefficient route, word recognition errors caused the system to

misinterpret rejection of the proposed route as acceptance. The subsequent subdialogues intended to improve the route were interpreted to be extensions to the route, causing the route to "overshoot" the intended destination. In each of these scenarios, the subject did get the train to pass through the destination specified in the task instructions, but the planned route did not terminate at the required city.

One of our hypotheses was that as word recognition accuracy degraded, the time to arrive at a solution would increase. Figure 5 depicts the relationship between task completion time and word recognition accuracy observed during the experiment, where the time to completion figures have been normalized by task. We expected to find a strong negative correlation in Figure 5. However, the correlation coefficient of the best fit line in the figure is only 15.7%. We've identified three possible explanations as to why we didn't find as significant a correlation between word recognition accuracy and time to completion as we expected:

- Robust Parsing

  The word recognition accuracy measurement used in the evaluation was computed as follows:

  $$WRA = \frac{N - D - S - I}{N}$$

  where
  WRA = word recognition accuracy
  N = the number of words actually spoken,
  D = the number of words deleted,
  S = the number of word-for-word substitutions, and
  I = the number of words inserted

  The TRAINS-95 system makes use of a robust chart parser which is often able to form an interpretation for an utterance even when many of the words are misrecognized. This makes it less important how many words are correctly recognized as *which* words are correctly recognized. Since the word recognition accuracy measure we used treated all words as having equal significance, it was not as telling a statistic as initially expected.

- Nonunderstanding vs. Misunderstanding

  Many times, the TRAINS-95 system cannot form an interpretation of an utterance due to poor word recognition. When the TRAINS-95 system experiences nonunderstanding, it enters into a clarification subdialogue, and takes no action based on the misunderstood utterance. The user is then able to repeat or rephrase the utterance, and the dialogue continues. When nonunderstanding occurs, both the user and the

system are aware of the situation and are able to quickly and easily rectify the situation.

Misunderstandings, on the other hand are often undetected initially, and are more time-consuming to fix, since the system *has* taken action based on its erroneous interpretation of the utterance. Misunderstandings are therefore significantly more detrimental to task performance than instances of nonunderstanding. Since misunderstandings occur even in dialogues where the subject experiences relatively good speech recognition, high word recognition accuracy and low time to completion do not always correlate.

Our experiment demonstates that a robust approach can create a high variance in the effectiveness of an interaction. Note, however that a comparison of two dialogue systems - one taking a conservative approach that only answers when it is confident (cf. [Smith and Hipp, 1994]), and a robust system that proceeds based on partial understanding - showed that the robust system was significantly more successful in completing the same task [Polifroni *et al.*, 1992].

- Random Routes

Designers of the TRAINS-95 system suspected that since the initial domain is so simple, there would be very limited dialogue between the human and the computer. In order to stimulate dialogue, the designers used a deliberately weak planner that has the following properties:

  - The planner needs to ask the human for help if given a destination more than four "hops" from the original location of the train.
  - The planner randomly selects between possible routes when the human does not specify a specific route.

The second property entered an amount of nondeterminism into the experiment that couldn't be compensated for by the small number of subjects we used. Some subjects that had good word recognition accuracy had to spend a significant amount of time improving inefficient routes generated by the planner. In some other tasks, where poor recognition accuracy was a problem, the planner fortuitously generated efficient routes the first time.

Exascerbating the problem was the system's poor handling of subdialogues where the subject attempted to modify routes. Most of the subjects expressed their frustration with this aspect of the system on the questionnaire. After several interactions with the system aimed at improving routes, subjects many times either gave up on the task or restarted the scenario, losing good routes that had been previously completed. Other subjects left the inefficient routes as they were, adversely affecting the quality of the

solution. These problems reveal a need for better handling of corrections, especially as resumptions of previous topics.

5 Future Directions

Development of TRAINS-96 is near completion. TRAINS-96 will have an expanded domain involving realistic distances and travel times between cities with associated costs. Time and cost contraints will be added to the given tasks. Since richer dialogues will be a natural consequence of the richer domain, the system will no longer contain a weak planner, thus eliminating many of the problems artificially introduced during the evaluation of TRAINS-95.

A fundamental design decision in TRAINS-95 was to choose a specific interpretation when faced with ambiguity, thus risking misinterpretation, rather than entering into a clarification subdialogue. We plan to evaluate the effectiveness of this strategy in later versions of the system.

Future evaluations will also involve a comparison of the effectiveness of a human solving a given routing task alone (with the aid of a spreadsheet and calculator) versus a human solving the same task with the assistance of the TRAINS system.

Future evaluations may also include additional independent variables such as task complexity and additional interface modalities. We have hypothesized that the expanded domain will stimulate more spontaneous dialogue, and we would therefore like to perform experiments that evaluate the richness and spontaneity of the language used by the subjects using new versions of the system. This type of evaluation will require that we establish a fairly objective measure of spontaneity and richness, perhaps based on the extent of the vocabulary used by subjects, the length of subjects' utterances, and the presence of various linguistic phenomena such as speech repairs, anaphoric reference, etc.

In our evaluation of the TRAINS-95 system, we relied on word recognition accuracy to give us an indication of the system's level of understanding of the dialogue. As Figure 5 demonstrates, word recognition accuracy did not have an especially strong correlation with task performance measured in terms of time to solution in our experiment. In the TRAINS-96 evaluation, we intend to employ more sophisticated techniques that will measure understanding of utterances and concepts rather than words, perhaps based on the recognition accuracy of certain key words used by the robust chart parser. Recent research indicates a linear relationship between word recognition accuracy and the understanding of utterances [Boros *et al.*, 1996].

The evaluation of the TRAINS-95 system had coarse granularity in that it measured the amount of time taken to complete the task and the quality of the solution as a whole.

During the TRAINS-96 evaluation we hope to also be able to perform an evaluation of the system's performance during various types of subdialogues. Unfortunately, an *automatic* task-based analysis will not always be possible at a subdialogue level, but, in general, a transcript review will indicate what the goal of the subdialogue was and whether the goal was met. Evaluating at a subdialogue granularity may give system developers a better indication of where system improvements are most needed.

References

[Allen *et al.*, 1996] James F. Allen, Brad Miller, Eric Ringger, and Teresa Sikorski, "Robust Understanding in a Dialogue System," In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, 1996.

[Boros *et al.*, 1996] M. Boros, W. Eckert, F. Gallwitz, G. Görz, G. Hanrieder, and H. Niemann, "Towards Understanding Spontaneous Speech: Word Accuracy Vs. Concept Accuracy," Technical report, Universität Erlangen-Nürnberg, May 1996.

[Cohen and Oviatt, 1994] P. Cohen and S. Oviatt, "The Role of Voice Input for Human-Machine Communication," In *Proceedings of the National Academy of Sciences*, 1994.

[Ferguson *et al.*, 1996] George Ferguson, James Allen, and Brad Miller, "TRAINS-95: Towards a Mixed-Initiative Planning Assistant," In *Proceedings of the Third Conference on Artificial Intelligence Planning Systems*, 1996.

[Hirschman *et al.*, 1993] L. Hirschman, M. Bates, D. Dahl, W. Fisher, J. Garofolo, D. Pallet, K. Hunicke-Smith, P. Price, A. Rudnicky, and E. Tzoukermann, "Multi-Site Data Collection and Evaluation in Spoken Language Understanding," In *Proceedings of the ARPA Human Language Technology Workshop*, pages 19–24, 1993.

[Huang *et al.*, 1993] D. Huang, F.Alleva, H.W. Hon, M. Y. Hwang, K. F. Lee, and R. Rosenfeld, "The Sphinx-II Speech Recognition System: An Overview," *Computer, Speech and Language*, 1993.

[Polifroni *et al.*, 1992] J. Polifroni, L. Hirschman, S. Seneff, and V. Zue, "Experiments in Evaluating Interactive Spoken Language Systems," In *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 28–33, 1992.

[Ringger and Allen, 1996] E. Ringger and J. F. Allen, "Error Correction Via A Post-Processor For Continuous Speech Recognition," In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1996.

[Rudnicky, 1993] A. Rudnicky, "Mode Preferences in a Simple Data Retrieval Task," In *Proceedings of the ARPA Human Language Technology Workshop*, pages 364–369, 1993.

[Shriberg *et al.*, 1992] 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 Natural Language Workshop*, pages 49–54, 1992.

[Smith and Hipp, 1994] R. Smith and R. D. Hipp, *Spoken Natural Language Dialog Systems: A Practical Approach*, Oxford University Press, 1994.

[Walter, 1992] S. Walter, "Neal-Montgomery NLP System Evaluation Methodology," In *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 323–326, 1992.

A TRAINS Practice Sentences

- Send the train from Atlanta to Philadelphia.
- Go to Chicago via Pittsburgh.
- No, go through Atlanta instead.
- The train at Toronto should go to New York via Buffalo and Syracuse.
- Good.
- Don't go through Scranton.
- And then up to Richmond.
- Cancel that.
- I'm done.

B Task Instructions

B.1 Task 1

Using the keyboard, perform the following task:

Montreal, **Boston** and **Chicago** each need a train to be moved there. Lake-effect snow is causing delays in Buffalo. Heavy traffic in Central and Southern Ohio is causing delays and should be avoided.

When you have completed your plan, or if you want to give up, type that you are done.

B.2 Task 2

Using speech input, perform the following task:

You need to construct a plan to get one train to **Toronto**, one to **Lexington** and a third to **Atlanta**. Expect delays through Indianapolis and Columbus due to bad weather. Heavy traffic in Detroit is also causing delays there.

When you have completed your plan, or if you want to give up, simply tell the system that you are done.

B.3 Task 3

Using the keyboard, perform the following task as quickly as possible:

You need to plan optimal routes to move trains to **Albany**, **Raleigh** and **Lexington**. Much of New York State has been paralyzed by freezing rain, so routes through New York should be avoided as much as possible.

When you have completed your plan, or if you want to give up, type that you are done.

B.4 Task 4

Using speech input, perform the following task as quickly as possible:

Trains are needed at **Lexington**, **Philadelphia** and **Washington** in as little time as possible. Traffic congestion in New York City is causing delays there. Additionally, the areas close to Pittsburgh and Buffalo are experiencing heavy snowfall and long delays are likely through those cities.

When you have completed your plan, or if you want to give up, simply tell the system that you are done.

B.5 Task 5

Using the input method of your choice (keyboard or speech), perform the following task as quickly as possible:

You need to find optimum routes to get one train to **Milwaukee**, one to **Lexington** and a third to **Washington**. Heavy storms are causing delays in Baltimore and Eastern Pennsylvania.

When you have completed your plan, or if you want to give up, inform the system that you are done using the appropriate method.

C TRAINS Questionnaire

Thank you for participating in the first TRAINS system evaluation. To help us understand the experiment better, please answer the following questions.

Name:

1. Have you ever used the TRAINS system before?:

2. Have you ever used a speech recognition system before?:

3. Why did you pick the mode you did (keyboard or speech) for the final session? You may pick more than one.

   (a) The mode you picked was easier for you to use?

   (b) It allowed you to solve the problem more quickly?

   (c) It was more fun to use?

   (d) Please mention any other reasons you had.

4. When you had difficulty, what reason do you think was mostly the cause? Rate each as very important to not at all.

   (a) The system had too many speech recognition errors.
   VERY IMPORTANT SOMEWHAT IMPORTANT NOT IMPORTANT

   (b) The system had trouble understanding the language in general.
   VERY IMPORTANT SOMEWHAT IMPORTANT NOT IMPORTANT

   (c) The system is not a very good route planner.
   VERY IMPORTANT SOMEWHAT IMPORTANT NOT IMPORTANT

5. What would you suggest to make the system more effective? You may use the back if you don't have enough space.

C.1 Questionnaire Responses

Subject	Previous TRAINS Use	Previous SR Use	Medium Used For Final Task	Easier To Use	Faster	More Fun
1	No	No	SPEECH	√	√	√
2	No	No	KEYBOARD	√	√	√
3	No	No	SPEECH	√	√	√
4	No	No	SPEECH	√	√	
5	No	Yes	SPEECH	√	√	√
6	No	No	SPEECH	√		√
7	No	No	SPEECH	√	√	
8	No	No	SPEECH		√	√
9	No	Yes	SPEECH	√		√
10	No	No	KEYBOARD	√		
11	No	No	SPEECH	√		√
12	No	No	SPEECH	√	√	√
13	No	Yes	KEYBOARD	√	√	
14	No	No	SPEECH	√		√
15	No	Yes	KEYBOARD	√	√	
16	No	Yes	SPEECH			√

Subject	Other reasons for input medium choice:
1	My main problems were with correcting incorrect paths, and that was just as hard with each input mode.
2	I can type fast. It was annoying to modulate the speed of my speech to allow the appropriate actions to occur.
7	I picked speech but wanted to switch.
11	It seemed like it was having trouble understanding my meaning instead of my words.
14	I can't type, so it is naturally faster for me to give oral commands (even if they must be repeated).
15	The system didn't understand me well. I had more of a chance to correct mistakes using the keyboard since things weren't processed until I pressed enter. I couldn't correct erroneous paths easily (i.e. at all) using speech.
16	I wanted to challenge Sphinx.

Perceived Cause of Difficulties									
	Speech Recognition Errors			Poor Language Understanding			Poor Route Planning		
Subject	Very	Some what	Not	Very	Some what	Not	Very	Some what	Not
1			√		√			√	
2	√				√			√	
3		√		√					√
4		√				√		√	
5		√				√		√	
6		√			√				√
7		√		√				√	
8		√				√	√		
9		√				√		√	
10		√			√			√	
11			√	√				√	
12			√			√		√	
13			√			√		√	
14			√			√	√		
15	√			√				√	
16		√		√			√		

Subject	Suggestions for the TRAINS system
1	Better understanding of the word "don't"
2	The interface would be much easier to use if I could merely list the cities along a route, or select them all by clicking.
3	The system staunchly resisted my efforts to backtrack when I found that I'd chosen an inefficient route. It couldn't understand that I wanted to "erase" a train's proposed route (hence (b), above), and misunderstood my requests to "start over" (hence (a), above). A means of refering to each train (color, number, etc.) would help, as well as perhaps an automatic trail/time calculator (for the mathematically lazy user
4	Allow better understanding of rerouting. Display estimated travel time for a train and for travel time comparisons between various human selected routes.
5	Label the trains, that way it would be possible to work on multiple routes more easily. I tried to change a route after starting on another train and couldn't get back from Charleston to Richmond. I think it needs work on the 'manipulation' words i.e. move back, back up from _ to _. etc. But, it was great.
6	I had trouble getting it to go back to a previously set route and change it. Words like "instead" or "change that" might help in other tasks as well. I like the different phrases it uses.
7	For such a simplified problem a concise set of words and command can probably constrain the problem better making it easier to solve. A way to cut and paste would make typing faster. Also it would be useful to know what the computer is thinking so you can know the appropriate way to rephrase your request.
8	Let people start over on the planning - my one lousy route wouldn't let me try again and I got lost as to where I was.
9	Confirming the speech pattern on the screen or using audio before implementing the commands
10	Ask more specific questions when the system is confused A spelling corrector for the typing interface A few times it picked routes with loops in them. Why?
11	It should pay more attention to prepositions, and be willing to accept the possibility that it made a mistake earlier.
12	Occasionally, I didn't know how much delay was introduced due to e.g. congestion, so didn't know if it was better to go around. Worked pretty well. But I think that using a mouse to point at cities would be faster for this application.

Subject	Suggestions for the TRAINS system
13	The biggest problem was if it made a choice for a route and I didn't like it often I would try to "back up" from where it had placed me, but it would just create cyclic routes.
14	Write a routine that simulates travel time (given the 5 hours between stations) and, like a state space, assigns values for each station. This way the computer can find the least expensive (time wise) route - even if that route includes delays at busy stations.
15	I ended up planning the routes because I had such a hard time correcting erroneous ones. Make it easier to correct errors - perhaps by erasing an entire route so that you can specify it if the planner generates errors. It often seemed to try to continue from the point at which I had stopped, but often that was already on the way to a hazardous situation and turning around is silly, seeing that I haven't started yet and that need could be avoided.
16	The user should be able to use cities as adjectives. The system shouldn't send one train through the same city more than once.

D Data Collected

Dialogs where the goals were not accomplished are indicated by an asterisk. Note that Subject 16 started with the second task and repeated the fifth task twice. The second set of data for his fifth task has been omitted from our results, since having previously solved the problem skewed his performance.

D.1 Task 1

Subject	Input Medium	Word Recognition Accuracy	Completion Time (seconds)	Route Length (hops)
1	KEYBOARD	NA	393	15
2	KEYBOARD	NA	402	26
3	KEYBOARD	NA	128	14
4	KEYBOARD	NA	242	14
5	SPEECH	72.1	165	12
6	SPEECH	76.4	425	13
7	SPEECH	66.7	126	16
8	SPEECH	82.2	79	15
9	KEYBOARD	NA	350	17
10	KEYBOARD	NA	200	14
11	KEYBOARD	NA	1364	17
12	KEYBOARD	NA	292	12
13	SPEECH	69.0	624	17
14	SPEECH	73.6	406	12
15	SPEECH	77.4	*	*

D.2 Task 2

Subject	Input Medium	Word Recognition Accuracy	Competion Time (seconds)	Route Length (hops)
1	SPEECH	78.0	318	13
2	SPEECH	71.4	*	*
3	SPEECH	74.4	159	15
4	SPEECH	92.3	151	13
5	KEYBOARD	NA	1223	13
6	KEYBOARD	NA	522	13
7	KEYBOARD	NA	376	13
8	KEYBOARD	NA	105	15
9	SPEECH	85.0	322	17
10	SPEECH	88.2	144	15
11	SPEECH	68.1	194	15
12	SPEECH	75.6	*	*
13	KEYBOARD	NA	122	13
14	KEYBOARD	NA	134	15
15	KEYBOARD	NA	282	31
16	SPEECH	80.5	120	9

D.3 Task 3

Subject	Input Medium	Word Recognition Accuracy	Completion Time (seconds)	Route Length (hops)
1	KEYBOARD	NA	126	16
2	KEYBOARD	NA	287	16
3	KEYBOARD	NA	222	13
4	KEYBOARD	NA	144	17
5	SPEECH	70.2	*	*
6	SPEECH	88.1	104	13
7	SPEECH	67.3	140	16
8	SPEECH	77.6	161	42
9	KEYBOARD	NA	166	14
10	KEYBOARD	NA	129	17
11	KEYBOARD	NA	809	16
12	KEYBOARD	NA	166	16
13	SPEECH	74.7	191	16
14	SPEECH	52.3	223	16
15	SPEECH	97.0	55	17
16	KEYBOARD	NA	129	13

D.4 Task 4

Subject	Input Medium	Word Recognition Accuracy	Completion Time (seconds)	Route Length (hops)
1	SPEECH	90.5	62	12
2	SPEECH	70.2	280	12
3	SPEECH	78.8	*	*
4	SPEECH	79.0	129	12
5	KEYBOARD	NA	232	13
6	KEYBOARD	NA	202	12
7	KEYBOARD	NA	349	13
8	KEYBOARD	NA	258	13
9	SPEECH	76.5	190	12
10	SPEECH	82.4	182	12
11	SPEECH	71.6	133	15
12	SPEECH	75.6	60	12
13	KEYBOARD	NA	178	13
14	KEYBOARD	NA	174	15
15	KEYBOARD	NA	77	13
16	SPEECH	82.0	106	7

D.5 Task 5

Subject	Input Medium	Word Recognition Accuracy	Completion Time (seconds)	Route Length (hops)
1	SPEECH	63.1	138	15
2	KEYBOARD	NA	219	24
3	SPEECH	77.3	155	15
4	SPEECH	86.1	*	*
5	SPEECH	79.6	275	17
6	SPEECH	67.2	536	17
7	SPEECH	65.2	267	23
8	SPEECH	79.1	*	*
9	SPEECH	83.3	202	15
10	KEYBOARD	NA	196	17
11	SPEECH	71.9	413	33
12	SPEECH	100.0	79	15
13	KEYBOARD	NA	844	20
14	SPEECH	60.8	177	15
15	KEYBOARD	NA	60	23
16	SPEECH	77.9	**	**