Effects of Prompt Style when Navigating through Structured Data

Vincent Vanhoucke, W. Lawrence Neeley,
Maria Mortati, Michael J. Sloan, Clifford Nass

Stanford University, Stanford, CA 94305, USA

vincent@vanhoucke.com, wlneeley@stanford.edu, maria@mortati.com,
mike.sloan@stanfordalumni.org, nass@stanford.edu

Abstract: This study examines how the structure of information presented via a speech interface interacts with the choice of a prompting strategy. Participants (N = 60) performed a series of searches with a telephone-based, voice-activated, tree-structured search engine in a 3 (prompt type: multiple choices listed up-front, open-ended prompts with a multiple-choice fallback, or open-ended prompts without any fallback) by 2 (broad vs. deep tree) between-participants experiment. There were significant interactions between the prompt type and tree structure for perceived ease of use and perceived usefulness, as well as for the participants' sense of control, sense of success, and liking of the system. In general, up-front prompts were most desirable for deep trees, while the other two strategies were more desirable for broad trees. Implications for prompt design are presented.

Keywords: voice user interfaces (VUI), prompting strategies, search engines, tree structures

1 Introduction

In recent months, the conjunction of industry's eagerness to leverage the success of the World Wide Web (Web) and a general market trend toward mobile computing has pushed more and more companies toward voice-activating their web services. The use of voice interfaces allows companies to potentially reach all phone and cellular phone users, an audience that represents a much greater percentage of the population than those having computer-based access to the Web.

Search on the Internet moved gradually from being address-based – starting with the gopher information service, akin to the telephone network as we know it – to using tree structures to access hyperlinked data, which became the basis of the Web. Specialized search engines now make use of keyword-based information retrieval techniques to access semi-structured data, but most of what drives information access on the Web still relies on navigating through hyperlinks. The idea that such architecture will eventually make its way back into the telephone network through voice-activated platforms poses some interesting challenges to designers.

Tools like the Voice eXtensible Markup Language (VoiceXML[1]) have made the implementation of voice user interfaces (VUIs) increasingly similar to the programming of HTML-based graphical user interfaces (GUIs). However, the similarities in programming strategies do not extend to design: cognitive differences between reading and listening require different design strategies. As a consequence, VUI design issues are driving an increasing amount of research (Gardner-Bonneau, 1999; Kamm, 1994).

In this paper, we address strategies for permitting users to navigate through large databases via a speech interface. In order to be as independent as possible from the data being searched through, we focused on tree structured navigation schemes, which have already be extensively studied in the GUI world (Larson et al, 1998).

We did not study keyword-based and natural language search strategies, for which measures of precision and recall, as well as the semantics of the information being searched, are likely to strongly affect the results. Recent work (Walker et al, 1998)

[1] VoiceXML Forum: http://www.voicexml.org/

contrasts directed prompting to natural language interactions in the specific context of an email interface.

To simplify the argument, in this experiment, we distinguish between two tree types: *deep trees* and *broad trees*. Deep trees have many levels and a limited number of entries at each level (Figure 1). Broad trees have few levels and a large number of entries at each level (Figure 2).

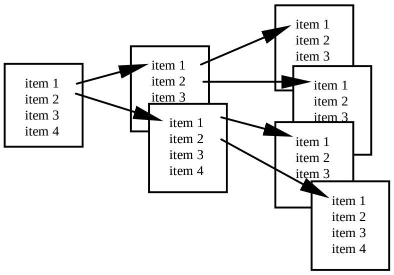

Figure 1: Deep Tree Structure

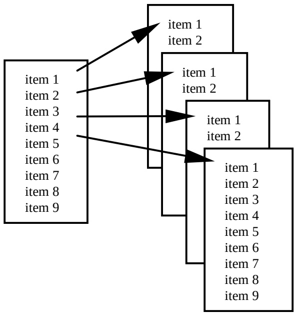

Figure 2: Broad Tree Structure

We considered three general strategies for designing voice prompts in support of search. First, the user can be presented with the list of possible utterances immediately; this is the *up-front* strategy. The other two strategies begin the interaction by inviting the user to say any utterance; the strategies differ when an utterance is not recognized. In the *delayed* strategy, the system then presents the available options. In the *on-request* strategy, the user is not presented with the available options unless they explicitly ask for help.

The argument behind this selection is that most designs will derive from these basic approaches: one can relate the *up-front* strategy to menu driven interfaces, the *delayed* being closer to an interface that uses “assistants”, while the *on-request* approach resembles more to a command-line interface. *Up-front* prompting gives the initiative to the system, while *on-request* prompting leaves initiative to the user; *delayed* prompting being a mixed case. While there are advocates for each of these strategies, it is possible that the efficacy of the different type of prompts is conditioned by the structure of the data. That is, prompts that might be efficacious for a broad tree structure might not be effective for a deep tree structure.

One of the strongest arguments for the existence of interactions between data structure and prompting strategy is the relationship between the perplexity of a spoken dialog and the cognitive load put on the user. Here, the term "perplexity" refers loosely to the information theory terminology: Perplexity is a measure of how much a random variable is predicted on average by a model. In the case of a prompted dialog, assuming that users have a mental model of the search objective, each newly-prompted item will affect their perplexity. Adding a new option will uniformly add ambiguity to the dialog, but depending on how accurately each item can be selected or discarded, the overall perplexity might increase or decrease as a result of introducing additional information.

All other things being equal, dialog designers would want to structure their dialogs to minimize perplexity: The trade-off lies between the number of prompted items — which increases the space of possibilities — and the ambiguity between items — which decreases as options get more specific.

Such optimization problems entirely depend on the prediction model, and heuristics have been proposed for such design task (Hansen et al, 1996). In spoken interactions, very little is known about the cognitive models involved, except for their strong connections to short-term memory (Luce et al, 1983). Indeed, in the case of spoken interactions, one can suspect that this cognitive limitation strongly affects how people process prompts. Thus, in the present experiment, we define a “broad” tree as one that has enough items at each level that all of the items cannot remain in short-term memory. Conversely, we define a “deep” tree as one that has too few items to strain short-term memory.

The early works of Miller (Miller, 1956) found the capacity of short-term memory to be of seven plus or minus two items, while more recent investigations of the concept of “working memory” reduced this number to four (Broadbent, 1975; MacGregor, 1987), or even less (LeCompte, 1999).

2 Experiment

This experiment was based on an interaction with a simulated search engine over the telephone. Its structure was similar to Web directories such as the Open Directory Project[2]. The context was a telephone-based directory assistance application.

A key component of the design of the experiment was to attempt to be independent of the underlying data. The following guidelines were used:

1- The total amount of information contained in each condition was exactly identical: A node with less out-going edges in one condition would consequently lead to a longer sequence of interactions.

2- The total amount of information known to the user at the end of each search task was exactly identical: Regardless of the actual number of interactions needed to complete the search, the overall number of clues presented was kept identical.

3- The prompts used exactly the same speech segments in all conditions: These were recorded using a female voice, spliced into units and concatenated on the fly depending on the condition. As a consequence, the end result controlled for audio quality and intelligibility.

4- The underlying data was neutral and balanced across two different domains: The directory permitted people to search for a professional by specialty and to search for a restaurant by type and geographic location. The content was made-up based on real and fictitious people and locations.

Here are examples of 2 typical search paths:

Main Menu → People → Contractors → Plumbing → Home → Carl Lippert → *connecting*

Main Menu → Restaurants → Seafood → Blue Fish, Two Stars restaurant in the Downtown Area → *connecting*

Each top-level category could lead to 195 distinct results. Subjects were asked to perform three specific search tasks in a random order:

- A high quality deli in the West Side area
- A phone consultation with a tarot reader
- A pediatric cardiologist

2.1 Procedure

The participants were 60 adults, randomly assigned to condition. The procedure was a 3 (prompting strategies) by 2 (data structure), between-participants experiment. All participants were sent e-mails pointing them to a web page. The web page contained a general description of the search engine. General tips were provided, including the possibility of interrupting the prompts at any time, saying "help" to get assistance, saying "go back" to go up one level, and "top level" to access the head of the tree.

The participants would key in their identification number, and be given directions for the three search tasks in a random order, as well as the phone number to be called[3].

The search engine itself was built using Nuance SpeechObjects[4]. These Java components have the different prompting strategies built in. The structure of the search tree was altered by pointing the software to different directory structures.

Upon calling the system, users were asked to say and confirm their identification number, and were assigned automatically to their condition. After a welcome prompt, participants were transferred to the main menu of the search engine.

2.2 Manipulations

The 3 types of prompting strategies were:

1. *Prompted lists up-front*: At every level of the search, users were prompted with the category they just selected, followed by all the options they could say. Here is the example from the restaurant listings:
   Computer*:* Mexican, Italian, French…
   User*:* Italian
   Computer*:* Italian…<pause>…Downtown, West Side…
2. *Delayed help*: Users are prompted with the name of the category they reached, and are prompted with an open-ended question describing their set of options. On a rejection or a timeout, the system would apologize and tell the user the available choices.
   Example:
   Computer*:* restaurants…<pause>…Which type of restaurant would you like?
   User*:* a cheap one
   Computer*:* Sorry, I didn't understand. You can say: Mexican, Italian, French…
3. *Help on request*: The initial prompts are identical to the ones used in the delayed help case. On a rejection or a timeout, the system would apologize and tell users that they can request help at any time.

[2] The Open Directory Project: http://www.dmoz.org

[3] See: http://vanhoucke.com/comm/. ID: 4000000

[4] Nuance Communications: http://www.nuance.com

Computer: restaurants…<pause>…Which type of restaurant would you like?
User: a cheap one
Computer: Sorry, I didn't understand. Please say help if you require assistance.
User: help
Computer: You can say: Mexican, Italian, French…

Each level of the broad tree had ten or more items. To ensure that the broad tree would strain short-term memory, the appropriate option was placed at one of the last points in each of the lists. Conversely, each level of the deep tree had four or fewer options. To ensure equivalent content, individuals were required to traverse more levels of the deep tree than the broad tree.

2.3 Measures

A Web-based questionnaire was used for all attitudinal dependent measures[5]. Each item used a ten-point Likert scale anchored by "Describes Very Poorly (=1)" and "Describes Very Well (=10)."

One set of questions asked: "How well do each of the following adjectives describe the Voice Search search engine?" Based on theory and factor analysis, we created two indices.

Ease of Use was an index composed of four adjectives: clear, convenient, easy to use and simple. The index was very reliable (Cronbach's alpha = .90).

Usefulness was an index composed of four adjectives: competent, efficient, reliable and useful. The index was very reliable (alpha = .89).

Another set of questions asked: "How well do each of these adjectives describe how you felt while conducting searches with the Voice Search search engine?" Based on theory and factor analysis, we created three indices.

Sense of Control was an index composed of 4 adjectives: dominant, powerful, relaxed and secure. The index was very reliable (alpha = .81).

Sense of Success was an index composed of 4 adjectives: competent, conclusive, effective and successful. The index was very reliable (alpha = .90).

Liking was an index composed of eight adjectives: engaged, comfortable, interested, pleasant, annoyed (reverse-coded), bored (reverse-coded), frustrated (reverse-coded), and vexed (reverse-coded). The index was very reliable (alpha = .86).

To evaluate performance, we examined the call logs. From these, the time spent on the system could be computed, as well as the number of interactions, success rate and accuracy. However, because some subjects hung up earlier than we expected (as soon as they heard the result they wanted, instead of selecting it by voice), several of these performance measures had limited reliability.

Because the utterances to be recognized were very similar and the complexity of the grammars was very small, recognition accuracy was not a significant factor in this study. It was important, however, that the recognizer fail for some interactions, because some prompting strategies differed only in a failure situation. Inspection of the logs showed that there were enough failures in all conditions to ensure that the manipulation was apparent.

3 Results

Manual inspection of the output logs show that about one caller (out of 10) in each condition did not succeed in a maximum of one of the search tasks. Additionally, a few subjects did more than the three calls they were asked to perform. However we cannot determine whether these were "trying out" the system, or exiting because they could not figure out how to proceed. These circumstances being hard to control for, and the vast majority of the participants not experiencing any difficulty, we did not perform any statistical analysis of success rate.

Since the utterances to be recognized were very similar and the complexity of the grammars was very small, recognition accuracy was not a significant factor in this study. It is important however that the recognizer fails for some interactions because some prompting strategies only differ in a failure situation. Inspection of the logs shows that these differentiating interactions were experienced by all participants for these conditions.

There was a significant interaction for ease of use, $F(2,54) = 8.05$, $p < .001$ (see Figure 3). Post-hoc analysis suggests that the source of the interaction is that for broad tress, delayed was easier than up-front, but for deep trees, up-front was easier than delayed. On-request prompts were easier than delayed prompts, $F(2,54) = 3.5$, $p < .04$.

[5] http://www.stanford.edu/~nouk/comm/

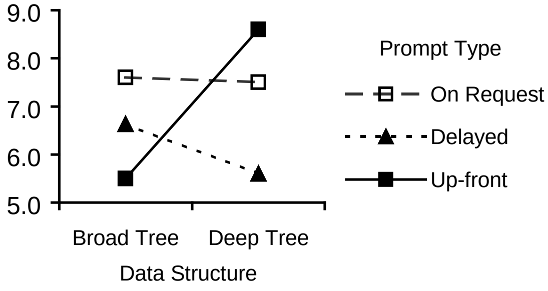

Figure 3: Ease of Use

There was a significant interaction with respect to perceived usefulness of the system, F(2,54) = 25.3, p < .001 (see Figure 4). In the broad tree case, prompting up-front was rated much less useful than any other strategy. In the deep tree case, the delayed help was rated less useful than the other prompts. In addition, the deep tree structure was seen as more useful, F(1,54) = 11.1, p < .001. Also, help on request was rated more highly than the other two conditions, F(2,54) = 16.5, p < .001.

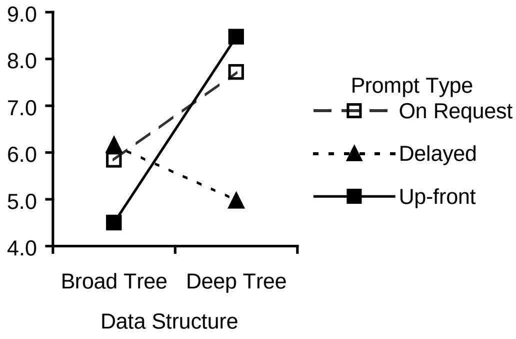

Figure 4: Usefulness

There was a significant interaction with respect to sense of control, F(2,54) = 6.75, p < .002 (see Figure 5). For the broad tree, the prompting strategies seemed to have little influence on the participants' sense of control, while in the deep tree case, the prompt up-front gave a stronger perception of being in control than did delayed help. There was a main effect for prompt, F(2,54) = 3.3, p < .05, but this was an artifact of the interaction. There was no effect for tree structure.

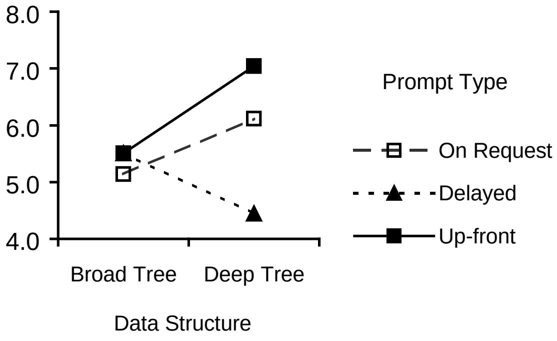

Figure 5: Sense of Control

There was a significant interaction with respect to user's perception of success, F(2,54) = 8.2, p < .001. Up-front prompts seemed to facilitate a sense of success much more for the deep tree structure as compared to the broad tree structure, while there was no difference for the other two types of prompts. On-request prompts gave users a stronger sense of success than did delayed prompts, F(2,54) = 6.26, p < .01. Deep-tree structure was significantly larger than broad-tree structure, F(2,54) = 19.1, p < .001, but this was an artifact of the significant interaction.

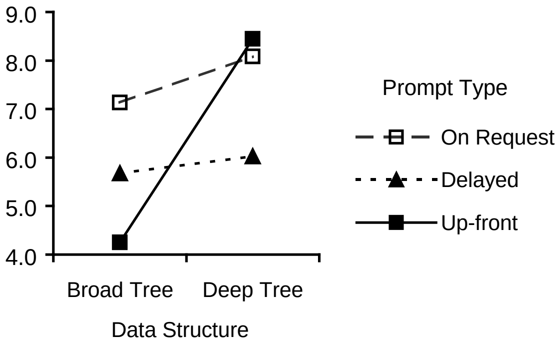

Figure 6: Sense of Success

There was a significant interaction with respect to liking, F(2,54) =5.51, p < .01 (see Figure 7). On-request prompts were liked better with a broad tree, while up-front prompts were preferred with deep trees. There was also a main effect for prompts, F(2,54) = 12.01, p < .001, with delayed prompts liked less than the other two types.

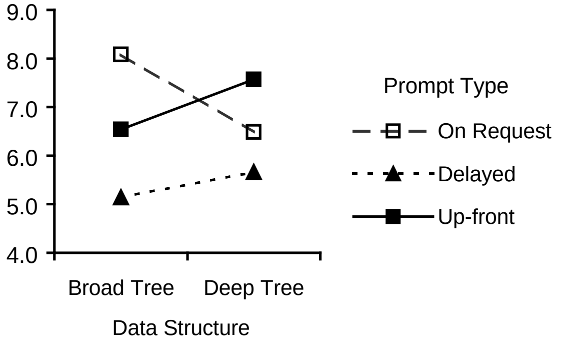

Figure 7: Liking

There was not a significant interaction with respect to time on task, F(2,54) = 2.42, p > .1 (see Figure 8). Users perform the tasks more rapidly with a deep tree as compared to a broad tree, F(2,54) = 7.98, p < .01. Up-front prompts allowed users to complete the search tasks more rapidly than on-request prompts, F(2,54) = 3.47, p < .04.

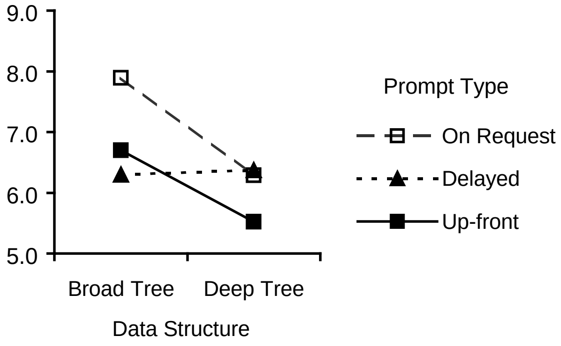

Figure 8: Time

4 Discussion

In VUI design, tree structure and prompting strategies are often addressed independently. The present results clearly demonstrate that decisions about prompts and data structure cannot be made independent of one another. Designers cannot simply combine and then implement their separate understandings of prompting and information structure and derive the results they anticipate.

As an example, a designer might strive to affect user control via his or her choice of prompting implementation. Working under the assumption that control is associated with active interaction rather than passive, the designer might choose an up-front or on-request prompting methodology. (With the delayed prompting structure, help is imposed; the user is removed from the role of decision-maker.) This analysis, while applicable to the deep tree structure, would, according to our results, have little effect on the user's sense of control in the case of a broad tree structure. Possibly, with the broad structure, the frequency of interaction with the system gives the user enough other opportunities for positive action that the effects of prompting are mitigated. In contrast, for the broad tree, where the number of commands that the user actually gives the system is significantly lower, prompting effects exert an increased influence upon the user.

Across a wide variety of measures, up-front prompts were most desirable for deep trees, while the other two strategies were more desirable for broad trees. However, of greatest significance is the fact that, as designers, we must be very careful in making generalizations about the effects of prompting and data structure manipulations. It is interesting to note that even in the case of a search task, where the interaction itself should not matter as much as the information to be retrieved, users did not necessarily prefer the interface that would take them the fastest to the desired result.

One could argue that this study involved participants that were exclusively first-time users. It is reasonable to suspect that the results would be significantly altered once the users start using the system on a regular basis. This would only strengthen the point that no single design strategy is universally appropriate. Additionally, many voice-driven interfaces are primarily designed for first-time users. Services that are meant to be used by occasional users – less than once a month – should probably assume that all users are rediscovering how the interface works each time.

Our attempts to control for the underlying data might also have hidden some important effects such as the influence of the user's familiarity with the domain, or the nature of the task performed. These effects might supercede the ones we describe here in specific contexts.

Future work might involve a study in which users were exposed to the system more frequently, thus addressing any issues that may have resulted from system novelty. However, regardless of this potential limitation, the interaction of prompting methodology and data structure speaks to the subtlety and complexity of VUI design and provides definite cause for re-examination of the paradigms that govern present design.

References

Broadbent, D.E. (1975), The magic number seven after fifteen years. In A. Kennedy and A. Wilkes (editors) Studies in Long-Term Memory, New York: Wiley, 3-18

Gardner-Bonneau, D. (1999), editor, Human Factors and Voice Interactive Systems

Hansen, B., Novick, D.G., Sutton, S. (1996), Systematic Design of Spoken Prompts – CHI '96: Proceedings of the Conference on Human Factors in Computing Systems, ACM press, 157-164

Kamm, C. (1994), User Interfaces for Voice Applications, Voice Communications between Humans and Machines, National Academy Press, 422-442

Larson, K., Czerwinski, M. (1998), Web Page Design: Implications of Memory, Structure and Scent for Information Retrieval - CHI '98: Proceedings of the Conference on Human Factors in Computing Systems, ACM press, 25-32

LeCompte, D. (1999), Seven, plus or minus two, is too much to bear: Three (or fewer) is the real magic number, Proceedings of the Human Factors and Ergonomics Society, 289-292

Luce, P., T. Fuestel, D. Pisoni (1983), Capacity Demands in Short Term Memory for Synthetic and Natural Speech, Human Factors, 25:17-32

MacGregor, J.N. (1987), Short-term memory capacity: Limitation or optimization ?, Psychological Review, 94(1), 107-108

Miller G.A. (1956), The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information, The Psychological Review, vol. 63, 81-97

Walker,M., Fromer, J., Di Fabbrizio, G., Mestel, C., Hindle, D. (1998), What can I say?: Evaluating a Spoken Language Interface to Email – CHI '98: Proceedings of the Conference on Human Factors in Computing Systems, ACM press