

Effects of Prompt Style when Navigating through Structured Data

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 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.

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.

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.

In this paper, we address strategies for permitting users to

navigate through large databases via a speech interface. This domain is important because search is the most fundamental and common activity on the Web.

One factor that influences search strategies is the structure of the underlying data. As a cursory examination of the major web search engines reveals, the most common method of data organization is a tree structure. 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

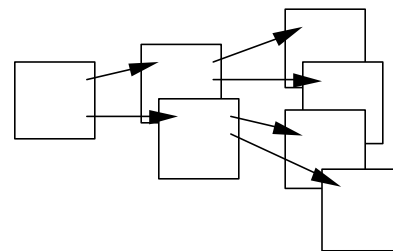
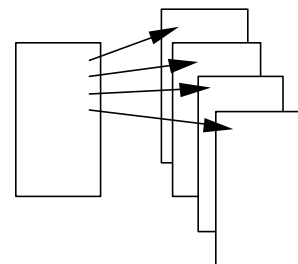


Figure 2. Broad Tree Structure



In GUIs, the dominant approach to search is a wholly open-ended query; that is, the user is presented with a text box and invited to type in anything. This strategy is not viable in VUIs, as the typical vocabulary of speech recognition engines cannot be made broad enough to support arbitrary utterances. Thus, VUI designers must develop new strategies for prompting the user in navigation.

There are 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.

While there are advocates for each of these strategies, it is possible that the efficacy of the different prompt strategies 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 [2]: 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. In spoken interactions, very little is known about the cognitive models involved, except for their strong connections to short-term memory. 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.

EXPERIMENT

This experiment was based on an interaction with a simulated search engine over the telephone. The engine structure was similar to directory-based web engines such as the Open Directory Project [3]. The context was a telephone-based directory assistance application. The directory permitted people to search for a professional by specialty and to search for a restaurant by type and geographic location.

Here are examples of the three typical search paths:

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

Main Menu → Restaurants → Seafood → Lake Front → Two Stars → Blue Fish → *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

Method

Procedure

The participants were 60 undergraduate and graduate students, 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 called¹.

The search engine itself was built using Nuance SpeechObjects. 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.

Manipulations

Prompting Strategies

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.

¹ See: <http://vanhoucke.com/comm/>. Sample ID: 4000000

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.

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...

Broad vs. Deep Tree Structure

To select an appropriate number of items for the broad and deep trees, we were guided by the notion that short-term memory can contain seven plus or minus two items.[4] Hence, 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.

Measures

A Web-based questionnaire was used for all attitudinal dependent measures². 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 ($\alpha = .81$).

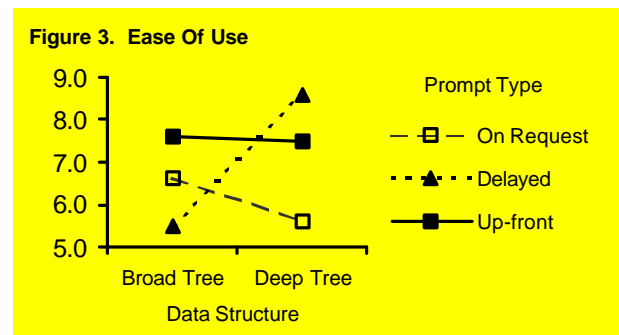
Sense of Success was an index composed of 4 adjectives: competent, conclusive, effective and successful. The index was very reliable ($\alpha = .90$).

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 to determine how much time the participants spent on the search tasks.

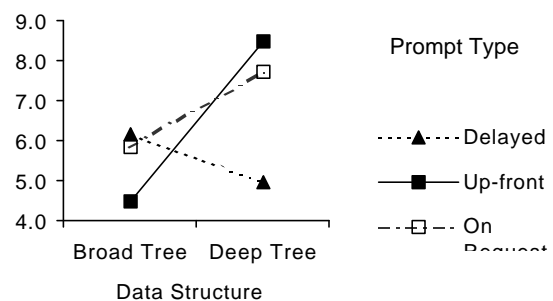
RESULTS

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 trees, on-request was easier than up-front, but for deep trees, up-front was easier than on-request. Up-front prompts were easier than on-request prompts, $F(2,54) = 3.5, p < .04$.



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, $F(2,54) = 16.5, p < .001$.

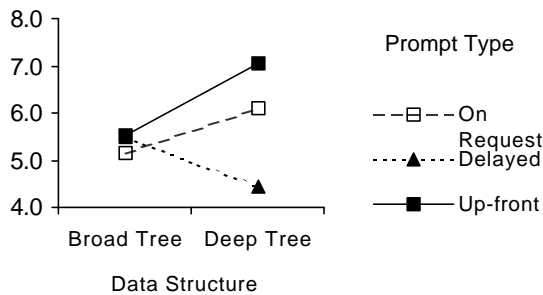
Figure 4. Usefulness



² <http://www.stanford.edu/~nouk/comm/questionnaire.fft>

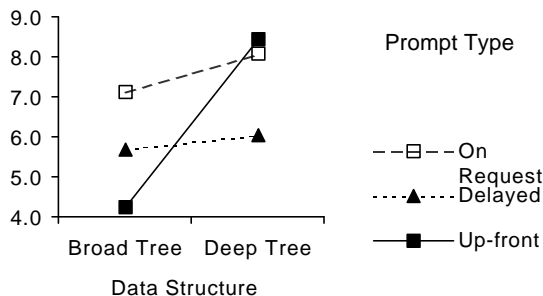
There was a significant interaction with respect to sense of control, $F(2,54) = 6.75$, $p < .002$ (see Figure 5). For the broad tree, the prompting strategies seemed to have little influence on the participants' sense of control, while in the deep tree case, the prompt up-front gave a stronger perception 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). Up-front prompts were liked better with a broad tree, while the other two types of prompts were preferred with deep trees. There was also a main effect for prompts, $F(2,54) = 12.01$, $p < .001$, with on-request prompts liked less than the other two types.

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, $F(2,54) = 3.47$, $p < .04$. The time on task was not significantly affected by error rates, as virtually all subjects completed all tasks successfully.

Figure 7. Liking

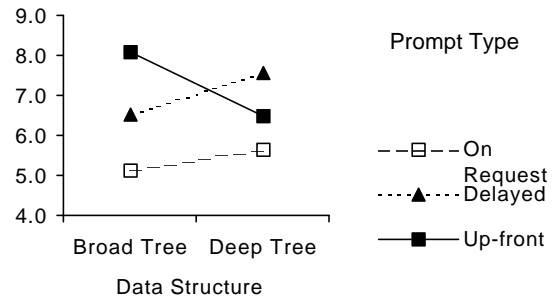
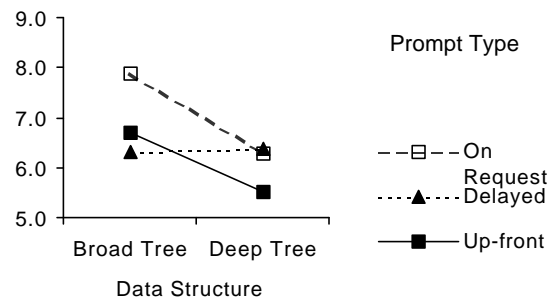


Figure 8. Time



DISCUSSION

In VUI design, structural 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 upon 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.

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 definitive cause for reexamination of the paradigms that govern present design.

REFERENCES

1. VoiceXML Forum. Online at: <http://www.voicexml.org/>
2. Manning, C.D., Schütze, H., *Foundations of Statistical Natural Language Processing*, Information Theory (2.2), Perplexity (2.2.8), MIT Press, 1999
3. The Open Directory Project. Online at: <http://www.dmoz.org>
4. Miller G.A., *The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information*, *The Psychological Review*, 1956, vol. 63, pp. 81-97.