Proceedings of the Institute of Acoustics

SPEECH AIDS FOR THE HANDICAPPED: DESIGN AND DEVELOPMENTS AT THE UNIVERSITY OF HERTFORDSHIRE

Christine Cheepen

University of Hertfordshire, Speech & Language Technology (*SLANT*), Hatfield, Herts, England, AL10 9AB

1. INTRODUCTION

For the past five years there has been constant, ongoing work on speech aids for the handicapped at the University of Hertfordshire. This work has been carried out within the Speech and Language Technology Team (SLANT). SLANT is a multidisciplinary group, involving academics from linguistics, computer science, education, business, law, mechanical and aeronautical engineering, electrical and electronic engineering. Using various combinations of their range of specialist fields, the team are involved in a wide range of different research projects focussing on speech and/or natural language as input/output media for automated systems. This paper focusses on the work of the SLANT team with regard to speech aids for the disabled, paying specific attention to the area of the user-machine interface in speech input systems.

2. EARLY WORK: THE ISDIP SYSTEM

The Intelligent Speech Driven Interface Project (ISDIP) ran from 1988 to 1991. The brief of the research team was to investigate problems and identify solutions in the design of speech input systems by developing:

- a hands free, voice driven word processor for disabled users who find hand movements difficult or impossible
- a generalised speech interface which would allow voice access to a range of software packages

(In 1990 the team won a BCS medal in the category Social Benefit, and a RITA award in the category Best Use of I.T. to help the Disabled for the prototypes produced.)

Both interfaces were designed to be speaker dependent and to use isolated word recognition. The following discussion will concentrate primarily on the development of the speech driven word processor, as the lessons learnt in that section of the project were directly applicable to the more general development in terms of interface design.

An early decision taken by the design team was to concentrate primarily on the user interface, and to develop a system which was, above all, simple to use. This inevitably meant that the power and flexibility of the system would be restricted to some extent, but it would have the great advantage of being easy to operate for even a novice user. Another important decision was to base our development on standard, low cost hardware which would be easily available to the disabled user. For this reason we decided to use a standard, pc platform, and our earliest prototypes were designed to run on floppy disks. Our underlying speech recognition hardware was the Votan vpc2000.

2.1 Problems for speech input systems
There are a number of problems inherent in the implementation of speech recognition systems. From early pilot projects in which we observed disabled users interacting with a speech driven system we established that these problems fall into three major areas - recognition problems, feedback problems and usability problems.

2.1.1 Recognition problems. It is inevitable that with any speech input system allowing a vocabulary of more than a handful of words the recognition will be inaccurate to some extent. This may be due to ambient noise or to variations in the user's voice. Whatever the cause, recognition will always be less than perfect. If the user has use of his hands then recognition errors can of course be easily corrected by resorting to the keyboard, but in the case of the disabled user this is not an option, and ways must be found to allow the user to correct errors using voice alone. An efficient solution to this kind of problem is essential if a speech input system is to be usable, as correction strategies are not only required when the recognition is imperfect, but also when the user's input is 'imperfect', i.e. when the user makes a mistake with an input item and then wishes to change his mind. A large part of our effort in this project was devoted to providing a workable solution to this problem (see section 2.2.1 below).

2.1.2 Feedback problems. These arise largely out of the restrictions imposed by the organisation of the underlying recognition device. The Votan vpc2000, along with most other commercially devices available in the late 1980s had a very restricted vocabulary structure, which allowed multiple vocabularies of only 64 words each. The recognition was done by matching the incoming signal with a stored template of the sound pattern for that item, which had been set up by an initial training session in which the user trained all words in the total vocabulary. If a user said a particular item while addressing the wrong vocabulary the system was unable to find a match, so recognition was totally dependent on accessing the right vocabulary. The responsibility for switching between different vocabularies in order to match the stored words with the appropriate templates lay with the user, who therefore had to know which items were stored in which vocabulary. In a task such as word processing, where a large number of vocabularies were necessary this became a problem, and a formidable source of potential errors. It quickly became clear that some form of elementary feedback was necessary to remind the user which vocabulary was currently being accessed. The immediate ISDIP solution to this is described in section 2.2.2 below, and section 2.4 deals with our work in the area of system intelligence, which went some way towards solving the problem of cognitive load on the user at a more general level.

2.1.3 Usability problems. A substantial amount of the usability problems were directly connected to the recognition and feedback problems outlined above. When errors arise it is not only important to ensure that the system can be used to correct them, it is also important to design the user interface in such a way that the user is aware of the nature of the problem, the facilities the system has for correction (and the limits on those facilities), and is equipped to exploit these facilities to correct the situation. System feedback is required here, not only to inform the user what is happening (and what can happen next), but also to guide the user into the appropriate actions for error recovery. This was done in the ISDIP project by developing a set of user-machine dialogues which were initiated by the system (see section 2.2.3 below). Various different dialogue types were produced, and these were used with novice users in evaluation experiments. These are discussed in section 2.3 below.

2.2 Solutions for speech input systems

2.2.1 Solutions to recognition problems. As indicated in section 2.1.1. above, solutions in this area are necessary for the correction of 'wrong' output items, whether they arise from system error (where the system incorrectly identifies an input item) or from user error (where the user changes his mind about the item he wants after it has been output to the screen). In the ISDIP project we developed two different techniques for dealing with this range of errors - we refer to them as *system repairs* and *user repairs*, to indicate whether they are initiated by the system or the user.

2.2.1.1 System repairs. These are used in cases where the system 'knows' there is a problem because the match between the input item and the trained template falls below the certainty level (an adjustable parameter which can be set very low for small vocabulary applications where a very few items are easily distinguished, but must be high for large vocabulary applications such as word processing, where there is a large potential for confusion). Below this level of certainty we programmed the system to output a screen message to the user indicating that the system was 'not sure', and offering the best match from the current vocabulary. There was also a prompt to the user to say "yes" if the item was correct and "no" if it was incorrect. If it was correct, the item was output to the screen and the user was free to input the next item. If it was incorrect the system offered the next best match, again with a yes/no option. If that was incorrect then the user was required to input the item again.

This kind of strategy is essentially an 'on the fly' solution to the problem of misrecognitions. In addition to this we also substantial work in the area of misrecognition prevention rather than repair, in order to reduce the number of occasions when system repairs would be necessary. This more fundamental work falls into the general area of machine intelligence, and is described in more detail in section 2.4 below.

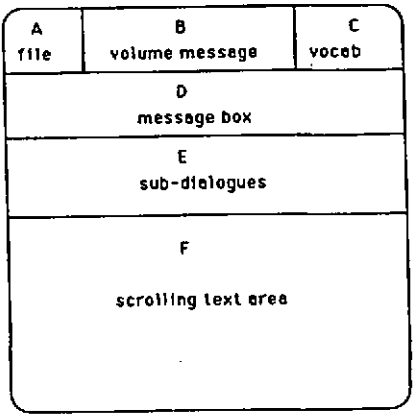
2.2.1.2 User repairs. These are used in two situations - firstly where the user wants to change his mind about the last input item, and secondly where there has been a total misrecognition - i.e. where the system has got it wrong and is 'unaware' of it. A very simple provision was made for this - an "undo" command, which reversed the last action. This was not a progressive 'undo', to allow the user to move back through a document undoing strings of previous input items, it was a toggle undo/redo command to allow him to change his mind and then change it back. (Corrections over previous stretches of text were done by editing functions, such as deleting, cursor movements etc.)

2.2.2 Solutions to feedback problems

The immediate solution to the problem of the user forgetting which vocabulary item was stored where was implemented by two techniques - *permanent feedback* and *user-request feedback*.

2.2.2.1 Permanent feedback. We provided for this by building into our screen design a slot which permanently showed the current, live vocabulary (section C in figure 1 below). As the user switched between vocabularies this slot was updated to show the name of whichever vocabulary was currently being addressed by the recogniser.

Figure 1.



2.2.2.2 User-request feedback. This was a "show keys" command, which allowed the user to view all the items in the current vocabulary. The vocabularies were structured hierarchically, with a "switch" vocabulary at the highest level, so that the user was required to move from one vocabulary to another via *switch*. The "show keys" command was operational for all vocabularies, including *switch*, so that if the user forgot the name of a particular vocabulary where an item was stored, he could simply say "switch", thereby accessing the *switch* vocabulary, and then "show keys" to display all the other vocabularies in the system.

The work carried out on system intelligence was designed to remove the bulk of the responsibility for switching between vocabularies from the user and onto the system, so releasing the user from having to remember vocabulary names and location of templates. This is discussed more fully in section 2.4 below.

2.2.3 Solutions to usability problems. Many of the usability problems were solved by the simple process of providing permanent and user-requested feedback to the user, both of which functioned to 'plug gaps' in the user's memory of how the system was structured. In addition to this, however, we also developed techniques designed to lead the user in a more positive way to exploit the full potential of the system. This involved developing machine initiated dialogues, which were designed to be simple and unambiguous for the user, while providing full information as to what the user was required to do next. This is a potentially very problematical area, as users' reactions to what the designers consider to be excellent dialogues can be disconcertingly unpredictable. Being aware of this, our method in this section of the project was to rely totally on empirical testing, bringing new users of the system in to carry out predefined exercises using the ISDIP prototype with a variety of different dialogues, in order to discover what level of human-machine communication was best received. For more details of this process and the conclusions we drew from it see section 2.3.

2.3 User evaluations of the ISDIP prototypes

As the prototypes were developed, each version was tested with volunteer new users, totally unfamiliar with the system. They were given exercises to do using the speech driven word processor, in the form of dictating a letter from a prepared text. After each test machine misrecognitions and user errors were counted, the user's ability to operate the repair mechanisms was noted, and the user was asked how confident he/she felt about operating the system.

A number of points were noted regarding users' reactions to the automated dialogues, and it became clear that the system's performance in terms of number of errors was at variance with users' *perceptions of that performance*. In cases where a 'partial misrecognition' occurred - i.e. where the input item was imperfectly matched with a template, and the system was 'not sure', and therefore consulted the user before printing the item to the screen - the users did not perceive this as an error. The dialogue between the system and the user in such cases transformed (for the user) a failure into a success, and this inevitably played an important part in building up user confidence in the system.

Many of the volunteers were secretarial and clerical workers with advanced touch-typing skills, and the expectation was that they would dislike the system as it was slow compared with their usual input by keyboard (the top rate by an experienced user was 36 words a minute, which included a few machine and user-initiated repairs). We found, however, that all those tested enjoyed using the system and quickly felt confident in dealing with it. There were no 'failures' - everyone was able to complete the task without intervention by a team member. The users who were disabled included people who had previously had to use such devices as head wands and mouth sticks in order to operate the computer. This group were particularly impressed with the voice driven system, as it is considerably less tiring than other aids for the disabled, and has the added benefit of using a 'normal' form of input, bringing them more in line with able bodied users.

2.4 System intelligence

As indicated above, one of our major concerns within the project was to reduce the cognitive load on the user with regard to accessing vocabularies and the inbuilt problems of remembering which items were stored where. As a way of achieving this we developed a method of building some intelligence into the system by customising the vocabulary structure for specific domains/users.

The first step in this procedure was to obtain a corpus of typical texts from the target user, and subject this corpus to linguistic analysis in order to discover both the specialised vocabulary and the recurring discoursal patterns (the sequences of items which characterise the text). This section of the work was done using the Oxford Concordance Program. We then structured the system vocabularies to include the specialised items, and set up a vocabulary manager which, having successfully recognised one input item, automatically loaded the vocabulary containing the most likely next item (given the user's usual practice). This not only removed responsibility for switching from the user to the system, it also improved the overall performance time, as the system was not 'waiting' for the user to do the switching.

3. CURRENT WORK: THE CALE PROJECT

3.1 Project aims and CALE end users

Our aim in the new CALE project (Computer Aided Learning Enhancement) is to provide a totally hands free, voice driven computer system for disabled school children. We are particularly concerned, in this project, to provide maximum flexibility in the system, so that the end users can access the full range of standard educational software, and one of our top priorities is to ensure that the system will run on low cost, standard school hardware, in order to maximise the number of students who can benefit from the system. We have an initial target group of end users already involved in the project - the children from Lonsdale School for the Disabled in Stevenage, Hertfordshire. These children suffer from a range of disabilities, including various kinds of paralysis, cerebral palsy, and muscular dystrophy, and they also all have some degree of learning difficulties. A first prototype is already in use in Lonsdale, and our final system will be installed there by Easter 1994.

3.2 The ISDIP and CALE projects - a change of focus

The ISDIP project was primarily an investigative, research project, in which the production of a working system served to test out hypotheses relating to requirements for speech input systems and interface design. CALE, on the other hand, is concerned to develop an effective, affordable system for use by a clearly identified group of end users. Because the final system is required to be usable across the full range of standard software our priority is to provide maximum flexibility, even though this inevitably implies some degree of trade-off in the area of simplicity in the user interface.

3.1 Early design decisions and constraints

As the target group of end users are school children, and our brief is to provide a system for Lonsdale School, we were not free to choose the basic platform to work on, but were obliged to use a machine which was already in use at Lonsdale. This was the Nimbus pc 386. Although this is clearly a constraint, it is a long term benefit for the project, as Nimbus machines are common in schools (we are carrying out a survey of schools in the South East in order to establish how common), and this ensures that the system will have wide applicability and usefulness throughout a wide range of schools coping with disabled pupils.

As to the underlying recognition hardware, we considered various systems, including DragonDictate and the Marconi system. We decided, however, on the basis of cost, transportability, and suitability for disabled use, on the IBM VoiceType. As this is a new system there was some initial doubts as to whether this would run on the Nimbus, but there has been no problem with either installation or operation, and to date we have found no incompatibility of any kind between the machine and the VoiceType system. Like many of the more modern systems, VoiceType is not subject to the constraints of the older designs, and there is no need for sectioning of the vocabulary. From the point of view of the user, all the vocabulary is available all the time. Another advantage of this kind of system is that there is no requirement for the user to train the whole vocabulary before beginning to use it. The bulk of the training is done *while* using the system, which constantly updates its voice files.

3.2 Customisation of VoiceType for the CALE system

Unlike our previous, in-house system developed during the ISDIP project, VoiceType is an extremely powerful and flexible system. This means, of course, that the user interface is inevitably very complex, in order to allow for the many options which VoiceType provides. For school use we need to preserve this wide functionality, but in order to accommodate naive users (and particularly those with learning difficulties) we are also prioritising customisation of the system in order to present a simpler interface.

Customisation of a complex, powerful system, in order to simplify the user interface is, in a sense, the reverse process of what we did during the ISDIP project. At that time we began with a very simple interface overlying a very simple system, and gradually added functionality as work progressed, and in response to user evaluations. Our task in CALE is to *conceal* some of the functionality from the user, in order to simplify the interface, without great reduction in the flexible power of the system. We are doing this initially by exploiting the facility offered by VoiceType to construct specialised macros (purely by voice), so as to cut out some of the steps which would otherwise be necessary for moving between applications and initiating various computer actions.

The project is being managed on an incremental prototyping basis, so that each version is transferred to the school as it is completed, and is there subjected to testing by using it 'for real' over a period of weeks. The user evaluations resulting from each testing phase are then fed back into the design cycle and are built in to the next prototype. Along with evaluations of the current prototype we are also getting printed output of the work the children have done using the system. This will enable us to make further customisations in terms of the user vocabulary, which we will be tailoring to match their requirements. For more general school use this step would not be so important, as there are ample facilities provided by VoiceType to allow the user to customise the vocabulary to include his chosen items. In the case of children with learning difficulties, however, customisation at this level will be of great benefit, particularly in cases where the child's spelling is very bad and they may be likely to add an item spelt incorrectly.

4. FUTURE WORK

Along with our current work for disabled school children, we are also investigating the possibilities for developing a more generally useful speech driven system for disabled adults (whether wishing to use a computer system in the work environment or at home). To be useful to a wide range of disabled people, this will require a different approach from the one we have adopted in the school project, as the expectation is that there will be a greater variety in the software packages the user may wish to access. This will almost certainly mean that the interface will need to be different - it may be, for example, that some of the functionality we are at present concealing in order to accommodate our school population will need to be reactivated.

5. BIBLIOGRAPHY

Cheepen, C., J.Monaghan, J.Hewitt & C.Hunter, 1989, "User and application simulation in the evaluation of an intelligent speech interface", **Proceedings of Simulation Conference**, The Ergonomics Society, Brighton, May

Cheepen, C., 1991, "The use of corpora in the design of an intelligent speech interface to a word processor", **Linguistics Association of Great Britain Spring Meeting**, Oxford, March

Cheepen, C. & J.Monaghan, 1991 "A system for reducing imprecision in speech interfaces to generalised text input devices", **Proceedings of RIAO '91, Intelligent text and image handling**, Barcelona, May

Monaghan, J. & C.Cheepen, 1991, "Dialogue design for speech interfaces in the office context", **HCI International '91**, Stuttgart, September

Richards, M.A. & K.M.Underwood, 1984, "How should people and computers speak to each other?", **Interact '84**

Zajicek, M., 1989, "An investigation into evaluation criteria for speech interfaces", MSc Thesis, Hatfield Polytechnic