A Task-Based Evaluation of the TRAINS-95 Dialogue System

Teresa Sikorski and James F. Allen

University of Rochester, Rochester, NY 14627, USA

Abstract. This paper describes a task-based evaluation methodology appropriate for dialogue systems such as the TRAINS-95 system, 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. We report recent experiment results which explore the effects of word recognition accuracy and speech versus keyboard input on task performance.

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 [1]. 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 an evaluation method that 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 [4] 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 about 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 are 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 [7, 12] 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.

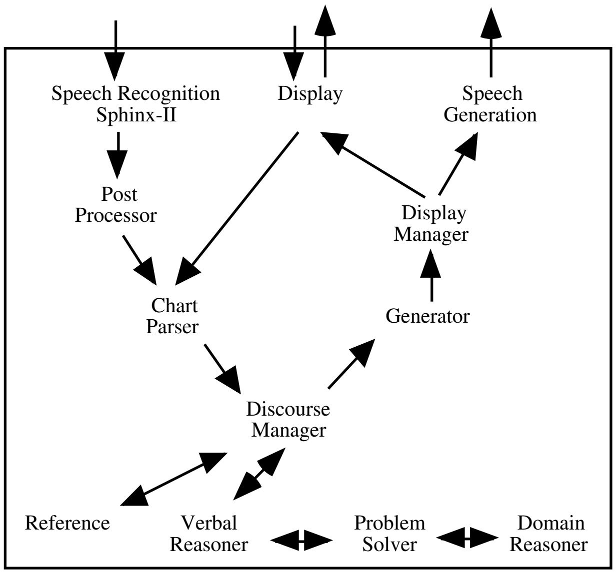

Fig. 1. TRAINS-95 System Architecture

2 The System

The domain in TRAINS-95 is simple route planning. The user is given a map on a screen showing cities, connections and the locations of a set of trains (see Figure 2), and a specification of a set of destination cities where trains are needed. The task is to plan routes to take the trains from the initial locations to the destinations. The route planner used by the system is deliberately weak so that interaction is needed to find good plans. Specifically, the planner cannot find routes longer than four hops without an intermediate city, and when it can generate a route, it randomly selects among the possibilities.

The TRAINS-95 system is organized as shown in Figure 1. At the top are the I/O facilities. The speech recognition system is the SPHINX-II system from CMU. The speech synthesizer is a commercial product: the TRUETALK system from Entropics. The rest of the system was built at Rochester. The display supports a communication language that allows other modules to control the contents of the display. It also handles keyboard input. The speech recognition output is passed through a statistical error-correcting post-processor. The parser, accepts input either from the post-processor (for speech) or the display manager (for keyboard), and produces a set of speech act interpretations that are passed to the discourse manager. The discourse manager is itself composed of a range of subcomponents handling reference, speech act interpretation and planning (the verbal reasoner), and the back-end of the system: the problem solver and domain reasoner. When a speech act is planned for output, it is passed to the generator, which constructs a sentence and passes it to both the speech synthesizer and the display. The generator is a simple template-based system. It uses templates associated with different speech act forms that are instantiated with descriptions of the particular objects involved. The form of these descriptions is defined individually for each class of objects in the domain.

In order to stress the system in our robustness evaluation, we used the ATIS language model provided from CMU rather than a model based on TRAINS data. This system yields an overall word error rate of 30%, as opposed to a 20% error rate that we can currently obtain by using language models trained on our TRAINS corpus. While this accuracy rate is significantly lower than often reported in the literature, remember that most speech recognition results are reported for read speech, or for non natural-dialogue applications such as ATIS. Natural dialogue systems encourage a more spontaneous form of speech that is much more difficult to recognize.

To illustrate the typical complexity of spoken interactions, an example spoken dialogue from the experiment is given below. In the example scenario, the trains were initially at Charlotte, Atlanta and Cleveland and needed to be moved to Boston, Chicago and Montreal. In the transcript, we show the input as it was produced by the speech recognizer in ALL CAPS followed by what was actually said in italics. When the recognition is perfect, we just show the speech recognition output. The dialogue has a 82.2% word recognition accuracy.

SYS: OK, I think I'm ready to start.
USR: THE OKAY LET'S STEAK THE TRAIN FROM ATLANTA AT TO CHICAGO VIA LEXINGTON IN INDIANAPOLIS
Okay, let's take the train from Atlanta to Chicago via Lexington and Inidanapolis.
SYS: Yes
SYS: City Cincinnati is congested due to unusually heavy traffic.
SYS: City Columbus is congested due to unusually heavy traffic.
SYS: An additional 5 hours will be needed to travel through them.
USR: OKAY LET'S GO VIA INDIANAPOLIS
SYS: OK
USR: LIST STEAK THE TRAIN FROM CHARLOTTE TO BOSTON
Let's take the train from Charlotte to Boston.
SYS: I need help choosing a route from Charlotte to Boston.
USR: GO VIA RICHMOND AND PHILADELPHIA
SYS: Yeah
USR: OKAY LET'S GO FROM CLEVELAND TO MONTREAL
SYS: Yep
SYS: The terminal at City Buffalo is delaying traffic due to localized heavy winds.
SYS: Trains will take an additional 5 hours to move through there, due to decreased visibility.
USR: GO VIA DETROIT
SYS: Yeah
USR: AND THAT
I'm done.
SYS: Excuse me?
USR: I'M DONE
SYS: Please confirm that you are done, using the dialogue box.

3 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.

While our evaluation appears similar to HCI experiments on whether speech or keyboard is a more effective interface in general (cf. [6]), this comparison was not actually our goal. Rather, we were using the various input media as a way of manipulating the input error rate. While keyboard performance is not perfect because of typographical errors (we had a 5% error rate on keyboard input during the evaluation), it is considerably less error prone than speech.

In addition to the differences in "word recognition accuracy" between keyboard and speech input, word recognition accuracy was further 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 [8].

The routing tasks were chosen with the following restrictions to ensure non-trivial dialogues and avoid drastic complexity variation among tasks:

- Each task entailed moving three trains to three cities, with no restriction on which train was destined for which city.
- In each scenario, three cities had conditions causing delays.
- One of the three routes in each scenario required more than four time units ("hops") to travel.

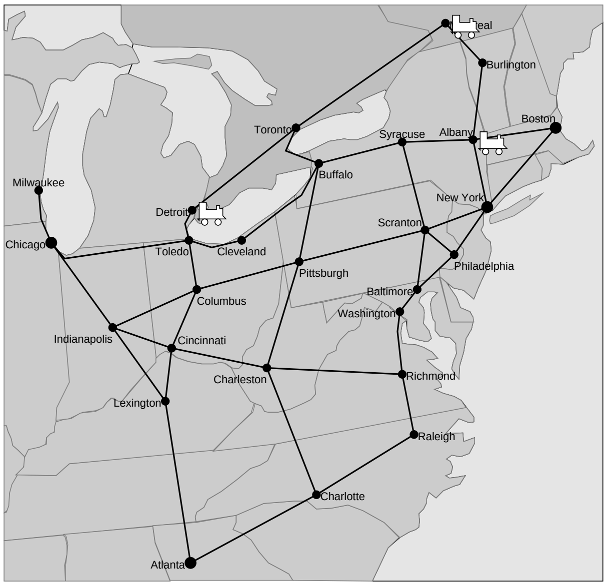

Fig. 2. TRAINS-95 Map

3.1 Experimental Environment

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.

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

- A Sphinx-II speech recognizer developed at CMU [5], running on a DEC Alpha
- TRAINS-95 version 1.3 and the speech recognition post-processor running on a SPARCstation 10
- TrueTalk, a commercial off-the-shelf speech generator (available from Entropics, Inc.) running on a SPARCstation LX

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 transcripts of what the subject actually said to transcripts of parser input from the speech recognition module and 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 dialogue 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.

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.

3.2 Procedure

The four phases of the experiment are outlined below.

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 to speak to the system through the examples in the tutorial, the subject was given no explicit direction about what could or could not 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 [10].

Since it was important for the subject to understand how solution quality 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 were charged if the planned route included cities experiencing delays or if different train routes interfered with one another. (The system informs the user if either of these situations exist.)

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. 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, a separate window displayed the output from the speech recognition module in textual format, indicating what 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 an 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 additional information about cities to be avoided. The subject didn't know the initial location of the trains until the map with the initial configuration was displayed on the monitor. 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 and the speech recognition module's output was not displayed.

Questionnaire After completing the final task, the subject was given a questionnaire that solicited the impressions about the cause of any difficulty encountered, reasons for selecting speech or keyboard input for the final task, and recommendations for improvements.

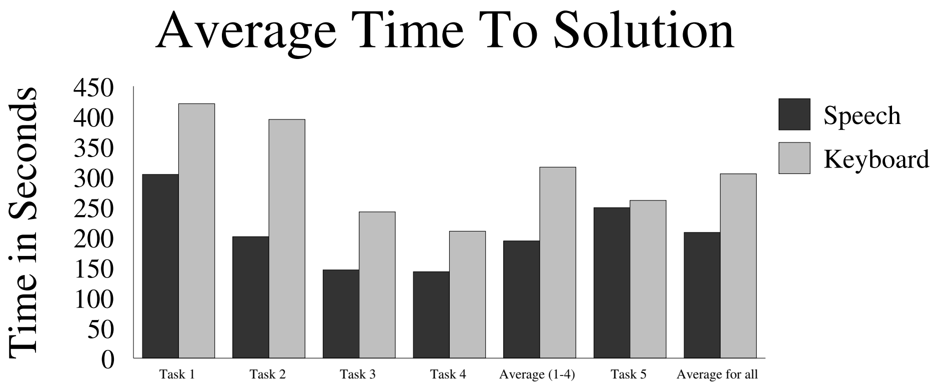

Fig. 3.

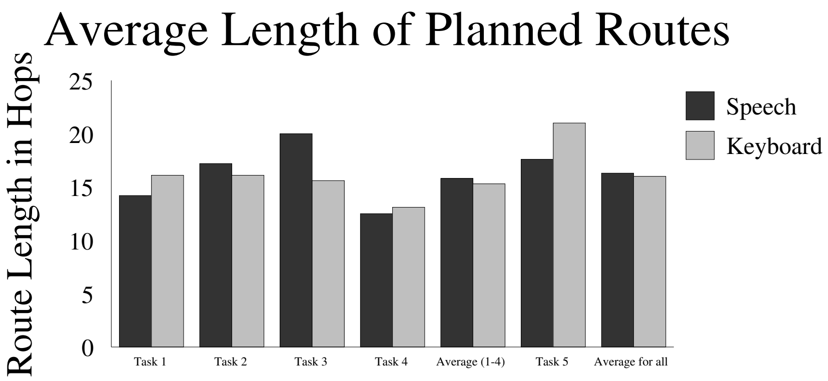

Fig. 4.

4 Experiment Results

Results of the experiment relevent to our hypotheses are as follows:

- 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.

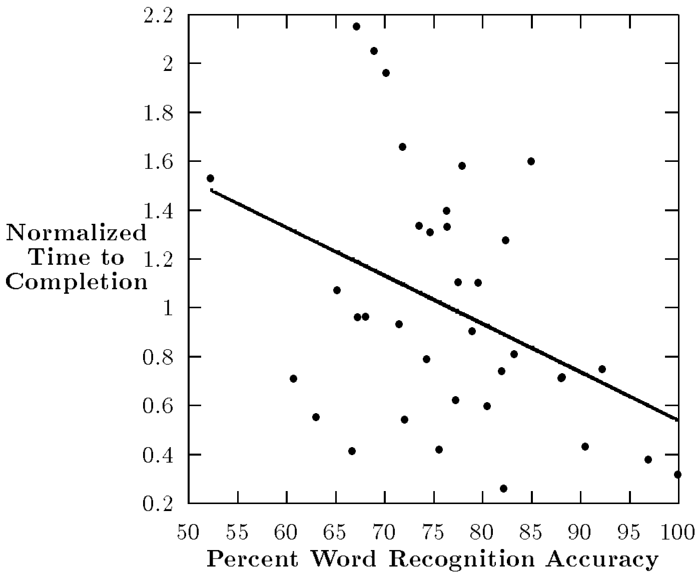

Fig. 5.

- 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 [9]. 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 [3].
- Figure 5 plots word recognition accuracy versus the time to complete the task for each of the 5 tasks.

Note that in Figures 3 and 4, we give the average for Tasks 1-4, as well as the average for all 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.

5 Discussion

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. [11]), and a robust system that proceeds based on partial understanding - showed that the robust system was significantly more successful in completing the same task [7].

– Random Routes

  Designers of the TRAINS-95 system feared 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.

6 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 [2].

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

1. J. F. Allen, G. Ferguson, B. Miller, and E. Ringger. Spoken Dialogue and Interactive Planning. In *Proceedings of the ARPA SLST Workshop*, San Mateo California, January 1995. Morgan Kaufmann.
2. M. Boros, W. Eckert, F. Gallwitz, G. Görz, G. Hanrieder, H. Niemann. Towards Understanding Spontaneous Speech: Word Accuracy Vs. Concept Accuracy. In *Proceedings of the International Conference on Spoken Language Processing*, Philadelphia, Pennsylvania, October 1996.
3. P. Cohen and S. Oviatt. The Role of Voice Input for Human-Machine Communication. In *Proceedings of the National Academy of Sciences*, 1994.
4. 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*, Princeton, New Jersey, March 1993. Morgan Kaufmann.
5. X. 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.
6. S. Oviatt and P. Cohen. The Contributing Influence of Speech and Interaction on Human Discourse Patterns. In J. W. Sullivan and S. W. Tyler (eds), *Intelligent User Interfaces*. New York, New York. 1991. Addison-Wesley.
7. 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*, Harriman, New York, February 1992. Morgan Kaufmann.
8. E. Ringger and J. F. Allen. Error Correction Via A Post-Processor For Continuous Speech Recognition. *Proceedings of ICASSP-96*, Atlanta Georgia, May 1996.
9. A. Rudnicky. Mode Preferences in a Simple Data Retrieval Task. In *Proceedings of the ARPA Human Language Technology Workshop*, Princeton, New Jersey, March 1993. Morgan Kaufmann.
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 Natural Language Workshop*, Harriman, New York, February 1992. Morgan Kaufmann.
11. R. Smith and R. D. Hipp. *Spoken Natural Language Dialog Systems: A Practical Approach*, Oxford University Press. 1994.
12. S. Walter. Neal-Montgomery NLP System Evaluation Methodology. In *Proceedings of the DARPA Speech and Natural Language Workshop*, Harriman, New York, February 1992. Morgan Kaufmann.

This article was processed using the LaTeX macro package with LLNCS style