Adding extra input/output modalities to a spoken dialogue system

Janienke STURM, Fusi WANG, Bert CRANEN
A^{2}RT, Dept. Language and Speech, Nijmegen University
Erasmusplein 1
6525 HT Nijmegen, The Netherlands
{janienke.sturm | f.wang | b.cranen}@let.kun.nl

Abstract

This paper describes a prototype of a multimodal railway information system that was built by extending an existing speech-only system. The purpose of the extensions is to alleviate a number of shortcomings of speech-only interfaces.

1 Introduction

For a long time, speech has been the only modality for input and output in telephone-based information systems. Speech is often considered to be the most natural form of input for such systems, since people have always used speech as the primary means of communication. Moreover, to use a speech-only system a simple telephone suffices and no additional devices are required. Obviously, in situations where both hands and eyes are busy, speech is definitely preferable over other modalities like pen/mouse. However, speech-only interfaces have also shown a number of shortcomings that result in less effective and less efficient dialogues.

The aim of the research described in this paper is to assess the extent to which multimodal input/output can help to improve effectiveness, efficiency and user satisfaction of information systems in comparison with unimodal systems. This paper describes how, within the framework of the MATIS[1] (Multimodal Access to Transaction and Information Services) project we developed a prototype of a multimodal railway information system by extending a speech-only version in such a way that it supports screen output and point-and-click actions of the user as input. This system is a typical example of a simple application that can be implemented using a slot-filling paradigm and may stand model for various other form filling applications.

First, a number of problems are described that arise in speech-only interfaces. Then we briefly describe the architecture of the speech-only railway information system. Next, we describe in more detail how we added multimodality to this version of the system and explain why we think this may help to solve the shortcomings of speech-only systems. We conclude this paper by discussing several open issues that we intend to solve by means of user tests with the multimodal system.

2 Shortcomings of speech-only interfaces

One of the issues that all dialogue systems with spoken input have to cope with is the imperfection of the speech recogniser. Even in very limited domains and with a small vocabulary speech recognition is never 100% accurate, if only because people may use OoD (Out of Domain) or OoV (Out of Vocabulary) words. To ensure that the user does not end up with wrong information, all slot values entered by the user must be confirmed. This can be done either explicitly in a separate question or implicitly, i.e. incorporated in the next question. Explicit confirmation results in a lot of extra turns, which means that the dialogue becomes less efficient and is often perceived as tedious, especially if all user utterances are understood correctly. Implicit confirmation, by contrast, does not necessarily increase the number of turns. However, it appears that users have difficulty in grasping the concept of implicit confirmation [Sturm, 1999]. Things run smoothly as long as the information to be confirmed is correct. If the speech recognition result is incorrect and wrong input expressions are confirmed implicitly, users tend to get confused and fail to repair the mistake that was made by the speech recogniser.

In order to reduce the need for confirmation, confidence measures may be used. A confidence score is an estimate of how certain one can be that the recognition result is indeed correct. Using confidence scores in combination with one or more thresholds, would for instance allow to decide upon 1) ignoring the recognition result (if the confidence is minimal), 2) confirming the

[1] http://www.ipo.tue.nl/projects/matis/

recognition result or 3) accepting the recognition result without confirmation (if the confidence is maximal). Unfortunately, it is virtually impossible to define thresholds in such a way that no false accepts (a user utterance is actually misrecognised but has a confidence score that exceeds the threshold) and no false rejects (user input was recognised correctly but has a confidence score that falls below the threshold) are caused. False rejects are not very harmful, although they do cause superfluous confirmation questions, and thus reduce the efficiency of the dialogue. False accepts, however, may become disastrous for the dialogue, since they cause incorrect values to be accepted without any confirmation. As a consequence, this strategy does not seem very attractive for speech-only systems.

Another problem with speech-only information systems is the way in which the eventual information is presented to the user. Shadowing experiments with different railway information systems indicate that users have difficulties understanding and writing down a travel advice presented in spoken form, especially if one or more transfers are involved [Claassen, 2000].

Last, and perhaps foremost, it appears that users have difficulty in building a correct mental model of the functionality and the status of a speech-only system. This lack of understanding explains problems with exceptions handling, and the user's uncertainty as to what one can (or perhaps must) say at any given moment.

3 Multimodality in MATIS

The first goal of the MATIS project is to investigate to what extent graphical output along with speech prompts can solve the problems that are due to the lack of a consistent mental model. If, for example, recognition results are not only confirmed (implicitly) in speech prompts for additional input, but also displayed in the corresponding field on the screen, detecting recognition errors may become easier. The same should hold for navigation through the list of possible connections that is returned after the input is complete and a database query can be performed.

If no keyboard is available speech is ideal for making selections from long implicit lists, such as the departure city. However, other fields in a form may offer only a small number of options, which can easily be displayed on a screen. In the railway information system this holds for the switch that identifies the time as departure or arrival time (and to a large extent also for entering the date, which usually is today or tomorrow). Selections from short lists are most easily made by means of point-and-click operations. Therefore, we decided to add this input mode to speech input.

3.1 System Overview

Our multimodal railway information system is an extended version of the mixed-initiative speech-only railway information system (OVIS) developed in the NWO-TST programme[2]. This is a very different starting point from most other projects in multimodal human-machine interaction, that seem to add speech to what is basically a user-driven desktop application. The user interface consists of a telephone handset in combination with a screen and a mouse. The MATIS system inherited an architecture in which modules communicate with each other using TCP socket connections under the control of a central module (Phrisco) (cf. Figure 1). The grey shaded modules have been added or extended for MATIS.

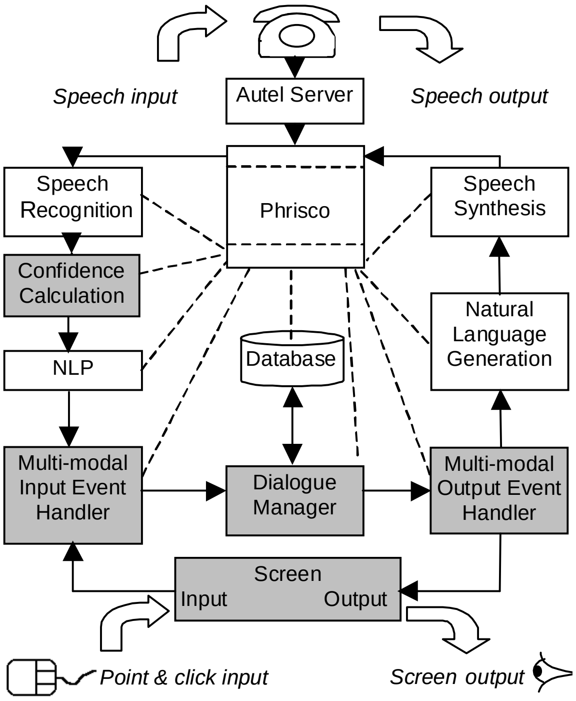

Figure 1 Overview of the MATIS system

In the next sections we will focus on the modules that have been added or changed and how these modules help to solve some of the problems described in Section 2.

[2] http://odur.let.rug.nl:4321/

3.2 Screen output

At the start of a dialogue an empty form is shown on the screen. In the course of the dialogue the fields are filled with the values provided by the user, who can use speech to fill all five slots in the form in a mixed-initiative dialogue, or use the mouse to select text fields and to make list selections. Once all slots have been filled, a travel advice is retrieved from the database and presented to the user in spoken and in textual form.

3.3 Mouse input

Experiments have been conducted using a Wizard of Oz simulation of the MATIS system, to establish to what extent subjects use the mouse in addition to speech and in what way mouse input is used in an interaction that is essentially the original mixed-initiative spoken dialogue [Terken, 2001]. It appeared that half of the subjects used mouse input as well as speech input and that mouse input was primarily used to make selections from short lists, and much less to select editable text fields. The latter was done mostly in the context of error correction.

3.4 Confidence calculation

Confidence measures (CM) for spoken input can be calculated in different ways. In the MATIS system the CM is based on an N-best list of sentence hypotheses that is generated by the speech recogniser [Rüber, 1997]. This N-best confidence score rests on the assumption that words that occur in more entries in the N-best list are more likely to be correct:

$$CM_{nbest} = \frac{\sum_{i=1:W\in h_i}^{N} P(h_i)}{\sum_{i=1}^{N} P(h_i)} \qquad (1)$$

where *P(h$_i$)* is the likelihood score of sentence hypothesis *i* in the N-best list. In this manner a CM is calculated for each word in the utterance.

The N-best CM may give rise to a specific problem: if the N-best list contains only one entry, (1) automatically yields a maximum confidence score for each word in the utterance. Off-line experiments have shown that 3% of all N-best lists consisting of only one sentence actually contained recognition errors. Consequently, even if we only trust words with a maximum CM score, the false accept rate will be at least 3%. Other off-line experiments have shown that some improvement may be expected from combining the N-best CM with another CM that does not have this artefact.

When a user fills a specific slot in the form using speech (s)he has to indicate which slot needs to be filled and provide a value for this slot. To obtain a CM for the slot value, the CMs of all words that were used to specify this value have to be combined. In the current implementation this was done by taking their mean.

3.5 Multimodal Input Event Handler

The information coming from the NLP module (in response to a spoken prompt) and from the mouse (that is active all the time) must be properly combined. This task is taken care of by the multimodal input event handler. To combine the information streams correctly, a time stamp must be attached to the inputs, indicating the temporal interval in which the action took place. This time interval is needed to decide which events should be combined [Oviatt, 1997].

Furthermore, speech and mouse input may contain complementary, redundant or unrelated information. Complementary information (e.g. clicking on the ‘destination’ field and saying ‘Rotterdam’) is unified before it is sent to the dialogue manager. Unrelated information (e.g. clicking to select departure time while saying one or more station names) is first merged and then sent to the dialogue manager. In the case of redundant information (e.g. clicking on ‘tomorrow’ while saying ‘tomorrow’), the information coming from the mouse is used to adapt the CM score attached to the speech input. Due to speech recognition errors, ‘redundant’ information may be conflicting (if the recogniser returns ‘tomorrow’ in the same time slot where ‘today’ is clicked). To solve this problem the information with the highest CM score will be trusted.

3.6 Dialogue management

The dialogue manager of the unimodal system was adapted in order to be able to use the CMs to decide on the confirmation strategy. In the present prototype we use only one threshold to decide upon the strategy. Values with a CM score below the threshold are shown on the screen and confirmed explicitly in the spoken dialogue. Values with a CM score exceeding the threshold are only shown on the screen. In case all or most values have a high CM score, this strategy speeds up the dialogue considerably. Preliminary experiments suggest that providing feedback visually as well as orally helps the user to develop an adequate model of the system. Also, since the user knows exactly what the information status of the system is at each point in the dialogue, correcting errors should be easier, which in turn will result in more effective dialogues. We are convinced that an increase in effectiveness and efficiency can be achieved, especially if the visual output is combined with auditory prompts that are more concise than in the speech-only system.

3.7 Multimodal Output Event Handler

In a multimodal system a decision has to be made as to whether the feedback to the user must be presented orally, visually, or in both ways. This is the task of the multimodal output event handler. For the time being we have decided to send all the output from the dialogue manager to the natural language generation module and the screen.

4 Discussion and conclusions

In this paper we have described the architecture of a multimodal train timetable information system that was built by extending a speech-only version. Most of the desired functionality of the modules that we added or changed was specified on the basis of off-line experiments and findings in the literature. The system is now ready to be tested by real users.

Adding visual feedback has been shown to help in several respects. In Terken (2001) it was shown that the visual feedback helps the user to build a mental model of the task at hand. Furthermore, we argued that visual feedback may be interpreted as a form of implicit verification, which helps the user to detect recognition errors. This allows to apply confidence thresholds to avoid confirmation turns, even if a number of false accepts occur. This is in contrast with speech-only systems, where false accepts will remain unnoticed.

User tests with our present prototype are needed to verify whether the additional modalities do indeed help to increase efficiency, effectiveness and user satisfaction. These tests will be conducted in the near future. In the current prototype a number of ad hoc choices were made. We expect that several of these choices will have to be revised based on the outcomes of the tests.

CM scores that are calculated for individual words must be transformed into scores for slot/value pairs. This can be done in several ways: by taking the mean score, the maximum score, weighting the scores for values and slots, etc. In the current prototype we take the mean of the scores of the words that yielded a certain slot/value pair, but more sophisticated methods may be needed.

In principle it is possible to go beyond the current design and give feedback on the status of the slots (confirmed or not, changeable or not) in addition to showing their values. This might prevent the user from getting lost in the dialogue. However, it is not yet clear whether additional visual attributes can be designed that are self-explanatory and will not confuse the user. It might be useful to enable the user to correct information by clicking the field that contains incorrect information and saying the correct information. Also, showing a list of alternative recognition hypotheses from which the user can select the correct one, might help. In the current system we have not implemented this option.

Currently, the complete output of the dialogue manager is sent both to the speech output module and the screen. Informal tests have shown that the speech output designed for a speech-only system is much too verbose. Especially the oral presentation of the travel advice can be a short summary, e.g. consisting of only the departure and arrival times, when the complete advice is also presented on the screen.

5 Acknowledgement

This work was supported by a grant from Senter/EZ in the framework of IOP-MMI.

6 References

B. Rüber (1997), *Obtaining confidence measures from sentence probabilities*, Proceedings Eurospeech’97, pp. 739-742.

W. Claassen (2000), *Using recall measurements and subjective ratings to assess the usability of railroad travel plans presented by telephone*, Technical Report #123, NWO Priority Programme on Language and Speech Technology.

S. Oviatt, A. DeAngeli, and K. Kuhn (1997), *Integration and synchronization of input modes during multimodal human-computer interaction*, in Proceedings of CHI '97, pp. 415-422.

J. Sturm, E. den Os and L. Boves (1999), *Issues in spoken dialogue systems: Experiences with the Dutch ARISE system*, Proceedings ESCA Workshop on Interactive Dialogue in Multimodal Systems, pp. 1-4.

J. Terken and S. te Riele (2001), *Supporting the construction of a user model in speech-only interfaces by adding multimodality*, Submitted to Eurospeech 2001.