Shaping Spoken Input in User-Initiative Systems

Stefanie Tomko and Roni Rosenfeld

Language Technologies Institute, School of Computer Science
Carnegie Mellon University, Pittsburgh USA
`{stef, roni}@cs.cmu.edu`

Abstract

A spoken dialog system performs best when users speak within the grammar that the system understands. We conducted a simple study to investigate how easily users can be persuaded to speak to a system using a restricted, less-than-natural-language input style. In a Wizard-of-Oz setting, users of a spoken dialogue system for information access were given brief instructions to "speak simply" to the system. During their interactions, conversational or complex input was rejected by the system while simpler, "just-the-facts" input was accepted. We found that all users were able to adapt their language to successfully complete tasks, and participants' post-experiment comments showed that they were consistently mindful of the form of acceptable input. These results will support further investigation into more precise shaping of user input, leading to more effective and efficient human-machine speech interaction.

1. Introduction

A central issue with spoken dialog systems is the grammaticality problem: getting users to speak within the bounds of the system's grammar. This can be a problem from both the user's and the developer's perspective. For users, it can be difficult to determine what the grammatical bounds of a system are, especially when the system is advertised as a conversational, natural language system. For developers, it can be challenging to create grammars that accurately reflect what users are likely to say to a system, and also to map such input to the user's intended *meaning*.

To reduce this issue on the development side, we have created Speech Graffiti [1]: a structured, subset language interaction protocol for interacting with simple machines. Both user input and system output are standardized in Speech Graffiti. User input takes the form of *slot+value* pairs or *what*-questions, such as "theater is the Galleria Six" or "what are the theaters?" Recognized input is confirmed by the system as a terse restatement (currently, this is simply the *value* of the pair that was entered: "`Galleria Six`"). Speech Graffiti is also designed to be a user-initiative system. The system does not prompt the user for input; instead, it confirms and acts on recognized input and signals confusion when it hears something it does not understand. This allows for quick interactions, since users do not have to listen to lengthy prompts or navigate through menus to get to a specific function.

Although we have shown that Speech Graffiti can be an effective interface [1], users often have trouble speaking within the bounds of its subset language grammar [2]. Our current research focuses on determining what aspects of natural language are easily avoided by users, and what aspects of the Speech Graffiti grammar should be relaxed, in order to make human-machine speech interactions more efficient and effective.

1.1. Related work

In a previous study comparing Speech Graffiti and natural language interfaces to the same movie information database [2], we noticed a difference in the scope of natural language constructions. After considering items like movie and theater names to be equivalence class members, the utterances used by participants when speaking to the natural language MovieLine reduced to about 580 patterns. In contrast, with the Speech Graffiti MovieLine, when users spoke outside the Speech Graffiti grammar and used natural language instead, their utterances reduced to only 94 patterns. One of the main differences between the patterns in the two systems was the lack of conversational phrases like "can you give me…" and "I would like to hear about…" when speaking to the Speech Graffiti system. Thus the knowledge that they are interacting with a restricted language system seemed to be enough to make users speak more simply.

This resembles results from Ringle and Halstead-Nussloch [3], in which users produced simpler input when interacting with a system that responded with formal, state-transparent responses instead of more conversational, natural language output. It has been shown that users of spoken dialog systems also adapt their lexical choices to match system vocabulary ([4], [5]), and Shriberg, Wilder and Price [6] found that users tend to simplify their input style when high speech recognition word-error rates occur.

Zoltan-Ford [7] found similar adaptation results in a Wizard-of-Oz study (in which, unbeknownst to participants, a human operator plays the role of the computer) involving speech and text interaction with an inventory system. In this study, half the participants were unknowingly assigned to a restricted-vocabulary condition in which the system would only accept input that was lexically identical to the system's output (with minimal syntactic transformation). She found that users adapted the length of their input to match that of the system, and that *explicit* shaping—by rejecting input that did not match the grammar in the restricted-vocabulary condition—was more effective in influencing user input than simply relying on users to model system output naturally.

As part of our research on influencing user input in order to support more efficient interactions, we are interested in further exploring the phenomenon of shaping. As a preliminary step, we were interested in determining what effect different instructions about a system's limited capabilities might have on user input, and how users would shape their input in response to *a*) the rejection of conversational, natural language input and *b*) Speech Graffiti-style confirmation. In contrast to Zoltan-Ford's experiment, in which the system prompted the user for input at every turn, we were interested in seeing how shaping would function in a user-initiative system. In order to factor out the effect of speech recognition errors, we designed a Wizard-of-Oz experiment to assess the effect of shaping factors on user input to a telephone-based information access system

2. Method

2.1. Participants

We recruited 18 participants from the Carnegie Mellon University community to participate in this study. Seventeen of the participants were students (15 undergraduates, two graduates) ranging in age from 19-26 years old. The final participant was a 55-year old adult working at the university. Only two of the participants reported that they considered themselves to be “computer science or engineering people;” the rest came from a variety of non-technical fields such as drama and history. Most participants had encountered speech recognition systems before but did not use them on a regular basis. Eleven participants were female; seven were male. All participants were paid eight dollars for their time.

2.2. Procedure

The participants came to our lab for the study and were told that they would be talking to “a system that understands your speech and allows you to get information about two different kinds of things: movies playing around town, and airline flight times.” Each participant was given a list of ten scenarios and asked to call the system via a telephone in the lab to find out the information required by each scenario. Three of the ten scenarios are shown in Fig. 1. When the user was ready to begin interacting with the system, the experimenter moved to a cubicle in the back of the lab to perform the wizard role.

A. A friend told you that *Miracle* was pretty good. Where is this movie playing?

B. A friend has told you that she’s flying to San Francisco on United flight 500. When will she get there?

C. You need to fly from Pittsburgh to Minneapolis on Saturday. You prefer to fly on USAir, and you’d like to get the earliest flight possible.

Figure 1: Sample tasks from the Wizard-of-Oz study.

2.3. Instruction conditions

When participants called the system, the system first played a brief introductory statement consisting of four parts: a welcome, an instruction, an example and a prompt. The instruction had three variations: short, medium, and long, while the other components were the same for all users. Fig. 2 shows the introductory message with the three instruction variations. Note that the instruction variations are additive; each longer instruction includes all the information from the shorter versions. Each participant was randomly assigned to one of the three instruction conditions.

2.4. Wizard-of-Oz interaction

The telephone used by participants was connected to the wizard’s cubicle via a listening device so that the wizard could hear what users said to the system. With the help of a form-based interface (Fig. 3), the wizard selected a response for each user utterance. Each response was then synthesized and played over the phone to the user.

welcome: `Welcome to the InfoLine.`

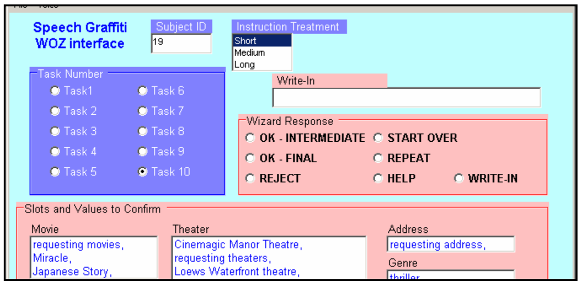
instruction-<u>short</u>: `The system you are talking to only understands very simple English, so please speak to it as simply as you can.`

instruction-<u>medium</u>: `The system you are talking to only understands very simple English, so please speak to it as simply as you can. It will understand you best if you tell it only one idea at a time.`

instruction-<u>long</u>: `The system you are talking to only understands very simple English, so please speak to it as simply as you can. It will understand you best if you tell it only one idea at a time. This system understands only keywords, and not the structure of sentences.`

example: `For instance, you might say "movie The Lord of the Rings,” or “airline is United,” or “what are show times?"`

Figure 2: System introduction with short, medium and long instruction conditions.



Subject ID: 19

Instruction Treatment: Short (Short, Medium, Long)

Movie: requesting movies, Miracle, Japanese Story,

Theater: Cinemagic Manor Theatre, requesting theaters, Loews Waterfront theatre,

Address: requesting address,

Genre: thriller

Figure 3: Partial screen shot of wizard interface.

The guidelines for the wizard role were to reject user input that contained non-task conversational words (*e.g.* “could you tell me…”), task-based non-content items (*i.e.* those that would be extraneous in a Speech Graffiti *slot+value* pair, like “what movies *are showing* in West Mifflin?”), or task-based vocabulary not in the current Speech Graffiti versions of these database systems (*e.g.* “films” instead of “movies,” “*earliest* flight”).

Three rejection messages were available. On an initial rejection, the system said, “`excuse me?`” On a second consecutive rejection, the system said, “`I’m sorry, I didn’t understand that.`” On a third consecutive rejection, the system replayed the *instruction* and *example* strings played for the user during the introduction, and then reset the rejection counter.

Speech recognition errors were not systematically simulated, but the wizard tried to recreate a least one such error per user session by confirming with an incorrect value (for instance, responding “`United`” when the user said “US Air”).

As in Speech Graffiti, the wizard confirmed values from any partially acceptable input, while ignoring “rejectable”

TASK A: User: Where is *Miracle* movie playing?
Wizard: `Excuse me?`
User: *Miracle* movie
Wizard: `Miracle`
User: Where is it playing?
Wizard: `Excuse me?`
User: Which theaters?
Wizard: `3 theaters: Destinta Theatres Plaza 22, Loews Waterfront Theatre, Northway Mall Cinemas 8`

TASK B: User: United flight 500 to San Francisco, what time does it arrive?
Wizard: `United, to San Francisco, flight 500`
User: Arrival time
Wizard: `Arriving at 12:33 pm`

Figure 4: Sample user – Wizard system interaction.

items. Fig. 4 shows user-wizard interactions for Fig. 1 tasks *A* and *B*; interaction *B* shows the partial-confirmation behavior.

Upon completing the experiment, participants were asked to comment on the system, noting if any aspects of the interaction were particularly difficult or easy and if anything about the interaction was unusual or surprising. They were also asked what they would say if they had to tell a friend how to use the system. Participants were then asked to fill out a short demographic questionnaire, debriefed about the Wizard-of-Oz manipulation, and paid for their time. Each participant spent 20-30 minutes in the lab.

3. Results

The overall task completion rate was 96.1%, with twelve users successfully completing all ten tasks, five users completing nine tasks, and one user completing eight. This is the strongest indicator that users can and do adapt their input, since in our study tasks could not be completed if only conversational, natural language was used.

Users mostly refined their input so that only values were conveyed ("US Air flight, to Minneapolis, on Saturday"); there is a distinct lack of verbs in the transcripts of user input.

We calculated the number of utterances each participant used in their interaction and the mean words per utterance. There was no effect of instruction condition (short-medium-long) on the number of utterances used in a session, but instruction condition did affect the average number of words per utterance: as the instruction condition changed from short to long, user utterances tended to be shorter, as shown in Table 1 (ANOVA, F = 4.24, $p < 0.04$). It is possible that this is due to the extra content of the longer instruction messages, which stress "one idea at a time" and "keywords" (but *cf.* section 3.2. for other details about the effect of instruction content).

Table 1: Average number of words per user utterance by instruction condition.

instruction condition	mean words per utterance
short	4.49
medium	3.36
long	2.98

3.1. Rejections

We analyzed instances in which the Wizard rejected user input and found that, on average, about 22% of each user's utterances were rejected, with a range of 9.4% to 39.2%. The number of rejection messages did not vary by instruction condition.

We can assess the power of rejection messages to shape user input by looking at sequential, or "spiral," rejection instances. As shown in Table 2, there were a total of 123 initial rejection episodes, each started by a rejected input *R*. After receiving an "`excuse me?`" message for input *R*, 50% of the immediately following, same-task user inputs (*R+1*) were accepted by the Wizard (in 9 instances, users switched to a new task). Of the remaining 52 *R+1* utterances that were rejected, 30 of the subsequent *R+2* messages were accepted for a total of 75% acceptable after two rejection messages. Following a third rejection, 73% of *R+3* inputs were accepted, bringing the total number of utterances accepted after at most 3 rejections to 84%. Input *R+4* was acceptable in the two of the three instances in which a fourth rejection message was generated. In the third *R+4* instance and the remaining 15% from Table 2, users switched to a new task.

Table 2: Number of rejection instances needed to shape user input successfully.

sequential rejection instance	# of occurrences	total utterances shaped after this level
1st	123	50%
2nd	52	75%
3rd	15	84%
4th	3	85%

It is interesting to note that shaping occurred despite the lack of content in the first two rejection messages. "`Excuse me?`" could imply that the system had a "hearing" problem rather than an "understanding" problem. Yet users almost always altered their input, repeating it verbatim in only seven of the 123 initial rejection instances.

We suspected that one effect of shaping would be that when users received a rejection message, the *R+N* utterance that was eventually accepted by the system would be shorter in length than the initial, rejected input *R*. This turned out not to be the case. Although one-third of the participants had an average decrease in utterance length when repairing rejected input (for instance, switching from "what airlines could I take?" to "what airlines?"), most participants actually tended to increase their utterance length, some by more than 50%.

This seems somewhat counterintuitive, but the following two issues appear to have contributed to this phenomenon. First, the wizard attempted to reject ambiguous input as much as possible. In the flight information tasks, users had to specify a departure and/or arrival airport, but they often did this ambiguously. For instance, they might just say "Phoenix" instead of "to Phoenix" or "departing from Phoenix." Such input was rejected to see if users would re-specify their input unambiguously, which added length to their input. Second, users sometimes restarted their queries after rejections. We did not specifically try to limit the length of user utterances and therefore accepted input that contained multiple simple phrases. Our final task (Fig. 1 *C*) asked users to find the earliest flight matching some constraints. Mimicking the current Speech Graffiti interface, which does not handle constraints like "early" or "late," we rejected such input to see if users would switch to a more generalized query. They usually did (only one user did not complete this task), but before trying the "query" part again users would often repeat the constraints (as in Fig. 4), which counted as acceptable input.

User: Earliest flight.
Wizard: `Excuse me?`
User: To Minneapolis, from Pittsburgh.
Wizard: `To Minneapolis, from Pittsburgh.`

Figure 4: Example of a user restating constraints after a rejected query.

3.2. User perceptions

One of the most interesting results from this study was users' post-experiment interview comments, which showed that participants were clearly aware of the limited style in which they had to speak to the system in order to complete tasks successfully. The most common themes participants mentioned were the simplification and minimization of input and the use of specific types of words like nouns or keywords (or, more accurately, *key* words).

Five users specifically mentioned simplification ("be very simple with commands," "simplify your input") and eight mentioned using specific types of words: "I knew just to say keywords," "use the subject of sentences," "use mostly nouns." Half of the participants specifically mentioned minimizing the information in their utterances *(*"I knew I had to condense everything into smaller pieces," "don't say the whole thing [at one time]," "use as few words as you can").

These themes parallel the information provided in each of the three instruction conditions. The short condition only tells users to "speak simply," the medium condition adds the minimizing concept of saying only "one idea at a time," and the long condition adds the notion of keywords.

Interestingly, participants' comments on these themes did not exactly match up with their assigned condition. The minimizing idea was only explicitly presented in the medium and long conditions, yet an equal number of users (three) from each of the three conditions mentioned that idea in their comments. The keyword idea was only stated in the long condition, but it was commented on by four short-instruction participants and two medium-instruction participants (as well as two from the long condition). Although most of our results suggest that there were few significant differences between users in the different instruction conditions, the recurrence of these ideas in post-interaction comments suggests that at least the content of the instruction messages was on the right track for describing the system to users

4. Discussion

Our findings show that shaping occurs in spoken dialog systems even when the interaction is user-initiated, with no explicit system prompts to respond to. Users tended mirror the value-only form of confirmation delivered by the system, although they occasionally also followed the example given in the introduction and provided a slot name and a value.

The most challenging issue for users seemed to be what words to use for querying (as opposed to the specification of constraints, where the correct words would be repeated in the system's confirmation). In the post-experiment interviews, one third of the participants commented on there being "no indication of which words to use." When asked what he would tell a friend about using the system, one participant said he would tell her "the actual words the system knows, so she wouldn't have to guess." In the real Speech Graffiti systems, an "`options`" keyword is available to provide this information. However, no one in this study explicitly asked the system for help, suggesting that the system may occasionally want to take the initiative and provide this information.

Although it seemed that rejections and confirmations had more of an effect on shaping user input than the introductory information, the initial descriptive instruction strings fit well with users perception of the system's abilities. The actual form of participants' comments on these abilities can suggest possible ways of conveying this information in a more user-friendly manner. For instance, in Speech Graffiti we tend to use the term *keywords* to refer to domain-independent function words in the system, such as "help" or "scratch that," while users in this study used *keyword* to refer to domain-specific slots like "movie" or "flight."

In the future, we plan to investigate methods for more precisely shaping user input to promote more effective interactions with dialog system. These results will inform the content of the help messages that will be provided to effect such shaping, as well as suggest possible changes to the Speech Graffiti protocol. Further analysis of the corpus of utterances generated by users in this study will also suggest what type of input can be expected from novice users of the system.

5. References

[1] Tomko S. and Rosenfeld, R. "Speech Graffiti vs. Natural Language: Assessing the User Experience." To appear in *Proceedings of HLT / NAACL 2004*, Boston MA.

[2] Tomko, S. and Rosenfeld, R. "Speech Graffiti habitability: What do users really say?" To appear in *Proceedings of the 5th SigDIAL Workshop on Discourse and Dialogue*, Cambridge MA, 2004.

[3] Ringle, M.D. and Halstead-Nussloch, R. "Shaping user input: a strategy for natural language design," *Interacting with Computers*, 1(3):227-244, 1989.

[4] Brennan, S.E. "Lexical entrainment in spontaneous dialog. In *Proceedings of the International Symposium on Spoken Dialogue*, 1996, pp. 41-44.

[5] Gustafson, J., Larsson, A., Carlson, R. and Hellman, K. "How Do System Questions Influence Lexical Choices in User Answers?" In *Proceedings of Eurospeech*, Rhodes, Greece, 1997, pp. 2275-2278.

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

[7] Zoltan-Ford, E. "How to get people to say and type what computers can understand," *Int. J. Man-Machine Studies, Vol. 34*, 1991, p. 527-547.