People interact through computers not with them

T.J.M. Bench-Capon and A.M. McEnery

A view of human–human interaction is presented and its implications for human–computer interaction discussed. Two propositions are advanced. The first is that interaction with computer systems is properly seen as mediated discourse, more akin to reading books and notices than interpersonal dialogues. The second proposition is a consequence of this, namely that unconstrained natural language is unlikely to provide an ideal form of interface.

Keywords: natural language, human–computer interaction, user models, pragmatics

An illocutionary act (Searle, 1969), such as the making of a statement, an offer, a promise or the like, is a basic human act of communication, and necessarily requires a speaker and a hearer. The speaker and hearer may not be individuals, or even be distinguishable, but objects for the two roles should be readily discernible. In natural language, the indefinite nature of the illocutionary act and the indeterminacy of realising the force of that act, i.e. the hearer actually understanding what the speaker actually wanted to say, is a phenomenon readily observable in the form of misunderstandings.

Thus it is clear that illocutionary acts are not straightforward, for the utterances used to convey meanings do not relate to those meanings in a straightforward way. A given utterance may be used to convey many meanings, and many utterances may be used to convey a given meaning. Thus the speaker must select an utterance from the many available, and the hearer must select a meaning from the many that could be attached to that utterance. The surface sense of an utterance from the speaker may not be actually what the speaker wanted to convey. Thus there is a distinction between what appears to be the meaning, and what the true force of the utterance is. This distinction is important and will be returned to later.

In order to guide the available choices, the situation of the discourse must be used to supply clues; indexicals in the utterance, such as pronouns and demon-

Department of Computer Science, University of Liverpool, PO Box 147, Liverpool, UK. Tel: (051) 794 3697. Email: sq35@uk.ac.liverpool.ibm

0953-5438/89/010031-08 $03.00

stratives, will often be disambiguated by the situation. But there are other clues, such as tone of voice and body language. These nonlinguistic channels of communication may be of varying importance. Some, such as tone of voice, are closely associated with language, in the respect that like language they are largely the result of social conditioning; thus the meaning associated with any given tone of voice is arbitrary (e.g. Sausurre, 1959). There is, for example, nothing inherently 'rude' about the physical characteristics of the tone of voice that people call a 'sneering' one: it is only because our society has decided to append this meaning to this tone of voice that it is conceived as so. This arbitrariness of the symbol leads to factors such as these being of lesser importance, for although they are recognizable their meanings, because arbitrary, are also relatively mutable and unreliable. More important nonlinguistic channels of communication are those that are genetically inherited. These channels gain importance because they are not arbitrary, as their meanings are genetically encoded. An example of this is the three basic cries a baby is born with, and which have been found by empirical study to have nonarbitrary meanings associated with them (Wolff, 1969). Yet in the pragmatic disambiguation paradigm, perhaps the most important disambiguating elements of all are the model of the hearer possessed by the speaker, and the model of the speaker possessed by the hearer. Consider the different ways in which you would express yourself to a child, a foreigner, an intelligent layman, and an expert in your own field. Consider also the way you listen to each of these.

It is worth, at this juncture, considering an example of how force and sense may not necessarily map onto one another. A good example is given by Wittgenstein:

> 'Make the following experiment: say "It is cold in here" and mean "It is warm in here". Can you do it? – And what are you doing as you do it? And is there only one way of doing it?' (Wittgenstein, 1953)

The example from Wittgenstein is good because it shows perfectly how sense and force may be disjoint, and how it is the user models, where secondary sources of data are absent, that are important in providing the force of the utterance. To elucidate, knowledge of the user model can force illocutions with an apparently similar, indeed identical, sense to have a different force. If we take the definition of sense to mean that which is governed by formal rules (Leech, 1983), i.e. the componential semantics of the illocution, while the force is the meaning understood when nonformal information seeking elements are brought to bear, e.g. pragmatics (Levinson, 1983), then it is easier to see that a user model of the speaker held by the hearer can fundamentally change the force of an illocution from the speaker. Returning to Wittgenstein's example, if we knew any speaker was standing in a cold room, then the sense and force of Wittgenstein's illocution would be identical, because we would know that the speaker was in a situation in which the illocution was supported by the environment. In contrast, if the room were noticably too hot, sense and force would be opposite, the speaker intending an irony readily appreciated by the hearer. Now, however, consider an Eskimo speaking to an Amazonian Indian.

If they were to stand in a room which was cold to the Indian, it would not necessarily be cold to the Eskimo. Thus if the Indian were to say, in all honesty, 'It is cold in here' unless the Eskimo has a good model of the speaker, pragmatic failure will ensue, because the Eskimo will presume that the Indian is using irony, and actually stating Wittgenstein's inverse, i.e. it is warm in here. Thus user models are very important, as they can be the key to success or failure in realising the force of a given illocution.

The terms 'speaker' and 'hearer' do not, as the nomenclature may suggest, presuppose the channel of communication. The terms are merely used to represent the two members of any basic illocutionary act, irrespective of the medium. This is important to note, because the above observations apply with equal force to both written language and spoken language. Yet in written language the problems of realising the force of an utterance are more vexed as many of the situational clues are absent, and so the models assume greater importance.

Two examples may be used here. First it is the case that Shakespeare was in his day a popular writer, appreciated by all levels of society, whereas today he is seen as an esoteric taste for an educated few. This is because the model of his hearers, the general public of the late 16th century, is inappropriate for the general public of today. And the model of author is not easily formed by a contemporary audience. The education necessary to appreciate Shakespeare enables the hearer both to have a better model of the author, that he believed in the 'Divine Right of Kings', and held typical Elizabethan views on cosmology, etc. and to think himself into the position of a person with the outlook of a contemporary of Shakespeare's, so that the hearer model used by Shakespeare can become once more relevant.

At a less exalted level, consider public notices like 'No Smoking' signs, and those giving information and instruction at stations and airports. The influence of the model of potential readers on the authors can be seen in the amount of detail they give, and even in the language used. Thus notices at airports tend to be in several languages, anticipating a polyglot readership, whereas those at local stations tend to be in the local language only.

The above is meant mainly to express received wisdom, and is broadly in line with, for example, Diaper (1988) and Winograd (1983). Now we must go on to consider the implications for human–computer interaction. If we consider the conventional computer system we find strong similarities with any other form of written illocution. The program will contain a number of messages to the user displayed as text on a VDU screen, and these both appear as, and function as, the more usual kinds of direction-giving notice. Thus simple information seeking, or action guiding illocutionary acts, e.g. 'How many?' or 'Press CR to continue' correspond to signs like 'Please state destination' and 'Tender exact fare' found on 'buses. When the programmer selects the text to use in those messages, the user will be being modelled, and assumptions made as to what the user will require instruction in and what the user can assume. But it is the programmer, not the system, which is doing the modelling. Thus an Ada program may contain the message 'Enter three real numbers:'. The use of such a message suggests that the programmer believes the program's user to be aware

of the following things: that a real number in Ada needs to be entered in the form '2.0' since '2' will give a failure; that the numbers can be delimited by spaces; and that input will only be sent to the program on pressing the return key. All three items of knowledge must be possessed by the user in order to obey the instruction successfully.

Had the programmer not made the assumption of this kind of familiarity with the underlying language, more detail would have had to be offered as to the precise form of the number, the delimiters to use, and the need to terminate input with a carriage return. One key to successful messages on such systems is given by the programmer's understanding of his potential audience. But no less important is the model of the programmer formed by the user of the system. Thus it is important in the above example that the user be aware that the programmer thinks the user will be conscious of the required format of real numbers; otherwise the user could reasonably have expected to be told about the need for the decimal point, and would not supply one unless explicitly told to do so. Whereas, importantly, if he were modelling the system rather than the programmer, he might expect a first mistake to be taken into account in future messages from the system.

In essence, the author, in any form of mediated illocution, must have, to put it in layman's terms, a 'sense of the reader' (and the reader a 'sense of the author'). Part of this is that assumptions are made about the audience; thus the models, and therefore the style of the illocutions, change with reference to the perceived audience: a program for the noncomputer literate will have a different set of error messages than a system for use by computer scientists (one would hope). However, before this point is made fully, at least two possible objections have to be addressed. First, there is a growing field of research into producing adaptive models of the user to provide graceful adaptation at the interface. Such an interface would clearly produce a wider series of problems for this hypothesis than those already presented. However, as the form of such models are not yet certain by any means, any discussion of them here would be somewhat speculative, as no model exists to provide a tangible basis. To quote:

> 'The development of user profiles is an area of continuing research and as yet no formalization exists'. (Coats and Vlaeminke, 1987).

A second point, which could be made here, is that attributing the target group user model to the programmer may be difficult, because errors in the software may mean that the software behaves as if it had a model different from that which the programmer intended. It would, however, be a mistake to say that the software is using its own rather than its programmer's model. All that is happening is that there is a faulty implementation of the programmer's model.

Another example will serve to show that a hearer's model of the speaker is also used in human–computer interaction, and again the model ought to be the user's model of the programmer. When presented with the not uncommon message 'type "Q" to quit and "X" to exit', the user will have to rely on a model of computer programmers and the terms they use to aware of the fact that 'X' will, and 'Q' will not, leave the file and save any changes.

The point being stressed here is that in designing, and using, the interface of a conventional computer system a good deal of modelling is going on, of precisely the same sort that is involved in the writing and reading of public notices. But note that it is the programmer who should be modelling potential users, and the programmer that the users should be modelling; the computer system is properly viewed only as the passive medium of communication, like the board on which a notice is painted. The programmer is not intending the computer system to model its users, and the user should not be trying to form a model of the system, in order to understand its messages, although, of course, a model of the system will be invaluable in understanding its behaviour. Just as a notice is intended to mean what the writer of the notice intended it to mean, so too the computer message is intended to have the meaning intended by the programmer. The system does not have any intentions, and if the user attempts to define them the user is committing the pathetic fallacy of anthropomorphising the system, and such a mistake can only produce error rather than insight. An example of where error may creep in can be seen by considering what happens when a discourse failure occurs. In human–human interaction there are a variety of ways in which such a breakdown can be repaired, which try both to save face for both parties, and to effect a revision of the force of previous utterances so that what has already been said can be of some use. Such tactics applied to a computer system are likely, in contrast, only to dig a deeper hole. Seeing the system for what it is, on the other hand, encourages the more appropriate strategy of simply aborting the interaction so as to return to a point of understanding and start again. Of course considerations such as saving face are entirely irrelevant.

The situation is not much changed when the system incorporates a range of facilities intended to cater for users with a variety of skill levels. Such a system admits (usually realistically) the inability of the programmer to devise a means of communication that the programmer believes will satisfy the entire gamut of potential users. Therefore the programmer is offered the opportunity to characterise the user in a limited number of ways, and devise a series of interfaces believed appropriate to the various classes of users. The system will incorporate a flag which corresponds to the user's choice, and will offer the interface appropriate to the setting of the flag. It is, however, quite unnecessary to say that the system has a model of the user and is tailoring its communications in accordance with that model, instead of saying that the programmer has a variety of competing models, and has designed the program to accommodate this.

Little of this can be disputed with regard to conventional systems, but we need to go on to consider AI systems, expert systems, and knowledge-based systems. Here, the user is often led to ascribe intelligence to the system, by the way in which the system is presented. And because communication and the appropriate selection of utterance in communication is a vital part of intelligence, the user may be tempted to try to ignore the programmer, and to believe that the system is communicating with the user, and so start to model the system, and, more significantly, expect the system to model the user, rather than confining these activities to the programmer. Such an attitude can only lead to error, in that the users will be giving clues in their utterances, assuming that

this will help the system in its modelling, and may even form later utterances in the belief that earlier clues have been taken on board. The true perspective is that the modelling underlying the system's messages has been already completed, and completed by the programmer, so that these clues are, in fact, useless. Thus the user is proceeding on a false premise, with the attendant dangers both of misunderstanding output, and of having false expectations as to the force of those utterances received by the system.

These real dangers are exacerbated by the extent to which the interaction is made to seem natural, a good example being proactive interference in the computer language ENGLISH (Schneiderman, 1980). The more the phrases presented by the computer are adapted to the specific situation, the more the user may be tempted to think them adapted to each user, and so read into them meanings that would have been recognised as inappropriate in a book or conventional system. For this reason it might well be better for the designer of the system to emphasise to the user that canned text or text crudely adapted to the user, is what is being offered, rather than trying to give the impression of output spontaneously generated and selected by the system. In truth, the correct model for most of the consultative expert systems that are seen nowadays is that of the handbook, in which the text put out by the system is seen as the text of the handbook, and the operation of the system as a (more or less) sophisticated routing through the handbook. With this model in mind, the user will understand the output better, and give responses more likely to have been anticipated by the author of the system, thus resulting in a generally more satisfactory interaction all round. And in this model, the users see themselves as communicating not with a computer system, but with another person, mediated through the use of a computer system.

But what of systems which purport to have a 'natural language interface?' By this one should understand that the user can offer to the system input expressed in whatever way preferred, and will receive output akin to that one would expect from an intelligent speaker of one's language. For such a system to function there would indeed need to be much more of the sort of modelling that we find in genuine interpersonal communication. For the user, now that there are no constraints provided by definite syntactic and semantic rules governing input, will have to make choices as to the form in which the input is offered. And the system too, must have a model of the user if the selection of the form output is to be genuine and natural, rather than the kind of predetermined fare found in current systems. Such might be an ideal sort of interaction, and would be arguably artificial intelligence worthy of the name, but it goes far beyond what is currently on offer. The best use contemporary systems can make of varied forms of input is to use its reflection in the form of output. Which means that the form input and output has no real significance beyond window dressing, whereas in genuine communication between intelligent individuals the form of expression will make genuine advances of mutual understanding. Again we must be wary of taking the illusion for the reality and using the intelligence of the user to foist significance onto utterances.

The purpose of this discussion is to suggest two things. First, that interpersonal illocutions are not a very fruitful model of human–computer interaction, so

that we should look to mediated illocutionary situations, books, notices and the like for our models, rather than interpersonal dialogues (even telephone dialogues which have become popular models of human–computer interaction of late, as in, for example, Kidd (1985)). Such a shift of attention would, we believe, yield substantial insights. The second thing follows largely from the first, and it is that even if we were to accept that full natural language is the ideal means of communicating with computer systems, we should not expect gains by approximating the superficial characteristics of such communication. Natural language is a rich method of communication not because of the choice it allows in the forms of expression, but rather by what it allows to be conveyed through the choices that are made. So unless these distinctions can be recognised and used, the surface appearance of varied forms of expression is going to be at best an inconvenience and at worst a positive hindrance. In many situations humans have developed formal languages for their interactions; thus cricket umpires have a set of universally understood signals to express their decisions. No natural language could compete with this formal language in terms of lack of ambiguity and instant and universal understanding. In computer systems we generally have situations where what needs to be communicated is of a similarly restricted nature, and there is no reason why we should look beyond a limited and formal language for such systems. Unconstrained natural language is not always best as a method of communication even in human–human interaction. Humans tend to specify a context-dependent sublanguage, a code as linguists call it, to meet the needs of a task. Thus even studies of natural language would suggest that the development of suitable codes for human–computer interaction would not only be sensible, but also what the use of 'natural language' would lead one to suspect would occur anyway, even if unconstrained language was allowed (Diaper (1988) reaches a similar conclusion, p40).

Communication requires the language used to be fitted to its purpose. For many human purposes the richness and nuances of natural language are needed. This is not the case for existing computer systems, and natural language has disadvantages too, and so is best avoided for such systems. Of course, were we to develop a computer system we wished to converse with as an equal, we would want the richness. But let us develop such a system before chasing natural language as a standard tool for such an interaction.

As a final point, it should be noted that unconstrained natural language may be a worthwhile computational goal in itself, and indeed both authors believe that it is, but such a system would have purposes quite distinct from effecting an improvement in the interfaces to computer systems.

References

Coats, R.B. and Vlaeminke, I. (1987) *Man computer interfaces, an introduction to software design and implementation* Blackwell Scientific Publications, Oxford, UK, 35

Diaper, D. (1988) 'Natural language communications with computers: theory, needs and practice' in **Duffin, P. (ed)** *KBS in Government '88* Blenheim Online, Pinner, UK, 21

Kidd, A. (1985) 'The consultative role of an expert system' in **Johnson, P. and Cook, S. (eds)** *People and computers: designing the interface* Cambridge University Press, UK

Leech, G.N. (1983) *Principles Of Pragmatics* Cambridge University Press, UK, 24–30
Levinson, S.C. (1983) *Pragmatics* Longmans, London, UK
de Saussure, F. (1959) *Course in general linguistics* Philosophical Library, New York, USA (translated by Wade Baskin from the original French edition, 1916)
Schneiderman, B. (1980) *Software psychology*, Winthrop Publishers Inc., Cambridge, MA, USA, 199–200
Searle, J.R. (1969) *Speech acts: an essay in the philosophy of language* Cambridge University Press, UK
Winograd, T. (1983) *Language as a cognitive process: syntax* Addison-Wesley, Reading, MA, USA
Wittgenstein, L. (1953) *Philosophical investigations* Blackwell, Oxford, UK, 140
Wolff, P.H. (1969) 'The natural history of crying and other vocalisations in early infancy', in **Foss, B.M. (ed)** *Determinants of human behaviour IV* Methuen, London, UK