System- versus User-Initiative Dialog Strategy for Driver Information Systems

Chantal Ackermann, Marion Libossek

Institut für Phonetik, LMU München, Germany
chantal.ackermann@phonetik.uni-muenchen.de
marion.libossek@phonetik.uni-muenchen.de

10.21437/Interspeech.2006-146

Abstract

In this paper we examine the advantages of a system-initiative approach for the voice user interface of a driver information system (DIS). The problems of user-initiative systems are the steep learning curve and the high demand on memory to recall the correct voice commands. This is especially true in a car environment where the main, and most important task is to drive the car. In a Wizard of Oz experiment, we compared this approach to one that uses a more system-initiative form of interaction. Furthermore, a context sensitive help prompt was included in the new system, instead of just a context sensitive list of commands. The results show that, for novice users, the error rate, the number and time of task completion, the mental workload, and the subjective *joy of use* (as measured by a semantic differential) are all better for the proposed system. Nevertheless, the possibility to use shortcuts remains. Thus, an expert user could still skip the supermenus and jump into the given submenu by saying the right voice command.
Index Terms: voice user interface, system initiative, user initiative, usability, human-computer-interaction

1. Introduction

In recent years, spoken dialog systems have become a fast-growing field as a new and, hopefully, better way to communicate with technical systems. A distinct improvement can be seen by now, for example, with telephone applications, where, instead of having to press buttons, full sentences can be used to accomplish certain tasks. Presently, research is still going on about how best to create user-friendly, helpful systems that get the job done. Several dialog strategies are being tested, though it seems very likely that each domain will end up with a different set best suited for its purposes.

2. Issues of Speech Dialog Systems for In-Vehicle Use

A very special and important field are dialog systems in vehicles to aid the driver configure and use the multitude of functions available in the DIS—e. g. entering the destination into the navigation system, changing the radio station, etc. Here, several restrictions apply, making the task of creating a helpful system a very challenging one.

2.1. Embedded Systems

Many of the dialog systems for telephone applications run on powerful computers with special hardware, dedicated to exactly this task. In the car environment, however, we have to deal with embedded systems that have to share the computing power with many other systems leading to restrictions and sometimes problems in quite a number of areas. One example is that the speech recognizer doesn't run continuously accepting user input only at specific points during the dialog. This prevents barge-in, so that the users cannot interrupt the system, e. g. when a list of options is presented. In this case, the user has to wait and remember the option, which creates an additional memory load, particularly when the option he wants is among the first being offered.

2.2. Driver Distraction

In contrast to many telephone applications where the caller can use his full concentration on the task of communicating with the system, the primary task in the car is driving. So, one of the main requirements for dialog systems in the vehicle is to distract as little as possible. A number of studies (e. g. [2]) on driver distraction and workload measures have shown that the use of the acoustic channel, whether for output (as in the navigation system) or input (dialog system) significantly reduces the mental workload, reaction time and visual distraction compared to manual / visual interaction.

However, other studies have shown that, as in the case of a telephone call, the problem is not holding the mobile phone, but having a conversation [3]. This shows that, although speech has great potential for making the use of driver information systems less distracting, it can also achieve exactly the opposite result.

2.3. Combination of graphical and voice user interface

Since not all cars with an information system are equipped with voice capability, the use of the system is primarily through its graphical user interface (GUI).

If all voice commands are valid inputs in every state of the system, the scope of the voice user interface (VUI) is wider than the scope of the GUI. Which means that the number of steps to select a certain function by hand is at best equal or greater than by voice. But whenever a menu entry of the GUI is not part of the ASR (Automatic Speech Recognizer) vocabulary, the scope suddenly changes: while the entry is always selectable by hand after a certain number of steps, it can never be reached by voice. In those cases, the entries should be marked clearly as non-voice-commands to prevent futile attempts.

In the worst case, all subentries of a menu are non-voice-commands. While a system-initiative strategy would offer a helping prompt or at least some information, a strictly user-initiative strategy might abandon the user in a dead-end (of the VUI).

2.4. User Model

Another important point is the user group. These systems are bought as part of a new car. As cars, even those with a lot of

new high-class gadgets, are something well-known to everyone but driver learners, most people usually don't take the time to study the manual carefully, before going for a ride.

So, most of the users come up against the DIS without previous training. And even though some, after a few false starts, invest the time and effort to learn about it, most people will use those parts of the system they understand and can control easily. And simply ignore the rest. The VUI in particular will be neglected if it cannot be used easily from the beginning, as it is still the most exotic modality, and there are always the more conventional means of manual control available. Thus, if the dialog system is not to be disregarded in favor of the latter, it is absolutely vital that it can be used intuitively with a gentle learning curve.

And, last but not least, even experienced users might have problems with the system when they are using it in situations requiring high levels of concentration for the traffic. Thus, in our opinion, even though such a system might talk a little bit more than strictly user-initiative versions, it compensates this by being more robust, and supporting.

2.5. User-initiative vs. mixed-initiative dialogs

For quite some time, it was held that a more system-initiative dialog in the car is an unusual feature. The idea was that, e. g. the "interaction with the car stereo would be largely user-initiated", whereas "the car telephone will demand a roughly equal mixture of user-initiated and system-initiated interactions" (see [4]). This shows the idea behind the two concepts. User-initiated is whenever all the action comes from the user. He starts a dialog at his convenience and is, for the most part, responsible for the course of the dialog, the system doesn't prompt him for the next step. The contribution of the system is, for example, limited to giving feedback as to what command was extracted by the recognizer. This is perfect for users who know the system by heart, and know exactly what to say, so they can skip any lengthy explanations, and detours through supermenus. Users ignorant of the functionality and the limitations of the system, on the other hand, have no other means of finding out how to use it than to make a lot of mistakes and try to learn from whatever error messages they get.

The advance in technology makes it possible today to evaluate quite a number of variables. By taking a step towards adapting the system to a specific user, or by considering system states, the system can be reasonably sure what to expect, and make suggestions. Thus, when the user chooses the navigation menu and no destination has yet been entered, and the car is still stationary or has just been started, a reasonable assumption by the system may be that a new destination or one from the address book is about to be given—instead of expecting an adjustment to the current route criteria. This reasoning makes it possible for the system to prompt the user for the destination directly, instead of waiting for him to try and do so.

This approach differs somewhat from the idea of system-initiated prompts mentioned above, where the system would only become active when a message from either the telephone, warning systems, or monitoring systems would make it necessary to give a message to the driver.

With regard to those considerations, the goal of this evaluation was to show that using a system-initiative dialog strategy is for this special case

- more efficient in terms of dialog steps and, in consequence, time to completion,
- more user-friendly,
- able to reduce the mental workload and thus the distraction from the driving task,
- more accepted by the user and perceived as more agreeable.

3. Evaluated Systems

3.1. User-Initiative System (System A)

The reference system uses a command-and-control structure and a user-initiative dialog strategy. The user has to learn a set of commands that make it possible to control the system very efficiently. The system gives feedback as to the system state, but leaves the choice of the next step entirely to the user.

In the GUI the possible functions are structured as hierarchical trees displaying most siblings (as far as GUI restrictions would allow) and the children of the selection to the user's sight. However, there are menu entries in the GUI that are not valid speech commands and result in an error if spoken.

3.2. System-Initiative System (System B)

The proposed system is also built as command-and-control, but the strategy is system-initated, as the speech output consists (mainly) of questions illiciting replies. This is done by presenting the most probable options to the user. With this strategy, it is possible to narrow down the speech recognizer's grammar whenever the system question leaves only few options as answer.

One example is the prompt when the user enters the navigation menu. Here he would be presented with three options "enter destination", "city" and "street" as possible input commands: "To enter a destination please say *enter destination*. If your destination is in Germany say *city*, if your destination is in Munich, say *street*."[1] This is in fact the part where the evaluated systems differ most. Of course, this prompt is build with the context information available through the GPS signal of the navigation system.

To keep both systems comparable, only the system prompts were changed while the GUI was identical. Basically, the same set of speech commands was possible. Only speech was available as input modality with no manual control.

3.3. Error Management

One of the main differences between the two evaluated systems was the error management, even though the general structure was similar: on the first error, an apology requesting a repetition would be presented, on the second one, some kind of context sensitive prompt. In case of System A, this consists of a list of available sub-options for the currently selected menu or, in case there are none, the siblings of the selected menu. System B presents up to three of the most probable input possibilities, depending on the context, in a more natural way.

On the third error in a row, both systems abort the current dialog and return to the main menu. We decided upon this reaction due to the fact that normally in this situation, the user would have to continue with another input modality (manual input). Table 1 opposes the different error management strategies for the two systems.

[1] "Um ein Ziel einzugeben, sagen Sie *Zieleingabe*. Wenn Ihr Ziel in Deutschland liegt, sagen Sie *Ort*, wenn Ihr Ziel in München liegt, sagen Sie *Straße*."

Admittedly, it is subject to further investigation whether the context help of the system-initiative version is still helpful when the user is trying to do anything else than entering an address.

error	user-initiated	system-initiated
1st	*I didn't understand, please say that again.* 132 times	*I'm sorry?* 39 times
2nd	list of options 90 times	context help 27 times
3rd	*Process terminated. Main menu.* 9 times	*Process terminated. Main menu.* 0 times

Table 1: The error management for both systems. The number shows how many times these specific system states were reached.

4. Method

4.1. Setup

For a comparison of these different dialog strategies, we decided to use the task of entering an address into the navigation system, set up in a Wizard of Oz environment. Two wizards controlled the dialog flow of the GUI and the speech output (spoken by one of the wizards) according to the speech input by the user. The "recognition" of the speech input was also done by the wizards. As our main interest lay in the evaluation of the dialog strategy, the wizards were instructed to have a high "recognition rate" from an acoustic point of view. However, off-talk (talking to the experimenter) was treated as input to the system and produced an error. This contributed to a feeling of reality. That way, we could evaluate the differences between the two dialog strategies instead of the quality of the speech recognizer.

Eighteen subjects (7 female, 11 male, aged between 40 and 61 years) participated in the evaluation.[2] Each person was asked to enter the same six addresses in exactly the same order. The first and fourth address consisted of a street name, a house number and a city (address type 1), the second and the fifth address of only the street name and the city (address type 2), and the third and sixth address were only city names (address type 3). Here, the first three addresses had to be entered into one, the second three addresses into the other system. The order of System A and B was alternated between subjects. Directly after each set of three addresses, the subjects were given a *semantic differential*: they were asked to rate the system they had just used with 33 opposing pairs of adjectives along a seven point scale.

4.2. Simulation and Measurement of Distraction and Workload

To simulate the driving situation we used the Lane Change Task (LCT) (see [5]). This is a driving task on a road with three lanes and no other traffic than one's own car. The task consists of driving on a straight road that is lined in irregular intervals by signs indicating which lane to choose. As soon as the sign is recognizable, the driver has to change to the indicated lane as quickly and exactly as possible. The analysis tool, part of the LCT, calculates the mean deviation from the ideal lane change trail.

[2]The age pattern and the distribution between the sexes represent the average audience of the cars equipped with a spoken dialog system.

This task was used to put the subjects in a state of concentration similar to that while really driving. The results show that the secondary task (entering a destination into the navigation system) had a measurable impact on the driving accuracy of the LCT. Equally, we could observe that for many of our subjects the LCT was demanding enough to mimic the concentration required for real driving as it had an effect on the secondary task, as well—e.g. staring straight on the main screen, instead of checking to see what options are available on the system screen (see 5).

5. Results

The results place the system-initiative dialog strategy significantly better than the user-initiative one considering all parameters analyzed.

A task was defined as entering a given address into the navigation system and starting the navigation. As there were six addresses, each subject had six tasks. The success rate is computed by counting each completed task. While more than 90 percent of the tasks were successful for System B, System A had only about 40 percent success rate.

In many cases the failure can be traced back to the following facts:

- The subjects chose the wrong option and consequently ended up in a dead end (see 6).
- The subjects did not know what to say and produced errors that could not be solved to their satisfaction by the error management of the user-initiative system (see 3.3 and table 1).
- The subjects took too long and reached the end of the LCT track before entering the complete address and starting the navigation.

While the shortest input times where observed for System A, the mean input time for all subjects, depending on the type of address (see 4), is significantly shorter for System B (figure 1).

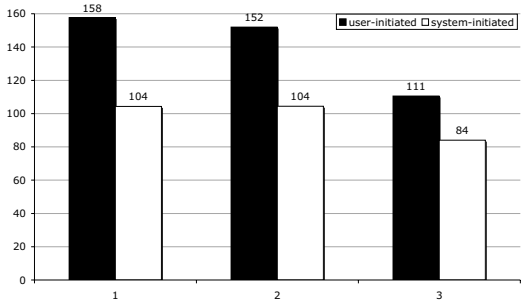

Figure 1: The mean time in seconds for all subjects, depending on the type of address (type 1, most complex, to type 3, least complex) and the system (black: System A, white: System B).

The LCT is handled very differently by the subjects. Some have great difficulties to adapt to the specific driving characteristics while others get used to it quite fast. To be able to compare the results of the subjects, we used only relative values consisting of the difference between the mean deviation from the baseline[3]

[3]Here, the baseline is the ideal lane change trail as calculated by the LCT analysis tool.

without second task to that with second task for each person separately. Here, too, the results were computed depending on the type of address: for all three types, the mean deviation is better for System B (see figure2). Although in the case of the first address type, the difference is not significant.

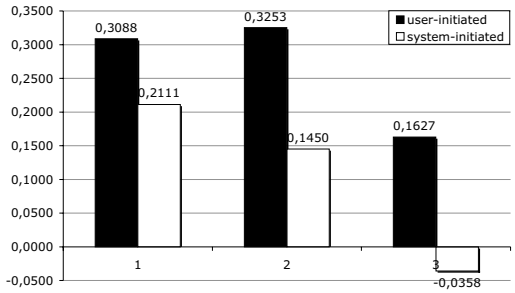

Figure 2: The mean deviation of all subjects per task type (type 1, most complex, to type 3, least complex) in relation to their mean deviation from the baseline without doing any secondary task. Black represents System A, white System B.

The good results for the objective parameters is mirrored by the interpretation of the semantic differential: whenever one pair of opposing adjectives shows significantly different trends for the two systems System B comes off better—this is the case for 24 out of 33 pairs.

6. Discussion

The feature both systems have in common is the fact that any command can be entered at any time.[4] Thus a power user who knows the system can jump right into the appropriate submenu by saying *enter destination* e. g. in the main menu—without seeing this option on the display.

This might appear a mighty strategy for power users but the courses that the dialogs took in our evaluation show that uninstructed users have no chance of learning about this feature in an easy way—or never have a chance of learning it at all—with the user-initiative strategy. This leads to the question whether user-initiative is really appropriate in an environment where uninstructed users have to learn

- in an easy way that does not distract from the driving task,
- on the fly, without introduction, without reading a manual,
- while really using the system (no tutorial with sample exercises),
- with a GUI that does not match the capabilities of the VUI (i. e. presents options that cannot be spoken).

It is probable, that after getting to know the system, the way in which a user enters an address will be the one that he perceived as the fastest, the most successfull and least frustrating one—and that this judgement will be reached after a few attempts, only. Thus, it can happen that the fastest way to enter an address could be by speech, using the appropriate commands, but that there is no user who has had the patience to find this out.

[4]Except the commands *city* and *street* that have been added to System B, and apply only to one specific submenu.

As stated in section 5, there are three main causes for the failure with System A.

Firstly, the dialog courses show that the subjects did not really know which option to select as the next command. Some obviously just guessed until they hit the right one. This is proof that the prerequisite for a user-initiative dialog is a well-defined and intuitive terminology—taken from the user's world of knowledge, not derived from engineering requirements (e.g. maximum phonetic difference, etc.).

Secondly, and this is closely related to the first cause, the option list that is displayed by System A on the second error in a row, is not helpful enough to avoid a third error (see table 1). The option that should have been chosen by the subjects was not clear enough.

Thirdly, the search for the right commands took the subjects so long that they often reached the end of the LCT track before completing the task. This is due not only to the terminology but also to subtle differences between the scope of the GUI and that of the VUI (see 2.3).

7. Conclusion

In current dialog systems, the trend is towards user-initiative systems. The idea is that the user should have a lot of freedom in fulfilling the desired task, and not be restricted by the system, e. g. about the order in which data is entered into a form. In the car, however, the driver profits more from a system-directed form of interaction. In this special case, reducing the learning effort, and the mental workload, and in consequence the driver distraction makes up for the seeming loss of initiative.

In this paper we compared the two approaches in a Wizard of Oz experiment, for the task of entering a navigation destination. The results show that the system-initiative system enables the unbiased user to fulfill the task faster while driving safer, and experiencing more joy of use.

With commands that apply system wide, experienced users can avoid the first steps from the main menu to the appropriate submenu. For that, we believe that a system initiative strategy—eventually allowing barge-in—satisfies the needs of both novice and expert users.

8. References

[1] Cohen, M. H. and Giangola, J. P. and Balogh, J., "Voice User Interface Design", Addison-Wesley, Boston, 2004.

[2] Jonsson, I., Zajicek, M., Harris, H., Nass, C., "Thank you, I did not see that: in-car speech based information systems for older adults". CHI '05 extended abstracts on Human factors in computing systems,

[3] Cohen, M. H. and Giangola, J. P. and Balogh, J., "The impact of auditory tasks (as in hand-free cell phone use) on driving task performance", ICBC Transportation Safety Research, 2001.

[4] Leiser, R., Driver-vehicle interface: dialogue design for voice input, in "Driving Future Vehicles", Parkes, A., Franzen S. (eds.), Tayor&Francis, London, 1993.

[5] Mattes, S., "The Lane-Change-Task as a Tool for Driver Distraction Evaluation." In: Strasser, H., Kluth, K., Rausch, H., Bubb, H.: Qualität von Arbeit und Produkt in Unternehmen der Zukunft. Ergonomia Verlag, Stuttgart, 2003.