Natural Language Engineering **4** (1): 57–71. Printed in the United Kingdom

© 1998 Cambridge University Press

A script-based AAC system for transactional interaction

RICHARD DYE, NORMAN ALM, JOHN L. ARNOTT, GILLIAN HARPER and ARTHUR I. MORRISON

Applied Computing, University of Dundee, Dundee, Scotland
e-mail: {rdye,nalm,jarnott,gharper}@mic.dundee.ac.uk

(*Received 10 September 1997*)

Abstract

Non-speaking people often rely on AAC (Augmentative and Alternative Communication) devices to assist them to communicate. These AAC devices are slow to operate, however, and as a result conversations can be very difficult and frequently break down. This is especially the case when the conversation partner is unfamiliar with this method of communication, and is a big obstacle to many people when they wish to conduct simple everyday transactions. A way of improving the performance of AAC devices by using scripts is discussed. A prototype system to test this idea was constructed, and a preliminary experiment performed with promising results. A practical AAC device which incorporates scripts was then developed, and is described.

1 Introduction

People who are severely physically disabled and are non-speaking have formidable challenges in communicating with others. Their physical capabilities can vary widely – many have very limited muscle control, and some can only interact with the world through the operation of a switch. To communicate, these people usually rely on some form of communication aid or system to interact with others. Depending on literacy ability (approximately 60% have some literacy problem), these communication aids may be text- or symbol-based. Speed of communication through a communication system can be very low, however, particularly when the physical disability is severe. Word rates which can be achieved with existing technology usually lie in the range 2–10 words per minute (Beukelman and Mirenda 1992). Conversations can thus be difficult to conduct and prone to breakdown. This is especially the case when the conversation partner is unfamiliar with this method of communication, and is a big obstacle to non-speaking people when they wish to meet people and interact with them.

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

2 Existing AAC devices

A basic form of AAC device consists of a word board or display panel on which a number of language items are arranged in a grid layout, and the user communicates by pointing to the language item he or she wishes to communicate. The communication partner then reads the language item directly from the word board or display panel. With the advent of microcomputers and speech synthesisers, it became possible for these word boards to become dynamic and for the user to be able to 'speak' with a synthetic voice. The resulting systems were still far from ideal, however. It takes a lot of time and effort for a user to use an AAC device to produce each speech act. This is because the speech acts usually have to be constructed word by word. The result can be dysfunctional communication caused by long pauses and low communication rate.

A number of techniques have been developed to try to improve the efficiency of use of AAC devices, so that users might be able to interact more effectively with their communication partners. Word dictionaries with optimal search methods (Woltosz 1988) have been applied to improve the situation, and automatic expansion of minimal text into full sentences using sentence compansion has also been tried (Demasco and McCoy 1992; McCoy *et al.* 1994). Word prediction systems have been investigated (Beattie *et al.* 1994; Swiffin *et al.* 1987; Tyvand and Demasco 1992) in the context of reducing the physical work-load and, in some cases, improving word rates for very slow AAC users. These techniques are not sufficient by themselves to enable users to participate in normal dialogues, however, and further attempts to improve the situation have centred on providing pre-stored utterances that the user can access quickly. Devices which do this rely on the user remembering where the pre-stored utterances are stored in the system, and how to select them. Mnemonic systems (e.g. semantic compaction) can be used to help users remember how to select stored phrases (Baker 1982; Levelt 1994), but these require considerable training (often years) to use the devices efficiently, and some users never manage to master them completely. Prediction of sentences from a large pre-stored collection has been investigated (Broumley *et al.* 1990), with the prediction based on user's interests, profile of frequent conversation partners and current conversation topics. Another system was developed to facilitate story-telling (Waller 1992), and phatic communication (using speech for social interaction and the communication of feelings) has been automated (Alm *et al.* 1992) so that the user can rapidly produce speech acts for the opening, small-talk and closing phases of a conversation. Features of such systems are available in commercial form (Don Johnston Inc.) as a result of research work in this area, and bring a number of advantages to AAC users. It has been shown that the content of computer-aided conversations in an AAC setting can rate well in comparison to that of unaided dialogue (Todman *et al.* 1995; Todman and Lewins 1996).

Despite these advances, there remains a need for improved AAC techniques to enable users to participate in lively, fast moving dialogues with a wide range of conversation partners, including people whom they have never met before. One of the goals of AAC research is to improve the design of AAC devices to allow

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

their users to communicate more effectively. An ideal AAC device would have the following characteristics:

- production of speech acts quickly with the minimum of physical effort;
- low cognitive load during use; and
- minimal requirement for training.

3 Scripts in AAC

This paper describes a new method of organising pre-stored utterances to attempt to overcome the memory load problem. It is proposed that a user's memory load can be reduced by making use of their existing long term memory to help them locate and select appropriate utterances from the AAC system. Schank and Abelson (1977) proposed a theory that people remember frequently encountered situations in structures in long term memory which they called 'Scripts'. A script captures the essence of a stereotypical situation, and allows people to make sense of what is happening in a particular situation, and to predict what will happen next. The AAC potential of the scripts concept as proposed by Schank and Abelson has been recognised for some time (Kraat 1987), and the project described here is part of an investigation into their application in AAC. Other research (Vanderheyden *et al.* 1996) has shown the potential that similar schema-based techniques offer to AAC; preliminary experimentation has been encouraging and justifies further research on their use.

Scripts are most likely to be useful in transactional type conversations, as these tend to be very structured, and predictable in content and flow. Many AAC users find this type of conversation particularly difficult as it very often involves them talking with strangers who are not familiar with their particular method of communicating. It is proposed that if pre-stored utterances are stored within script-type structures inside an AAC device, then a user could use their own memory to assist them to quickly access these scripts and thereby find appropriate utterances. A suitable user interface for the device would be needed to realise this concept.

4 Initial experiment

An initial experiment was devised (Alm *et al.* 1995a) to investigate the potential of a script-based approach to transactional interactions in AAC, and a prototype AAC system was developed to facilitate this experiment. The aim was to ascertain whether or not a transactional interaction could be conducted using a script-based communication system. It was decided to simulate a particular transactional interaction which could be reasonably expected to follow a predictable sequence of events, i.e. one which would be amenable to the script approach, to find out whether a computer-based script could enable a successful interaction to happen.

The transaction chosen for the experiment was that of arranging the repair of a household appliance over the telephone. A script was devised which contained all the stages of a typical telephone interaction in which someone calls a maintenance company to request that they come to repair a domestic appliance (in this case, a

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

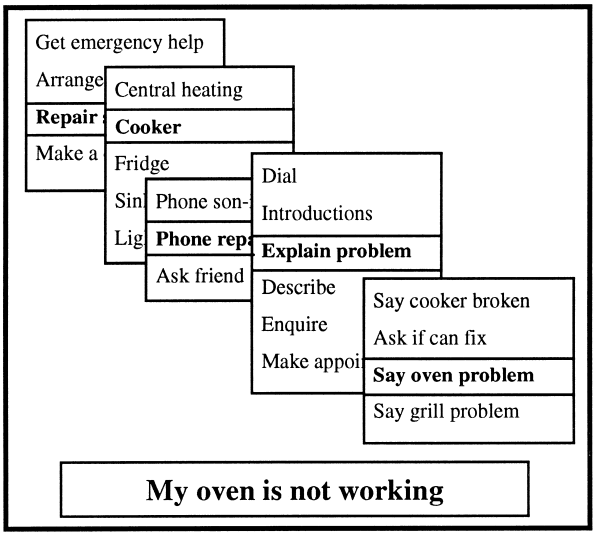


Fig. 1. A snap-shot of the user interface in use. The user moves through a goal-plan-script hierarchy to finally select the wanted utterance.

cooker (kitchen stove) was chosen as the example appliance). The caller was assumed to be a non-speaking person, trying to interact with the appliance repair person at the maintenance company. The script was devised to cope with contingencies which might arise in the interaction with the repair person, and no facilities were provided for the user to type new material.

Phrases were composed for each stage of the interaction which the user could select easily using a simple text-based interface (Figure 1). This enabled the user to identify the topic which needed to be discussed, then to step through the interaction. Keystrokes (on the cursor control keys) were used to navigate pop-up menus within this simple interface. In the example in Figure 1, the user has used the first three successive menus to identify that the cooker (stove) needs repaired and that the repair company should be telephoned. The next menu offers a sequential script for the telephone conversation, including introductions, explanation and description of the problem, enquiry of availability, and appointment making. The user can step through these stages one-by-one, with the system producing a phrase for each stage, as the conversation progresses. In Figure 1, the user is at the 'Explain Problem' stage, and this has invoked a menu of options for explaining the nature of the problem. The user has gone on to select 'Say Oven Problem', and the system has therefore presented 'My oven is not working' for output via synthetic speech. The user would move back thereafter to the previous menu to select 'Describe' in anticipation of giving further description of the oven problem. The system contained phrases for all of the statement options which the system could offer about the cooker, and these were structured into sequential scripts to facilitate the user's progress through the interaction. The system also contained eight utility phrases such as "You are speaking to a disabled person using a computer to speak", "Please can you repeat

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

that", "Yes" and "No" which could be selected directly using function keys on the keyboard.

An experimental situation was devised in which a subject was telephoned in order to arrange the repair of a household appliance. Forty subjects, chosen at random, were asked to play the role of the appliance repair person at the maintenance company. None of the subjects had any previous experience in communicating with someone using an AAC device. The subjects were divided into two groups. With one group the non-disabled experimenter used his own voice to perform the task, i.e. he conducted a conventional telephone conversation, and with the other group he used the script-based prototype system as sole output medium. The conversations were recorded and subsequently transcribed for analysis.

5 Results

Every subject managed to complete the required task. One subject had some difficulty completing the task because of the poor quality of the synthetic speech. All of the conversations were timed, and the computer-mediated conversations were found to take on average approximately 50% longer than the natural conversations. (The mean length of calls made using the script-based prototype device was 150 seconds, with a standard deviation of 26 seconds. The mean length of calls made with normal voice was 102 seconds, with a standard deviation of 14 seconds.)

In a typical script-based conversation, the experimenter needed to use one quarter of the key-strokes which would have been required to type all of the words produced by the script system. (The numbers of key-strokes which would have been required to type the script phrases character-by-character were counted.) All of the key-strokes made in the script-based conversations were used to control the script system through the simple menu-based interface. Although the script interface was a relatively simple one devised for the purpose of this experiment, this represents a significant saving in the amount of physical effort required. Further refinement of the interface, including the incorporation of alternative access methods, could produce improved efficiency and ease of use for users with disabilities.

A small number of the subjects commented on the pauses during computer-mediated conversation. The average time taken for the experimenter to respond to an initiator from the subject was 4.94 seconds (standard deviation 2.19 seconds). One subject said it was "rather like a long distance call – pauses". These pauses are on average longer than the three seconds 'awkwardness limen' identified by McLaughlin and Cody (1982) as being the boundary between acceptable and unacceptable delays in conversation, so the pauses were likely to be noticed. They did not appear, however, to have a serious effect on the conversations.

Although the experiment was conducted by a non-disabled person, who was constrained to requesting one form of assistance over the telephone, these results were sufficiently successful to encourage further investigation of scripts within AAC. It was particularly encouraging that the experimenter could achieve a functional goal (requesting a domestic repair) using only a script-driven system with synthetic speech output when telephoning inexperienced subjects. This prompted the view

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

that scripts could play a very valuable role during functional interaction within a general AAC system. It was therefore decided to pursue the concept more generally by developing an experimental script-based AAC system to enable users to conduct functional interactions and general conversation in a number of everyday tasks and situations.

6 The realisation of a practical device

To take this work further a collaborative project was started to incorporate scripts into a more practical device, involving IRV (a research institute in The Netherlands), and two commercial companies (Kompagne BV, The Netherlands and IGEL GmbH, Germany) and was funded through the European TIDE programme.

A practical AAC device cannot be restricted for use in a limited number of transactional situations. It was therefore decided to embody the concept of scripts into a more general model of conversation, which accounts for the opening and closing stages of an encounter, the giving of feedback remarks to another speaker, and the creation of unique text. This model incorporates the results of previous research projects at Dundee University on conversational modelling (Alm *et al.* 1992) and word prediction (Swiffin *et al.* 1987). A group of AAC users was set up to advise during the design process (Alm *et al.* 1995b).

The AAC device was implemented as a hardware/software system. The hardware chosen was standard PC laptop technology as specially developed AAC platforms using this technology are already on the market. The software was developed using Visual C++ and runs within the Microsoft Windows operating system. The device developed was given the name ScripTalker.

The user interface is made up of three main components: rapidly produced speech acts, the scripts and a unique text facility. The rapid speech act component contains high frequency utterances used in the opening and closing portions of a conversation and in giving feedback, and consists of groups of speech-act buttons. The Scripts component is used in the discussion phase of a conversation, and consists of a set of scripts with which the user can interact. The Unique utterance component is used when there are no appropriate pre-stored utterances available, and consists of a virtual keyboard, a word prediction mechanism and a notebook facility. To cope with the limited literacy of many AAC users it was decided that the user interface would as far as possible be graphically based.

6.1 Rapidly produced speech acts

To assist users in performing high frequency speech acts, a rapid speech acts facility is provided in the system. The facility supports a range of speech acts used in opening and closing an encounter, giving feedback comments to another speaker, and controlling the conversation. The design intention was for the facility to produce these speech acts as quickly as possible, without the need to bring up sub-menus. This facility is based on previous work with CHAT (Alm *et al.* 1992) which provided users with just these speech act possibilities in an attempt to gauge the feasibility

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

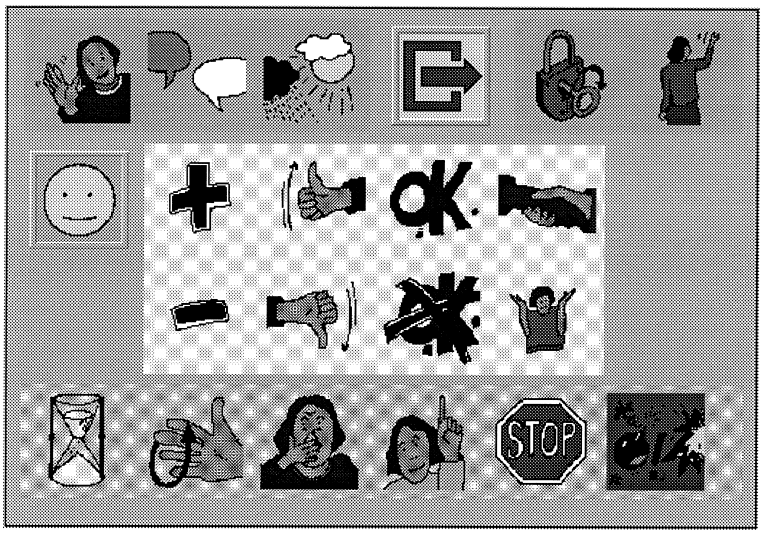


Fig. 2. Rapid speech act user interface.

The speech acts and functions represented by the above icons are as follows:
Row 1: Openers(Greeting, Response, Small-talk), Exit, Closers(Wrap-up, Farewell)
Row 2: Mood button, Feedback(Yes, That' good, Agree, Thank you,
Row 3: No, That's bad, Disagree, Don't know)
Row 4: Control(Please wait, Repeat, Oops, Interrupt, You misunderstand, Expletive)

of emphasising speed of response over specificity in aided conversations. Trials of the prototype showed that reasonable sounding interactions of particular types were possible to achieve at greatly enhanced rates. Users of current systems communicate at rates of about 2–10 words per minute (Beukelman and Mirenda 1992). Users of the prototype achieved rates of between 19 and 54 words per minute. Both users and conversation partners reported positively on the impact of the system on the quality of the interaction (Alm *et al.* 1992).

In the script-based communication system, the interface for the rapid speech act facility consists of a number of buttons; one for each of the speech-acts being modelled. The speech-act buttons are arranged into five groups (see Figure 2). The speech acts are grouped according to the function they perform in conversation. The group categories are: I'm listening, Openers, Closers, Feedback and Conversation controllers. A speech-act button contains a set of five utterances, each of which is worded to have the same effect on the listener when spoken. When a user selects a speech act to be spoken, an utterance is chosen randomly from the contained utterances of that category. This provides some variability and helps prevent the user sounding repetitive. Table 1 details the speech-act buttons provided and how they are grouped together.

A set of icons to represent the speech acts within the interface was decided upon. To investigate how appropriate these icons were, the user advisory group in Dundee was asked to identify the icon that they felt most closely represented each of the

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

Table 1. *Speech-act buttons*

Group	Speech-act buttons
I'm listening:	Uh-huh
Openers:	Greetings, Responses and Small talk
Closers:	Wrap-up, Farewells
Feedback:	Yes, No, Good, Bad, Agree, Disagree, Thank you
Control:	Oops, Please wait, Tell me more, May I interrupt?, You misunderstand

speech acts. This exercise showed that there was little confusion between the majority of icons.

A mood button was implemented to allow the utterances selected to reflect the users' mood. The mood button allows one of four moods to be selected. The moods available for selection are: polite (the default mood), informal, humorous and angry moods.

When appropiate the interface will predict the next speech act the user is likely to output given the last speech act spoken. For example, after giving a greeting the interface will predict that a response to a greeting will most likely follow.

6.2 Scripts

To provide access to a set of scripts, a scene-based user interface was devised. The scene approach was taken in order to give users a pictorial indication of subject matter, and to assist users with varying levels of literacy skills. In this interface, scripts are presented to the user as a sequence of cartoon-style scenes. The scenes give the user an indication of the subject matter and purpose of the script, and assist the user to assess whether the script is appropriate for current needs. Each scene is populated with realistic objects chosen to represent the conversation tasks that can be performed. The user thus receives a pictorial overview of the script, what happens in it, and the options that are available. This assists the user to see quickly what the script will be able to do in the context of the current conversation. An example of a scene can be seen in Figure 3, which shows a scene within the doctor script.

Many users of AAC systems have literacy problems and use symbol-based communication systems. The pictorial interface was devised to be accessible to users with varying levels of literacy skills. Research into picture recognition and memory structures has demonstrated that groups of objects organised into realistic scenes corresponding to stereotypical situations better assist recognition and memory compared to groups of arbitrarily placed objects (Mandler and Parker 1976; Mandler 1979; Mandler 1984). A scene-based interface using a realistic arrangement of objects within a scene was therefore chosen to facilitate recognition and remembering by the user and thus reduce the cognitive load required to locate suitable objects during a conversation.

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

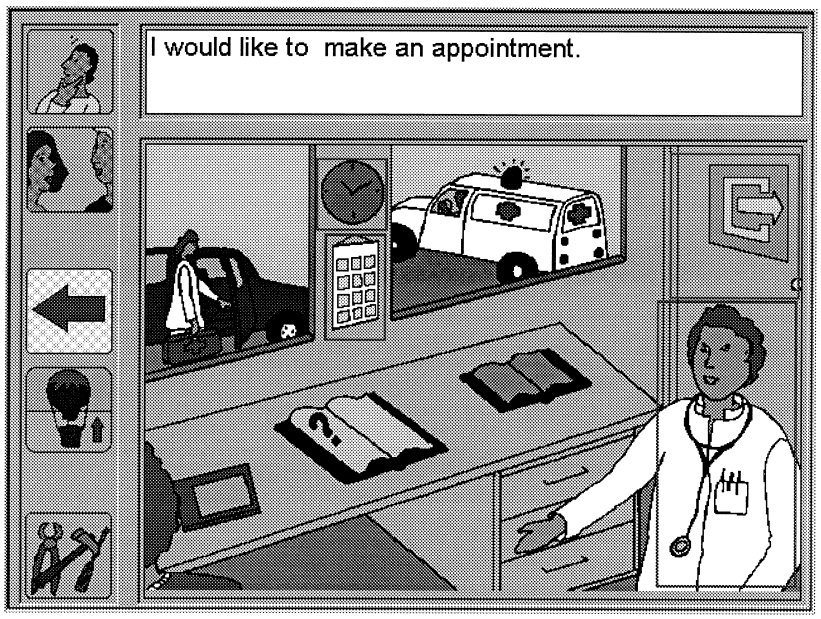


Fig. 3. The Script user interface showing a scene from the doctor's script.

A text preview and display box appears at the top of the user interface. The main interface area (bottom right) contains the scene image. The function buttons on the left side of the interface are, from top to bottom: 'I'm listening' rapid speech act button; button to access the main rapid speech act interface; scene navigation backtrack to previous scenes button; scene navigation overview button; tool button to access the notepad and additional system control facilities.

Several types of objects exist, some allow the user to switch between scenes and others allow the user to speak useful utterances. Special objects allow the user to speak detailed information regarding times, dates and numbers.

As it would be impractical to provide scripts for every conceivable situation, it was decided to provide users with a limited number of scripts together with an authoring package with which they can develop their own custom scripts with help from their therapists.

Access to all the scripts is provided through a 'town plan' scene. To move from one script to another requires the users to navigate through the 'town plan'. To make it easy to locate the 'town plan', a special control, the overview button, was provided.

The phatic communication and unique text facilities are also available to the user at all times should it be necessary to say something which is not available in the script. It is always possible that an unpredictable event will occur which will require deviation from the typical script and is therefore necessary for the system to offer flexibility, so that the user can respond to unforeseen events and maintain the flow of conversation.

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

6.2.1 Development of scripts

To begin with it was decided to develop six complete scripts. These were chosen after discussions with the user advisory group about situations in which they found difficulty communicating. The scripts developed were: 'at the doctor', 'at the restaurant', 'going shopping', 'activities of daily living', 'on the telephone', 'meeting someone new' and 'talking about emotions'.

The scripts were developed by considering all the conversation tasks or goals the user would wish to achieve in the given situation. An utterance was then generated for each conversation task. The utterances were grouped into sequences and scenes and then mapped onto suitable props. It was decided to control the complexity of the scenes to make them easy to use. Users with physical disabilities would find complex scenes difficult to use as mouse and rollerball users have difficulty in accurately pointing to small objects, and switch users will take a long time to scan through a large number of objects. Complex scenes could also be a problem to users with visual problems, and would take longer for all users to learn to use. It was therefore decided that, wherever possible, scenes would be designed to contain at most nine objects as this is approximately the limit of human short term memory.

6.2.2 Prediction within scripts

The system uses the script to guide the user through a dialogue. There is a prediction mechanism which predicts the next most probable stage in the dialogue which the user will need (based on the script), so the user can usually follow a predicted path through a conversation. This prediction mechanism monitors the sequences of objects selected and uses this information to modify future predictions.

On entering a scene the user is offered a prediction of the first utterance or action he or she is likely to require within the context of the interaction. The prediction is indicated by a distinctive prediction icon which is displayed over the object that will produce the utterance or action. The prediction mechanism monitors the objects selected by the user and offers a prediction of the next stage based on the last object selected. The prediction icon moves to indicate the next predicted object. The user can choose to select objects in an unconventional order however, if a particular dialogue situation requires it. If the user selects a different object and deviates from the typical pattern, the next prediction will be based on the actual selected object.

A simple example of the use of the scene and script prediction in Figure 3 (Reception scene in the Doctor script) is "Making an appointment to see a doctor". The typical sequence of events would be for the user to introduce him- or herself to the person at the reception desk, ask to make an appointment, negotiate a day and time for the appointment and then leave.

On entering the scene the prediction icon would be displayed over the 'user' image in the bottom left corner. On selection this produces the utterance "Hello, my name is [_username_], I can only talk using this computer so please be patient". The prediction icon then moves to the 'appointment book' object predicting the utterance "I would like to make an appointment". The receptionist is now likely to

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

ask for a preferred day and time and therefore the prediction moves to the 'calendar' object which is used to specify a preferred day. The prediction now moves to the 'clock' object to assist the user in specifying a preferred time. At any stage during the interaction the rapid speech acts facility can be used for feedback remarks such as yes, no and thank you. At the end of the appointment negotiation phase the prediction moves to indicate the 'rapid speech acts' button which the user selects to invoke the rapid speech acts facility in order to say thank you and goodbye.

6.2.3 Templates

In addition to pre-stored utterances, ScripTalker was designed to include pre-stored utterance templates. An utterance template consists of a pre-stored phrase with an empty slot that can be filled in and spoken during a conversation. The purpose of templates is to make available utility phrases which can be modified easily by insertion of key words into an empty slot. An example of a template might be "I would like to buy [__] tickets, please". In this example, the user could use a number pad to complete the utterance. This approach is useful because it can reduce the number of similar utterances that need to be stored and browsed by the user and allow stereotypical types of utterances to be modified dynamically to suit the current situation. This approach has been shown to be useful in work by Vanderheyden *et al.* (1996).

6.3 Unique utterances

The Unique component of the user interface consists of a keyboard (the keyboard can be real or 'on screen'), word prediction and a notebook facility for storing and retrieving previously prepared utterances.

6.3.1 Word prediction

To assist users to enter unique text strings a word prediction mechanism was provided. The word predictions made are based on initial characters typed and on the recency and frequency of use of words contained in a dictionary. The predictive mechanism used is comparable to that used by many typical word predictors, e.g. the PAL system developed by Swiffin *et at*. (1987). When using this type of word predictor, key savings of up to 50% are possible.

The ScripTalker word predictor automatically captures new words and purges itself of old infrequently used words whenever the dictionary becomes larger than a pre-defined size. A dictionary of 10,000 words can be used with no perceptible delays on a typical laptop PC. With this size of dictionary, users can adopt sophisticated strategies in order to try and improve their speech rates. For example, when a user wishes to communicate a compound word such as 'something', they can choose to select word 'some' if it is predicted first and then backspace, the system will then offer predictions like 'someone', 'something' and 'somebody'. Figure 4, shows the on-screen keyboard displaying predicted words.

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

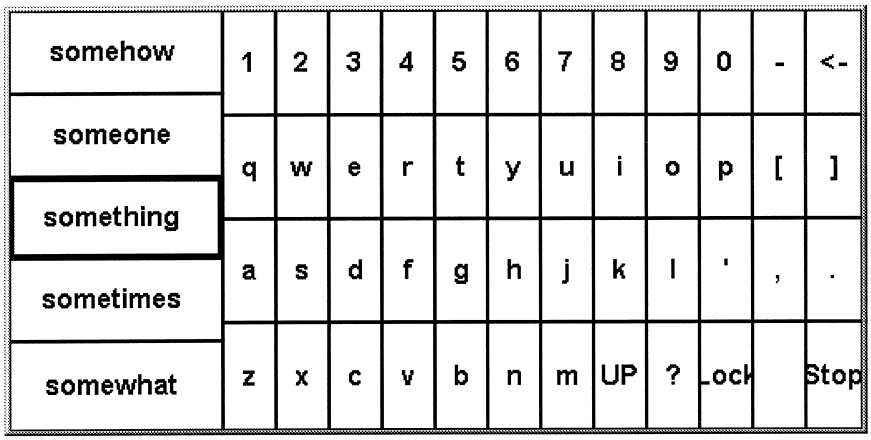


Fig. 4. The on-screen keyboard and word prediction. The keyboard can be operated using a switch with automatic scanning of rows and columns.

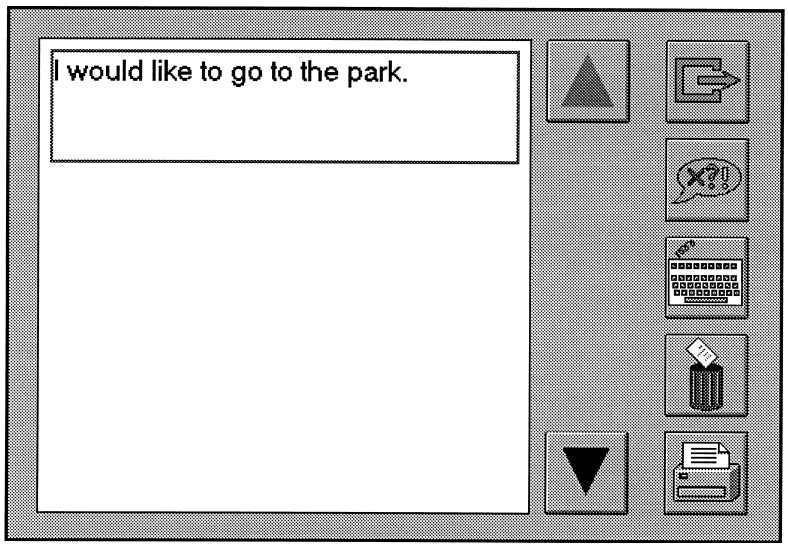


Fig. 5. The notebook: functions are available to speak, add, delete and print notes.

6.3.2 *The Notebook*

To allow users to prepare utterances in advance of a conversation a Notebook facility is provided (see Figure 5). The Notebook interface is activated through the tool bar, and temporarily covers the area of the screen previously occupied by a scene. The Notebook allows the user to leave prepared utterances at appropriate places within the scene stucture. To leave an utterance at a particular scene, the user first locates the required scene and then activates the Notebook to enter the utterance. After the utterance has been entered the Notebook disappears to be replaced by the scene again. The next time the user visits the same scene and activates the Notebook, the

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

utterance left the previous time is displayed. This utterance can then be spoken if required.

The Notebook can also be used to complete template utterances attached to particular scenes. For example, the template "I would like to speak to [_..]" is attached to the 'making a telephone connection' scene of the telephone script, and can be completed using the Notebook. A hotspot of type 'fill slot in template using the Notebook' is used to activate the notebook from the scene. After the Notebook is activated, selecting an item completes the template which can then be spoken. In this case the items are likely to consist of a list of peoples names that a user has entered previously.

6.4 Authoring

Scripts are organised as ordered sequences of scenes or graphical representations of situations. A scene consists of a background picture with props superimposed on it. Props have an active area, or hot-spot, which the user can click to perform a particular task (e.g. to speak a message or to display the next scene in the sequence). Scripts, scenes, props, and hot-spots are represented by a set of tables in the script database and a number of bitmap files.

The authoring system for users and therapists allows new custom scripts to be created. After creating a new script a number of scenes can be created which are automatically linked together in a sequence. For each scene created the user can select a background image from a limited number supplied. The scene background image can then be populated with objects chosen from a library of objects. Hot-spots are automatically generated over the object images. Utterances can then be linked to these hot-spots to complete the scene. Links from hot-spots to existing scenes and special objects can also be specified.

The development of a practical device is nearing completion. The device is currently undergoing evaluation with disabled non-speaking people. The commercial companies involved in the project intend to produce a commercial AAC system based on this research.

7 Conclusions

Using current AAC devices, non-speaking people can still have great difficulty in conducting conversations. This is especially true when their conversation partner is unfamiliar with their way of communicating. This is a big obstacle to many people when they wish to get out and about, as this will often mean talking to strangers when conducting everyday transactions. It was proposed that incorporating scripts within an AAC device could help solve this problem. A prototype device was constructed, and the potential of this device was demonstrated experimentally. Using a script-based AAC system a non-disabled person was able to conduct a transaction over the telephone with a group of subjects. The time taken to conduct the conversations was on average only 50% longer than similar conversations conducted with another group of subjects using the person's natural voice.

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

Following from this study, a practical AAC device which can be used by a wide range of users was implemented. The device contains a set of useful everyday scripts. If necessary, users can easily add to this set of scripts by using a script authoring system. To allow users with literacy problems access to the scripts a scene-based user interface was developed. In this interface the scenes within a script are displayed pictorially, and users 'speak' by interacting with objects in the scenes.

Acknowledgements

The development of a practical AAC device was supported by the TIDE Programme of the Commission of the European Union under the ALADIN project (project no. TP1035). The project partners were The University of Dundee (UK), IRV (The Netherlands), Kompagne BV (The Netherlands), and IGEL GmbH (Germany).

References

Alm, N., Arnott, J. L. and Newell, A. F. (1992) Prediction and conversational momentum in an augmentative communication system. *Comm. ACM* **35**(5): 46–57.

Alm, N., Morrison, A. and Arnott, J. L. (1995*a*) A communication system based on scripts, plans and goals for enabling non-speaking people to conduct telephone conversations. *Proc. IEEE Conf. on Systems, Man & Cybernetics*, pp. 2408–2412. Vancouver, Canada.

Alm, N., Dye, R. and Harper, G. (1995*b*) ALADIN — Advanced Language Device for Interaction. *Proc. ECART 3 Conference*, pp. 150–151. Lisbon, Portugal.

Beattie, W., McKinlay, A., Arnott, J. L. and Gregor, P. (1994) An investigation of the use of recency in word predictive algorithms. *Proc. ISAAC '94 Conference*, pp. 496–498. Maastricht, The Netherlands.

Beukelman, D. and Mirenda, P. (1992) *Augmentative and Alternative Communication: Management of Severe Communication Disorder in Children and Adults*. New York: Paul Brookes Publishing Co.

Baker, B. (1982) Minspeak. *Byte* **7**(9): 186–202.

Broumley, L., Arnott, J. L., Cairns, A. Y. and Newell, A. F. (1990) Talksback: An Application of Artificial Intelligence Techniques to a Communication Prosthesis for non-speaking People. *Proc. European Conference on Artificial Intelligence*, pp. 117–119. Stockholm, Sweden.

Demasco, P. and McCoy, K. (1992) Generating text from compressed input: an intelligent interface for people with severe motor impairments. *Comm. ACM* **35**(5): 68–79.

Don Johnston Incorporated. *Talk:AboutTM*. Don Johnston Inc., 1000 N. Rand Road, Wauconda, IL, USA.

Kraat, A. (1987) Speech Therapist, Morton Roberts Center for Augmentative Communication, Queens College Speech and Hearing Center, Flushing, New York, USA, *Private Communication.*

Levelt, W. J. M. (1994) What can a theory of normal speaking contribute to AAC? *Proc. ISAAC '94 Conference*, pp. 18–20. Maastricht, The Netherlands.

McCoy, K., McKnitt, W., Peischl, D., Pennington, C., Vanderheyden, P. and Demasco, P. (1994) AAC-User Therapist Interactions: Preliminary Linguistic Observations and Implications for Compansion. *Proc. RESNA '94 Conference*, pp. 129–131. Nashville, TN.

McLaughlin, M. and Cody, M. (1982) Awkward silences: Behavioural antecedents and consequences of the conversational lapse. *Human Comm. Res.* **8**: 299–316.

Mandler, J. M. and Parker, R. E. (1976) Memory for descriptive and spatial information in complex pictures. *J. Experimental Psychology: Human Learning and Memory* **2**: 38–48.

Mandler, J. M. (1979) Categorical and schematic organization in memory. In: Puff, C. R. (ed.), *Memory Organisation and Structure*, pp. 259–299. New York: Academic Press.

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press

Mandler, J. M. (1984) *Stories, Scripts and Scenes: Aspects of Schema Theory*. Hillsdale, NJ: Lawrence Erlbaum.

Schank, R. and Abelson, R. (1977) *Scripts, Plans, Goals, and Understanding*. Hillsdale, NJ: Lawrence Erlbaum.

Swiffin, A., Arnott, J. L., Pickering, J. A. and Newell, A. F. (1987) Adaptive and predictive techniques in a communication prosthesis. *Augmentative and Alternative Comm.* **3**(4): 181–191.

Todman, J., Elder, L. and Alm, N. (1995) Evaluation of the content of computer-aided conversations. *Augmentative and Alternative Comm.* **11**(4): 229–234.

Todman, J. and Lewins, E. (1996) Use of an AAC System for Casual Conversation. *Proc. 7th Biennial Conference of the Society for Augmentative and Alternative Communication (ISAAC '96)*, pp. 167–168. Vancouver, Canada.

Tyvand, S. and Demasco, P. (1992) Syntax prediction in word prediction. *Proc. ECART 2 Conference*, Paper 11. Stockholm, Sweden.

Vanderheyden, P. B., Demasco, P. W., McCoy, K. F. and Pennington, C. A. (1996) A preliminary study into schema-based access and organization of re-usable text in AAC. *Proc. RESNA '96 Conference*, pp. 59–61. Salt Lake City, UT.

Waller, A. (1992) Providing Narratives in an Augmentative Communication System. *PhD Thesis*, University of Dundee, Dundee, Scotland.

Woltosz, W. (1988) Stephen Hawking's communication system. *Comm. Outlook* **10**(1): 8–11.

https://doi.org/10.1017/S1351324998001867 Published online by Cambridge University Press