User and Application Simulation in the Evolution of an Intelligent Speech Interface

Presented at the Ergonomics Conference
Brighton, May 1989

Technical Report No. 112

Christine Cheepen
Jill Hewitt
Cliff Hunter
James Monaghan

November 1990

USER AND APPLICATION SIMULATION IN THE EVOLUTION OF AN INTELLIGENT SPEECH INTERFACE

Christine Cheepen, Jill Hewitt, Cliff Hunter, James Monaghan,
Hatfield Polytechnic, College Lane, Hatfield, Herts. AL10 9AB.

1. INTRODUCTION

This paper considers the particular nature of speech interfaces and some of the problems inherent in speech as an interface medium, and it attempts to overcome these problems by improving the interface design. The need for an adaptable interface is identified and the use of models simulating the application and the user is described. Reference is made to a particular project - The Hatfield Polytechnic Intelligent Speech Driven Interface Project (ISDIP) - and the attempts being made to implement such an interface in the context of an intelligent knowledge base.

1.1. Speech applications - current and future

We are at present moving into an era when speech will be considered as an acceptable communications medium between human and machine. Until now, speech has been used only in specialist applications where the hands or eyes are otherwise engaged (eg: speech input for hands-busy parcel-sorting or speech output for eyes-busy aircraft control systems). In the world of disabled users, speech output systems are an invaluable aid for the blind, but speech input has not yet been exploited to the same degree for the manually disabled; such a development will be of inestimable value to those manually disabled, because speech output can exploit the user's recognition skills, which surpass those of the machine. Synthesis is less of a problem than recognition, because human speech recognition goes beyond the mere analysis of the acoustic signal.

With the development of digital telephone networks, there is a growing requirement for human-machine communication over a telephone link, and whereas the usefulness of synthetic speech in this medium has long been recognised (Smith and Goodwin, 1970), the use of speech recognisers at the end of a telephone is the subject of much current research. The British Telecom VODIS (Voice Operated Database Inquiry System) project has shown the feasibility of using a speaker independent recognition system in conjunction with a speech synthesiser to answer train timetable inquiries over a telephone (Cookson, 1988). A scenario of a businessman interrogating a database through his (hands-free) car-phone is a very real possibility.

Currently, there are more applications using synthetic speech output than those using speech input, but a new generation of large-vocabulary speech recognisers may soon make speech input a viable interface medium for many applications.

1.2. Problems in speech interfaces

There are several well-documented factors which contribute to the difficulty of implementing speech interfaces compared with 'conventional' I/O by keyboard, mouse and screen. These problems will need to be overcome before speech is considered favourably in situations where other I/O media are available.

The transitory nature of speech output when there is no visual feedback is an obvious problem which needs to be overcome, but there are other difficulties. Some types of output lack a linguistic representation - for example mathematical and programming symbols and the mediation of graphical concepts. Users can often acclimatise themselves to the unnatural quality of synthetic speech but this may be at the expense of an increased mental load (Cox, 1982).

There are still considerable technical problems to be overcome before recognition systems can realise their full potential. Most of them work on a limited vocabulary size and many are speaker-dependent and require the user to speak in isolated words. Their recognition accuracy is always less than 100% and is affected by environmental noise and changes in the user's voice. In addition, users may have an unnaturally high expectation of the capability of such systems, believing that if they can talk to it, it must understand them.

As the technology improves, some of the problems inherent in speech interfaces will disappear, but we should not assume that a more sophisticated system will be easier to use. For example, a user of a 20,000 word recogniser would not be able to remember the whole vocabulary and would need a strategy for dealing with words that were not recognised. The system could easily appear more confusing to a user than a simple 100 word recogniser, because his expectations would be much higher (Nusbaum, 1986).

1.3 Improvements which can be made by considering human factors issues

A careful consideration of the human factors issues can often considerably improve the usability of a speech interface.

In a study of conversational systems (ones that use both speech input and output), Richards and Underwood (1984) identify a need to match the dialogue structure to the system's capabilities. They found that as polite requests elicited polite (i.e. loquacious) responses, it would be better to make requests more terse as a recogniser could cope better with short responses.

An evaluation of a prototype (isolated word) speech operated word-processor (Hewitt & Furner, 1988) identified three areas for improvement in the design of the user interface which could be implemented without specific user or application knowledge. Feedback to the user could be improved by providing status information on current vocabulary, and offering error handling and recovery in the case of misrecognitions; the cognitive load on the user could be reduced by simplifying the vocabulary structure; and easy amendment of the recognition vocabularies should be allowed, to deal with badly trained words and dynamic requirements to add new words.

These improvements would have the effect of increasing user confidence in the system and consequently user performance, since a person's voice tends to change under stress, causing more misrecognitions

Recent research carried out at The Hatfield Polytechnic has revealed that among the requirements of blind users of speech synthesis systems there are two major features which promote user confidence by allowing him/her to feel in control of the interaction. Firstly, it is essential to allow user control of the output, so that, for example, it can be stopped, speeded up or repeated; the experienced user should be allowed to by-pass prompts and get on with the next command. Secondly, the user should be able to fine-tune the system parameters in order to change pitch and volume.

1.4 Improvements which need user and application knowledge

Some of the desirable improvements are simply not possible without further knowledge of the application and the user.

An essential feature of any speech input system is an undo/redo hierarchy of commands, since this allows easy recovery when a misrecognition occurs. In order to implement these, knowledge of the application is needed, as is a log of user actions to date.

If the system is to be able to predict user input and/or to choose 'intelligently' between two possible meanings of an utterance, it will need specific knowledge of the user (what does this user, or a typical user, normally say in these circumstances?) and knowledge of the application (what makes sense in the current context?) in addition to knowledge of general characteristics of natural language of the requisite type. If generic application and user models are built into the system, a specific user model can evolve, based on the user's current practice. We will return to this in section 4 below.

A further requirement is that the system can automatically adjust to changes in noise level and to changes in the user's voice; for the latter requirement it will need a model of the user's speech production. Use and development of models will be discussed later in sections 3 and 4.

For speech output, an application model would allow the system to assist the user by choosing appropriately sized phrases, and by varying the speed according to the redundancy of the information contained in them. A user model would allow it to adjust according to the particular user's preferences, and the recent use history - frequent requests for repeated information could be met by a reduction in the phrase size or by reduced speed.

In interactive dialogues, the system's choice of vocabulary and style of dialogue can be matched to the user's. Application knowledge will be needed to decide phrase content in a synthesis from concept situation (e.g. database retrieval where the 'bare' fact needs to be put into a phrase to aid its intelligibility).

2. SIMULATION MODELS AND A SOLUTION TO THE PROBLEMS OF SPEECH INTERFACES

2.1. Requirements

The concept of using user-simulation models to improve system performance is not new - it has been incorporated into a number of different projects involving non-speech based office applications. In his discussion of a user interface reference model, Lantz (1987) identifies the need for an intelligent dialogue management system to allow the user interface to be independent of the application and independent of the interaction media. General attributes of human-computer interfaces are summarised as resilience, portability, configurability, intelligence, modularity and responsiveness. It is intended that the proposed system, whilst being developed particularly for speech interfaces, will display these attributes. It should be capable of exploiting the latest technology, not bound to particular devices, and it will be configurable to different applications and different classes of user. In addition, it will be capable of adapting to a particular user and making appropriate decisions for that user. The criterion of responsiveness will be met if necessary by the use of parallel processing architecture.

A further requirement is that the cost of the system should be commensurate with its functionality, i.e. the cost of providing a speech interface should not be excessive when compared to the overall cost of a system without this interface.

General systems theory (Bertalanffy, 1968, etc.) outlines three methods of handling the problems that arise at an interface: buffers (the use of some intermediate store to keep the consuming system supplied), slack resources (having one or both systems idle at times whilst waiting for the other), and formalisation (constraining the types of traffic across the interface in order to increase handling speed). All three have been extensively used in human-computer interfaces. Buffers exist in many forms, perhaps the best being the full-screen display. Slack resources are employed on both sides of the interface, waiting for a response. Formalisation of the traffic across the interface is used so much that the degree is often overlooked.

The reduced cost of processor power, so that the machine is, in personal computer applications at least, the cheaper resource, and the vastly increased speed of serial processing along with the introduction of parallel processing, mean that the cognitive load should now shift across the interface from the human to the machine. This will involve all three areas of interface handling.

In the case of slack resources and buffering, the machine should, as the cheaper resource, supply the slack, keeping the user working productively at all times, by predicting user requirements and having the facilities or results available in advance. The greatest change should however be in the degree of formalisation required in the interface - why should the user have to spend valuable (that is, expensive) time learning exactly what command in this package does some operation common to all packages of the same general type? The user is often supplied with a user manual which should show the interface required by the particular application - that is, the user is instructed how to interact with the machine. Our thesis is that the interface problem should be approached from the opposite side - the computer interface system should be equipped with knowledge equivalent to a 'user manual' showing, inter alia, the interface characteristics preferred by this particular user. The requirement to adapt would then be placed upon the machine rather than the user.

2.2. Simulation models

In order that this preferred interface style could be used across a range of application software, the interface manager would also need simulation models of the various applications. Taking, as an example, a word-processing application, the facility to select a block of text is almost (if not completely) universal, but the commands differ in each case. The user's preferred method would be stored in the user model, the exact commands in the application model.

These models would not be built-in, unchanging parts of the interface manager. For the user modelling system, there would be a generic model of the user, with defaults, available for the simulation of any new user. Certain changes could be made to this model immediately due to the individual's position within the organisation structure. As the user utilises the various packages, the model can evolve so as more accurately to reflect the particular preferences, skills, knowledge and habits of that particular user, thus performing a continuous simulation of user behaviour. For a further discussion of what is meant by evolution of models, and how this evolution might proceed, see section 4 below. A generic application model would also be required. Generic models for each class of application would exist (for example, word processor, spread-sheet, etc.). Each application model would then inherit (by default inheritance) the properties of its generic parent model. However, to propose a purely hierarchical tree structure for the application models is probably a little simplistic as a sophisticated word-processor package might provide some of the facilities more usually associated with a desk-top publishing system, for example. The resultant structure will be a lattice of models. This will require a sophisticated inheritance system, selecting the appropriate characteristics.

To cope with the special problems of a speech driven interface, it is necessary to rely heavily on the simulation of user processes, and three further models are required - a language model, a dialogue model and a speech model - all of which are used as the default starting point for the evolution of the model of the specific user. An application task model is also required as this provides the immediate context for any interaction between the user and the system. As the diagram in section 4 shows, these components are modelled separately in order to improve the modularity of construction of the system.

3. TOWARDS A SOLUTION - MODELLING FROM THE HUMAN/HUMAN INTERFACE

The two most obvious reasons to start with the human-human user model in designing any interface to a computer-based system are, firstly, that successful simulation of human interaction will make user efficiency much higher in the final product, and, secondly, that it is always worth looking at speech, the most successful natural language interface in existence, in order to get some ideas of how an artificial system might be designed. Already, great strides have been made in areas such as signal processing and the development of the hardware supporting it. There have also been important theoretical strides in our hearer models in speech perception and it is to this that we now turn.

Human speech perception, like human vision and other perceptual skills, does not work linearly in the way classic computational models do. Human beings endeavour to impose a *Gestalt*, an holistic framework, on the input presented to the senses. This has been demonstrated in experiments in such areas as chess playing and facial recognition. Because human beings process sensory input at all levels according to predefined schemata derived from experience, some of the most interesting results derive from presenting subjects with highly degraded or deliberately ambiguous input data. The resulting false perceptions are usually elaborate 'reasonable' constructions with often scant regard for the data.

3.1 The process of speech perception

Speech is not in fact a sequence of discrete sounds, but rather a continuously varying stream of air pressure variations. Any segmentation has to be imposed on the stream of speech by the hearer. Although humans can only reliably identify individual units of their sound system at the rate of 7-9 per second, speech is transmitted and decoded by normal users at somewhere between 20-30 segments per second. The effect which takes place at this speed has been compared to the way we perceive the sequence of cinematograph frames as a moving picture. In fact, there is more to it than that. Whatever the average speed of transmission, the speaker will, among other things, slow down at particularly significant points and will speed up the number of syllables per unit time where the information being provided is designed to provide background or context for the main message.

The 'units of speech' we have been referring to rather coyly above have often been taken to be items called phonemes, surprisingly similar to the familiar letters of the roman alphabet. In fact, from the point of view of perception, the syllable is a much more important phonetic unit, containing as it does many more perception cues than the smaller stretches usually examined by recognition systems. A close analysis of the articulation times of various sounds shows that the various articulatory motions are initiated in an interleaved manner in order to have them arrive in the right order. As a consequence, the hearer interprets often very small gestures of the articulators in the directions of the positions they would have to take up in order to pronounce the sound fully. How does he achieve this feat?

This is a daunting task. However, the human processor employs expectation-driven techniques to match hints of incoming patterns with what is expected. In addition, he operates in a massively parallel fashion, matching patterns at the phonetic, morphological, lexical, grammatical, semantic, pragmatic and discourse levels.

The continuous temporal speech pattern is 'unpacked' by the hearer and matched against an internal representation of the word-forming syllables of the language in question. The main problem here is the discouraging lack of **acoustic invariance** in the signal. The reader might like to try saying the English words *try* and *rye* one after another and note the wide differences in the qualities of the two sounds represented as *r*. There are lots of similar cases of quite gross differences in the actual signal of sounds that count as the same.

In a classic article on speech perception, Ladefoged and Broadbent (1957) demonstrated that hearers, at the start of the speech input from a particular source, use certain sounds to create a model of the vowel space of the speaker, and Gerstman

(1967) has shown that a computer program can be written to derive a speaker's vowel space from the formants of the vowels [i] or [u].

3.2. Discourse patterns and situational constraints

The previous section has discussed the ways in which the human processor employs minimal clues from the speech input and matches these against his/her internal representations of the syllabic patterns of the language, in order to construct a hypothesis about the incoming signal, which is then confirmed when the rest of the signal is received.

The same kind of procedure operates at all levels of language from the lowest to the highest. For reasons of space we will concentrate here on only one more level - the discoursal level. As with the other levels, the human processor has internal representations of the various discourse structures of the language, and these are used together with input clues to aid comprehension of the discourse as it develops and to enable the hearer to hypothesise about where the discourse will go next.

Any language has a multiplicity of discourse patterns, ranging from very generalised ones, such as *question/answer*, and *problem/solution* (Winter, 1982), which are likely to occur in many kinds of discourse, through to very specialised ones for use in particular kinds of discourse; the most specialised of all tend to be ritualised, as in those governed by the law, where the form and order of the discourse elements are predetermined, and must be adhered to for the discourse to be legally effective. The number of discourse patterns which are not so strictly predetermined is, of course, very large, and for a hearer to sort through all possible patterns to find the correct one to match with an incoming clue is an enormously difficult task, given that the hearer is required to (and nearly always does) understand what he hears instantly, while at the same time (when the discourse is in dialogue form) preparing his own contribution, which must be relevant to what has gone before. So how is this very difficult task performed?

The human hearer employs a wide variety of skills and different kinds of knowledge when processing and comprehending incoming speech, and is, for a large proportion of the time, expecting what comes next; that is, much of what he hears is already, in some way, understood, because, in all but the finest detail, he knows what the speaker is going to say next. This is done by the application of what can be thought of as "situational" knowledge (Monaghan, 1979). This means that certain speech situations specifically call for, or at least set up, the strong likelihood of certain kinds of language manifestations at all levels, - lexical, grammatical, semantic, pragmatic and discoursal, all of which serve to provide extra clues for the perception of the incoming phonetic patterns.

That language both represents and is bound by the situation is clearly observable from consideration of any speech situation, - the kind of discourse, vocabulary etc. which is likely in a doctor/patient consultation is markedly different from that which is likely between friends shopping together. So clearly defined are the many different kinds of situation and their corresponding discourse patterns that the use of the 'wrong' kind of discourse in any real speech situation will be noted (and quite probably commented on) as odd, deviant, strange, or a sign of mental stress or illness in the perpetrator, while such misuse for entertainment purposes is recognised as humour, and produces amusement in the hearers.

Discourse patterns, in addition to being situationally bound, are, to a large extent, hierarchical, so that while large, rather generalised patterns are used to organise what is said at the highest level, lower level, more precise patterns occur within the larger structures. In the office context, for example, the dictation situation has an overall discourse pattern something like: **introduction**, which will make some kind of specific reference to the task in hand, and, therefore, the particular kind of discourse which will follow, e.g. 'I've got some letters for you'; followed by **letter**, or **memo**, or both, repeated an indefinite number of times; followed by **conclusion**, to indicate that this particular discourse is at an end, - e.g. something like 'That's the lot for now'. Below this general pattern there are other patterns for each element.

A business letter can, of course, fall into a number of different discourse patterns, but the range of possible patterns once the hearer has reached this level is by no means too large to scan in an attempt to understand and process the incoming speech. As dictation of a particular letter progresses, more precise and detailed levels of the discourse emerge, usually signalled in advance by special items of vocabulary which function to alert the hearer (and the ultimate reader of the letter) to the way in which what follows should be interpreted: the word "Unfortunately" at the beginning of a paragraph is a clear indication that what follows is bad news of some kind, the word "Finally" unambiguously signals the last section of the message, the word "but" tells the hearer (and reader) that what follows contrasts with what has gone before.

The process of discourse comprehension is not, then, for a human hearer, a matter of frantic sorting through an almost infinite number of possible patterns for the correct one to match with incoming speech clues. At the outset of any speech situation (indeed, any language situation, as these comments apply equally well to written discourse), the hearer will already have narrowed down the choices to what is possible in that situation, and each piece of spoken information received will narrow that range of choices still further, through what is probable, to what is safely predictable (Cheepen, 1988).

This hierarchical aspect of discourse is important in two ways to the design of the user interface and user application models. Firstly, in terms of the proposed speech system, the user can be viewed largely as the sum of his potential discourse patterns, so a simulation of those patterns serves as a simulation of the user. Secondly, and perhaps more crucially for the implementation of the system, the progressive narrowing of choice of possible 'next happenings', which is an integral part of this hierarchical organisation, will simplify dialogue design and will speed up machine response time considerably, particularly in cases where there is a requirement for the machine to anticipate the needs of the user, or to initiate some suitable next step.

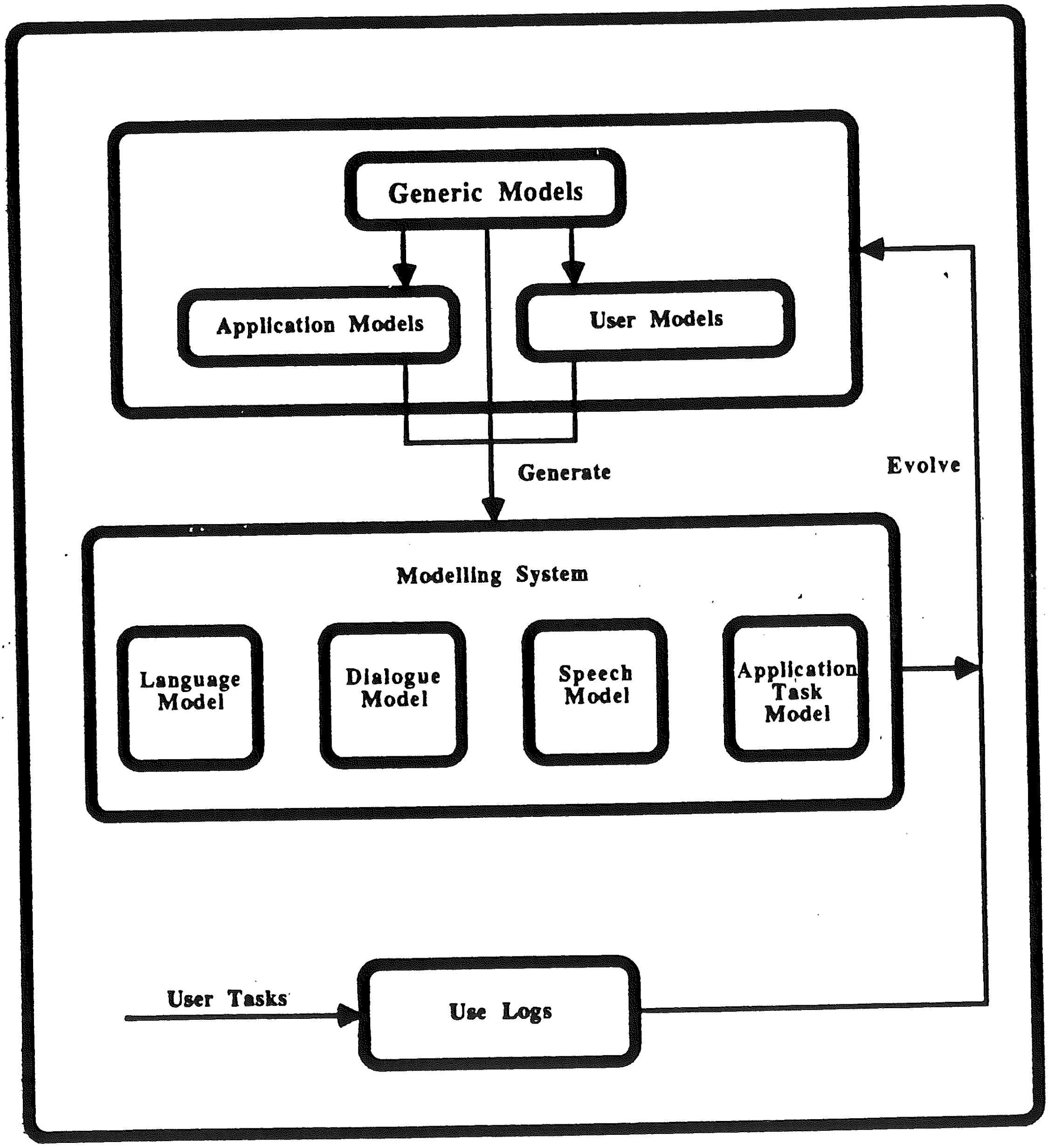
3.3. Summary

As sections 3.1 and 3.2 above have indicated, the comprehension and production of natural spoken language is a complex phenomenon. Human beings, however, are able to manage it with such ease that they are able to listen to and comprehend language input while simultaneously preparing a suitable output response, and, although mishearings and misunderstandings are frequent in human/human communication, the repair and recovery rate is extremely fast. To be wholly acceptable in a real working environment, any automated system incorporating speech input and output must be capable of a similarly high performance; our view is that this can only be achieved by basing the system on models of speech related behaviour in human/human communication, thus ensuring that the eventual interaction between the system and the user is a true simulation of the interaction between human communicators.

With such a simulation in mind, part of our ongoing work is a dialogue analysis of the interaction which takes place in a sample of real, working offices. This is being carried out with a view to producing a dialogue knowledge base which will be fed into the Interface Management System (see section 4 below).

4. PROPOSED SOLUTION

Proposed Interface Management System Cycle



A new task or new user will immediately be assigned the generic task or user model as the base for simulation. As the simulation model is used, the original default settings can be specialised to match more closely the task or user. If previous models are retained, over-specialisation can be corrected by reverting to the earlier model, at least for that aspect.

Evolution of simulation models will be based upon the use-logs kept of each session. The searches of the use-logs to find patterns which are useful will be very time consuming, but this is not a problem as the necessary processing can be performed during idle time which, for most desk-top machines, is substantial. Such idle time processing could, perhaps, be compared with dreaming in humans.

Not only will this evolution, in time, ensure that the interface is presented in the desired way, but also that the degree of verbosity of requests and answers is suited to the skill and knowledge levels of the user. Further, given patterns of behaviour it will be possible to predict user activities at two levels: firstly, during a session the next command can be forecast and pre-executed (with suitable undo routes), and, secondly, the likely request in the next session can be forecast so that, for example, the usual Monday morning search of incoming mail could be pre-executed and waiting for the user. In neither case need the user be aware of this except in terms of vastly improved reaction time. It would also be possible to suggest, implement and make available, macro-commands based on previous usage.

The evolutionary adjustment of the models of application and user will apply at all levels of abstraction, so that the generic models will themselves evolve when possible. Such generic models should only change in accordance with the use logs of a reasonably large sample of users. One method of obtaining such a sample of users would be to allow a network of machines to compare their user models so as to generalise from them.

5. PROTOTYPE AND FURTHER DEVELOPMENTS

Throughout the Hatfield Polytechnic project we intend to have a working prototype at all times. At present we are approaching the task from the directions of speech input and output simultaneously, and we have a speech driven word processor, and a talking keyboard and diary. As work progresses, these will be combined, and the user and application models will also be incorporated into the system, thus ensuring constant updating and improvement of the prototype, which is re-evaluated and redesigned at each stage as the sub-systems are developed.

REFERENCES

Bertalanffy L. von, 1968, General Systems Theory, Braziller.

Cheepen C., 1988, The Predictability of Informal Conversation, Pinter Publishers Ltd.

Cookson S., 1988, Final Evaluation of VODIS, Proceedings of Speech '88 7th FASE Symposium.

Cox A.C., 1982, Human Factors Investigations into interactions with machines by voice, Proceedings of the International Conference of Man-Machine Systems 1982, IEE publication 212.

Gerstman L., 1967, Classification of self-normalised vowels, Proceedings of the IEEE Conference on Speech Communication and Processing.

Hewitt J. & Furner S., 1988, The Design of a User Interface Management System which provides Speech Input for Text Processing, Proceedings of Speech '88 7th FASE Symposium.

Ladefoged P & D. Broadbent, 1957, Information conveyed by vowels, Journal of the Acoustical Society of America, 29

Lantz K.A., 1987, On User Interface Reference Models, SIG HCI Bulletin Vol 18 No.2.

Monaghan J., 1979, The Neo-Firthian Tradition and its Contribution to General Linguistics, Niemeyer, Tubingen

Nusbaum H.C., 1986, Human Factors Considerations in the Design of Large Vocabulary Speech Recognition Devices, Speech Tech '86

Richards M.A. & Underwood K.M., 1984, How should People and Computers Speak to each other?, Interact '84

Smith S.L. & Goodwin N.C., 1970, Computer-Generated Speech and Man-Computer Interaction, Human Factors 12 (2)

Winter E.O., 1982, Towards a Contextual Grammar of English, George Allen & Unwin Ltd.