A wizard of oz framework for collecting spoken human-computer dialogs.

Conference Paper · January 2004

Source: DBLP

CITATIONS
20

READS
126

10 authors, including:

Elizabeth Shriberg
Microsoft
214 PUBLICATIONS **9,717** CITATIONS
SEE PROFILE

Fuliang Weng
Bosch Research and Technology Center North …
67 PUBLICATIONS **851** CITATIONS
SEE PROFILE

Stanley Peters
Stanford University
124 PUBLICATIONS **2,795** CITATIONS
SEE PROFILE

Lawrence Cavedon
RMIT University
121 PUBLICATIONS **1,379** CITATIONS
SEE PROFILE

A Wizard of Oz Framework for Collecting Spoken Human-Computer Dialogs

Hua Cheng[1], *Harry Bratt*[2], *Rohit Mishra*[3], *Elizabeth Shriberg*[2], *Sandra Upson*[4], *Joyce Chen*[3]
Fuliang Weng[4], *Stanley Peters*[1], *Lawrence Cavedon*[1], *John Niekrasz*[1]

[1]Center for the Study of Language and Information, Stanford University, Stanford, CA
[2]Speech Technology and Research Lab, SRI International, Menlo Park, CA
[3]Electronics Research Lab, Volkswagen of America, Inc., Palo Alto, CA
[4]Research and Technology Center, Robert Bosch Corp., Palo Alto, CA
harry,ees@speech.sri.com; huac,peters,lcavedon,niekrasz@csli.stanford.edu
rohit.mishra,joyce.chen@vw.com; sandra.upson,fuliang.weng@rtc.bosch.com

Abstract

This paper describes a data collection process aimed at gathering human-computer dialogs in high-stress or "busy" domains where the user is concentrating on tasks other than the conversation, for example, when driving a car. Designing spoken dialog interfaces for such domains is extremely challenging and the data collected will help us improve the dialog system interface and performance, understand how humans perform these tasks with respect to stressful situations, and obtain speech utterances for extracting prosodic features. This paper describes the experimental design for collecting speech data in a simulated driving environment.

1. Background

Research in human-computer interfaces has been carried out in applications where the user is focused on tasks such as driving a car [4] or operating other machinery, with the goal of designing interfaces that will help reduce the user's overall cognitive load. In such applications, the user normally controls several devices simultaneously. Existing applications maintain little or no dialog context, and require the user to learn and remember complicated sets of device-specific commands. To overcome some of the shortcomings of such systems, researchers have been investigating designing spoken interface systems which can converse with the user more naturally, allowing more flexibility in the user's speech and keeping track of the dialog context, similar to how a human speech partner would [6, 7]. However, human-human speech in such scenarios is highly context- and situation-dependent, full of disfluencies (e.g., false starts and pauses) and sentence fragments (abandoned or repaired utterances), and is highly interactive and collaborative. We believe that the easiest interfaces to use will be those that mimic human-human interaction in some, though perhaps not all, respects. Therefore, our data collection focuses on collecting the kind of speech that would occur between a human and a system that is as flexible and capable as that user would desire.

Our goal is a system should mimic human-human interactions by understanding the user's requests and producing responses based on the user's knowledge, the conversational context, and the external situation. We use the car-driving domain as a testbed of such a dialog interface for operating in-car equipment, such as obtaining navigation information (e.g., turn-by-turn instructions) and information about local points of interest. Figure 1 illustrates the system components, which include a language understanding component, a response generator, a dialog manager and a prosody classifier. We use off-the-shelf technologies and tools for speech recognition, speech synthesis, and knowledge management.

1.1. Purposes for Data Collection

The ultimate goal of our dialog system is to enable natural interactions between the driver and the system to be like those between humans. Therefore collecting human-human dialogs for the above tasks helps us to develop and tune the system to simulate such interactions. As the first step of our system development, data collection has the following specific purposes.

Improve the system interface and performance: Language coverage has been a bottleneck for existing dialog systems. A robust dialog system should allow the user to speak freely and be able to understand the user's intention expressed through various utterances. The robustness of a system can only be enhanced using a large amount of data that are expected to cover most language phenomena in the target application. Therefore we aim to collect dialogs from many subjects and to use these data to train the language understanding component. The data will also provide evidence as to what features users would desire in an in-car conversational system.

Understand how humans give navigation instructions in a driving situation: Although human navigation data has been collected for developing systems that automatically generate navigation instructions, e.g. [1], the data is often written descriptions based on the subject's mental recap of the route. In a driving environment, humans might choose to give navigation information differently with respect to the current position of the vehicle (e.g., close to a turn) and external situations (e.g., emergency stop). There is a need, therefore, to collect new data to discover what kinds of strategies humans would use to convey navigation information in a real-time setting.

Obtain speech utterances for extracting prosody features: Drivers are likely to produce disfluent and distracted speech with potentially complex syntax when focusing on tasks other than talking. Such data contain rich prosodic information that captures variations in timing (e.g., lengthened sounds, pauses), intonation (e.g., pitch rise/fall at the end of utterance), and loudness. These features convey information beyond that carried by the words themselves. They can help a dialog system detect utterance boundaries, driver intention and stress level, and subsequently generate appropriate responses which take into account the driver's emotional state. They can also help augment the information available to a natural language parser, to help

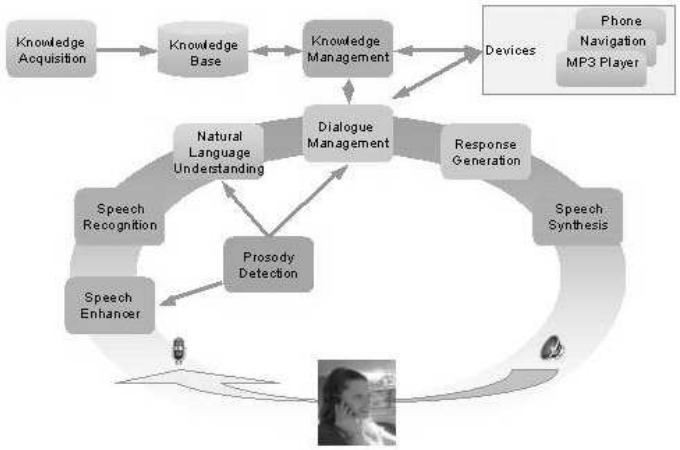

Figure 1: The in-car dialog system components

it handle disfluencies and multiple-sentence utterances, and to provide information such as sentence-level intonation pattern or word emphasis.

In addition to features in the driver's speech, the status of in-car devices such as the steering wheel, pedal and brake can also tell us if the driver is involved in attention-demanding activities, such as emergency braking or sharp turns. Such activities increase the cognitive load on a speaker, and we would therefore expect them to increase the disfluencies in a driver's speech. In addition, data from these devices may be useful in predicting the driver's stress level, attentiveness, or other factors that could be relevant information to the dialog component.

In summary, data collection will help improve the performance of the language understanding, response generation and prosody detection functionalities of our in-car dialog system.

1.2. Related Work

Although a number of annotated corpora have been made available for research purposes, such as those published by the Linguistic Data Consortium, they cover limited domains. There are only a handful of efforts in collecting data for in-car applications, mainly for acoustic modeling. [4, 5] describe a process for collecting Japanese speech corpora in moving car environments for research in robust ASR and spoken dialog systems under high-noise conditions. Those studies collected speech data in both idling and driving situations. The spoken dialog corpus consists of the subject talking to an operator who rides with them to answer their queries and navigatate them through a driving route. The DARPA CU-Move project[1] records speech in real driving environments using similar methods. The subjects are asked to say direction phrases, numbers, and street locations as well as phonetically balanced sentences prompted from several predetermined sets. They are also asked to use an online navigation system where a human "wizard" will guide them through various routes.

None of the existing data, however, address the needs raised above. Therefore we set up experiment rooms to collect data for developing our in-car dialog system. In the rest of this paper, we describe the experimental design that enables us to collect dialogs we are interested in.

[1] http://cumove.colorado.edu/cumove-Phase2page.htm

2. Data Collection Setup

We adopted a "Wizard of Oz" (WOz) approach [2], where the subject talks to what appears to be an automatic system, but the system's responses are in fact generated by a human (the "wizard") in another room. This approach allows eliciting high quality dialog while setting appropriate expectations for the subject in terms of language complexity, because humans usually use simpler language when talking to a machine, and therefore avoids the need for automatically understanding unconstrained language. This also filters out the conversations irrelevant to in-car tasks which may occur in human-human interactions.

2.1. Driving Simulator

We use a driving simulator to safely allow us to elicit emotions such as frustration and states such as "rushed."

As a part of the simulator, we use Microsoft's Midtown Madness 1® driving game on a Windows™PC, which is set in downtown Chicago and has a variety of road types such as highway, city streets (including one-way streets) and tunnels. The game can be configured with respect to the amount of traffic, pedestrians, etc. We mount a steering wheel in front of the game display. There are six buttons on the wheel, one of which is used to put the car into reverse and the rest as push-to-talk (PTT) buttons (explained below). The wheel has force feedback to more closely simulate a driving experience. There are gas and brake pedals on the floor below the wheel.

We set up two divided rooms, the subject room and wizard room, so that the subject and wizard do not see each other, as shown at the left and right of Figure 2. One experimenter acts as the wizard and the other as the administrator in the subject room to monitor the progress of the experiment and resolve any problems the subject may have. The two experimenters swap roles for each successive subject. The wizard acts as an ideal dialog system and the subject interacts with the wizard without knowing or seeing her. The wizard can see the game screen through a video splitter so she knows the subject's current location and heading.

When subjects need navigation or entertainment information, they press a PTT button on the wheel and talk into a close-talking microphone. The wizard only hears the subject's speech when PTT is pressed. This barrier is set up to reduce the work load of the wizard by not requiring her to judge whether the subject is talking to her and whether she needs to reply. The subjects can also press the PTT button to cancel the wizard's speech when they want to interrupt and start their new query. The administrator will endpoint the subject's utterances, so the subject will not need to hold down the PTT button while speaking or have to remember to press the button again after speaking.

2.2. Speech and Event Recording and Routing

Our data collection software has three main components that run on three separate computers. The subject, the wizard and the administrator are each seated at a three computer. Audio is generated at each computer station, and the audio input at each station needs to receive its own specific mix of the audio output (e.g. the recorder needs to receive the subject's microphone and the wizard's output, while the subject needs to hear the game sounds and the wizard's output). This is achieved by routing all audio through a 12-channel 4-bus analog mixer and using the mixer's controls to adjust levels and route the audio to the correct combination of its six outputs. The components running on each of the three computers are:

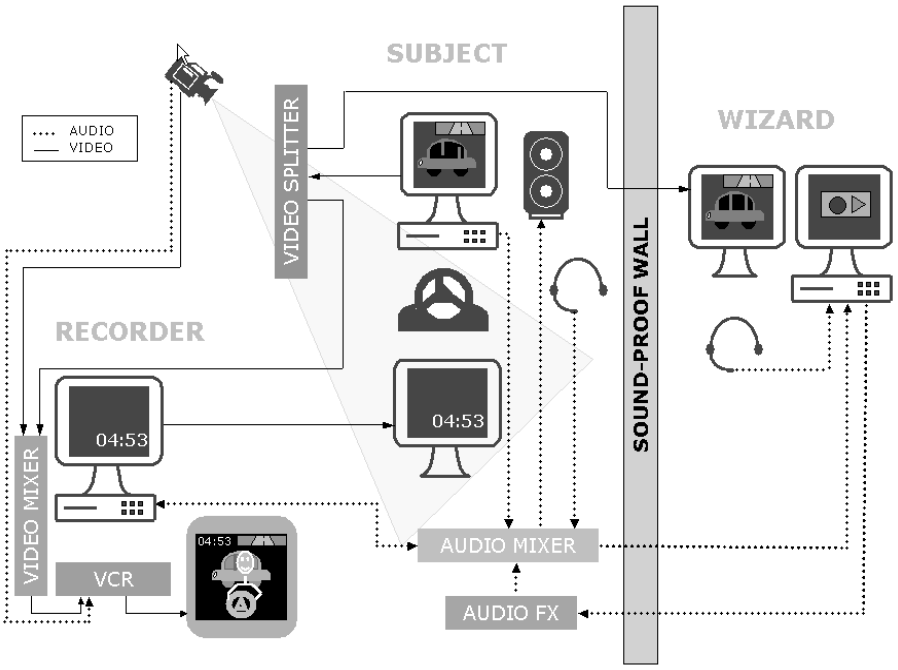

Figure 2: The data collection room setup

Recorder runs on the experimenter's computer in the subject room to provide a central GUI for recording the experiment. It samples subject and wizard speech at 16kHz and records stereo audio with subject and wizard on separate channels. It receives wheel and pedal events from the software running on the driving simulator through a socket connection, and records them to the hard disk. All speech and events are timestamped. Recorder has two additional functions: 1. Route the subject's audio to the wizard when PTT is pressed, and mute the audio output when the subject speech ends; 2. Let the administrator endpoint subject utterances.

Simulator runs on the same computer as the driving game. Every 1/100 second it samples the same wheel, button, and gas and brake pedal events that are used by the driving game, and sends this event data to the Recorder.

WizardRecorder and SentenceSelector run on the computer in the wizard room. SentenceSelector helps the wizard quickly choose a response from a finite list of pregenerated Text-To-Speech (TTS) utterances by typing only a few salient words. As words are typed, they autocomplete, and as they are entered, the list of sentences available is shortened to only include those that contain all words entered.

WizardRecorder allows the wizard to record her voice before sending it to the subject in order to self-monitor for disfluencies. This reduces the risk of exposing the wizard as a human. The wizard is instructed to speak "formally" as opposed to how she would in a human-human conversation, and to use a speaking style similar to the TTS voice. On playback, WizardRecorder randomly removes short segments of recorded speech at zero-crossings, to "mess up" the timing of the wizard's utterance, making it sound more machine-like.

Both wizard and TTS voices are processed through a digital audio effects box, set to its "lo-fi" filter program, which downsample the audio and lowers the bits per sample, producing an effect similar to a band-pass filter. This has the effect of introducing a machine-like "sonic signature" so that the TTS and wizard voice sound similar to each other, and is intended as another way of keeping the wizard from being thought of as a human. The reason for using both TTS and a human voice is discussed at the end of Section 2.3.

We also videotape the entire experiment session to help understand any problems in the session. We capture a close-up front view of the subject's face and hands, as well as the session timestamp displayed on a monitor positioned behind the subject. This signal is mixed with that of the game display through a video mixer ("dissolve" mix), and is recorded by a VCR.

2.3. Scenario Definitions

In this data collection, we focus on the tasks of navigation and operating in-car MP3 players while driving. The dialog-enabled MP3 player allows the driver to use natural language to query and operate the player in many ways, such as inquiring about albums, songs and artists, etc., creating song lists, adjusting the volume and even performing multiple tasks simultaneously.

Appropriate scenarios are essential for eliciting the desired types of dialogs and emotions from the subject. We have one scenario that asks the subject to rush a target location so as to simulate a stressful situation. We also use two relaxed scenarios for comparison, in which the subject simply drives from their current location to a target location. We expect differences in prosodic features in the subject's speech in stressful and non-stressful scenarios. Such comparison requires the scenarios to be tightly controlled, that is, they should have similar distances and contain similar road types, even similar numbers of turns.

We call these the Controlled Scenarios. We alternate the ordering of these scenarios to balance the effect of differences in individual scenarios.

Another factor that might have unwanted effects is variation in wizard speech, so the wizard uses SentenceSelector to choose from a finite set of navigation responses by typing key words and the selector automatically completes the sentences. The set contains a few generic responses, such as "I don't have an answer for you", that can catch unexpected queries.

The most difficult problem we are facing is to elicit speech from the subject for modeling dialog and extracting prosody information. Since the subjects have not interacted with such a system before, they are not sure what they can say to the system. Very often they expect the system to give them navigation information without their talking. Therefore, we ask the wizard not to take initiative in the dialog, to encourage subjects to speak when they do not know where to go. Another approach we use is to have the wizard guide the subjects to blocked roads or one-way streets. When the subjects find that they cannot drive through, they will seek more information from the system.

It is also possible to give hierarchical descriptions of routes rather than turn-by-turn instructions, e.g., "first go to 101". If the subject does not know how to implement an instruction, they will have to ask the system. This strategy is often used by humans [3]. However, it only works when the driver has some local road knowledge. Since our subjects do not have prior knowledge about the roads as laid out in the game (which do not map exactly to reality), we cannot use this strategy.

On the downside, the controlled scenarios might elicit subject speech that would not happen naturally in a real driving situation, because the navigation system is expected to give driving instructions whenever necessary. Therefore we designed another navigation scenario (called Open Scenario) where the wizard can speak freely and take initiative whenever she thinks necessary. This scenario collects data on how humans give navigation instructions given the subject's driving situation. Since we have two experimenters, we collect instructions from both of them to reduce personal bias.

The last scenario we use is for operating the MP3 player. The subjects are asked to create two song lists while they are driving around the city in a non-stressful mode. To create the song lists, the subjects may ask about songs in their collection, listen to songs, and add to or delete songs from their lists. This scenario provides multi-threaded dialogs.

Ideally both controlled and open scenarios should use TTS voices to disguise the wizard's identity. However, the open scenarios are more sophisticated and it is impossible to foresee all possible human utterances without using some very general answers. Therefore we use TTS voices for the controlled scenarios and human voices for the open scenarios, and route the speeches through an audio box to make them sound similar.

In addition, we have a short initial scenario for subjects to get familiar with the experiment, including the handling of wheel and pedals and how to interact with the dialog system to get a sense of what to say and how the system responds.

3. Data Collection Status and Future Work

We have run 12 of the first 20 pilot subjects, and recorded speech and video. Transcription of the pilot data has not yet begun, though we have transcribed pre-pilot data. The pre-pilot sessions were used for debugging the system setup and training the wizards. Data from those sessions will not go into our corpus, but it does show some interesting phenomena that we expect to see occur in the pilot data, especially in terms of disfluent and distracted speech when the subject is focused on driving, for example: "sorry could . . . could you stop listing the uh . . . the mp3s?" The syntactic structure of the utterances can potentially be complex, and understanding such utterances is made even more challenging by background noises (i.e., game sounds and music being played), for example, "I want to um . . . just see some of the sights, so, like uh . . . old town, downtown, Chinatown . . . uh, could you . . . just . . . sort of give me a tour of the city". We also find interesting navigation patterns in the wizard's speech, for example, using landmarks and junction shapes to identify an intersection (e.g., "Follow this road for a few blocks. After you pass the John Hancock Center, make a left."), and giving multiple instructions for a turning point based on the position of the car (e.g., after the last example, "Turn left here" when the subject is approaching the intersection).

We are in the process of collecting more data and transcribing the audio data using Praat[2]. The transcription will then be annotated with lower level features such as punctuation, utterance boundaries, and disfluencies, and higher level features such as phrase structures, prosodic information, dialog acts, and message types [1]. The wheel and pedal events will be automatically extracted from the log, and time-aligned to the waveform. We plan to make the audio data, transcriptions and annotations, and video data if desired, available through the Linguistic Data Consortium once collection is complete.

4. Acknowledgments

This project is partially supported by NIST ATP funding, Robert Bosch Corp., and VW of America.

5. References

[1] Dale, R., Geldof, S. and Prost, J. "CORAL: Using Natural Language Generation for Navigational Assistance", *Proceedings of the 26th Australasian Computer Science Conference*, Oudshoorn, M. (Ed.) Adelaide, Australia, 2003.

[2] Fraser, N. and Gilbert, N.S. "Simulating speech systems", *Computer Speech and Language*, 5:81-99, 1991.

[3] Hook, K. *An Approach to a Route Guidance Interface*, Licentiate Thesis, Dept. of Computer and Systems Sciences, Stockholm University, 1991.

[4] Kawaguchi, N., Matsubara, S., Iwa, H., Kajita, S., Takeda, K., Itakura, F. and Inagaki, Y. "Construction of Speech Corpus in Moving Car Environment", *Proceedings of 6th International Conference on Spoken Language Processing (ICSLP 2000)*, vol.3, 362-365, Beijing, China, 2000.

[5] Kawaguchi, N., Matsubara, S., Takeda, K. and Itakura, F. "Multimedia Data Collection of In-Car Speech Communication", *Proceedings of 7th European Conference on Speech Communication and Technology (Eurospeech 2001)*, pp. 2027-2030, 2001.

[6] Larsson, S. and Traum, D. "Information State and Dialogue Management in the TRINDI Dialogue Move Engine Toolkit", *Natural Language Engineering*, 6(3-4), 2000.

[7] Lemon, O., Gruenstein, A. and Peters, S. "Collaborative Activities and Multi-tasking in Dialogue Systems", *Traitement Automatique des Langues (TAL)*, 43(2), 2002.

[2] http://www.fon.hum.uva.nl/praat/