EXPERT SYSTEMS AND NATURAL LANGUAGE ANALYSIS

Yorick Wilks

Rapporteur: Mr. J.G.B. Heal

Abstract

Some of the most successful computer dialogue systems have virtually no knowledge structures of any sort, but they have a very simple criterion of success - persuading the human dialogue partner that they are human-like. Other systems are able to perform useful advice tasks by "understanding" the stage reached in the task of, say, repairing a leaking tap, by the role of that stage in the overall task expressed by the whole frame. The key notion is that of frame - a rich knowledge structure available to the computer understanding system.

Human dialogue understanding is also closely bound up with the possession of a model of the speaker (his goals, beliefs, plans, intentions, etc.), where these, of course, may differ from those of the hearer/understander. Some work recently done at SRI in California and at Toronto is described, that attempt, in a suite of computer dialogue systems, to model these features. Ways are also suggested in which their systems are implausible as models of beliefs.

Introduction

Problems of natural language analysis can complicate any definition of what it is to be an expert system, and so affect the choice of which existing computational systems to describe in a paper such as this.

It is clear that a natural language component is not essential to any expert system. Any such system could, in principle, be questioned and give advice in a formal language; one that could approach a natural language as closely as was thought desirable. Considering an expert system as natural language independent, just for a moment, I shall also assume that, to be an expert, a system must have more structure than that of a fact retrieving question-answer system. If we do not observe this restriction, then the whole universe of question-answer systems, and data base interrogation systems (normally with natural language front-ends) come into the picture.

Where consideration of natural language makes the definition of an expert system more complicated is when the expertise is, in whole or part, expertise in natural language communication itself. I shall assume further, in this paper, that expertise about natural language performance is not a sufficient condition for consideration here, but it is a necessary one. Were this latter condition not observed, this paper would have no raison d'etre, for we could discuss only natural-language-free expert systems.

This point may be sharpened if we look very briefly at typical output from Colby's PARRY system (Colby 1975).

I: what trouble did you have with the police?

P: COPS DONT DO THEIR JOB

I: what do they do instead?

P: COPS ARREST THE WRONG PEOPLE.

I: They arrest the right ones sometimes,

P: THE LAW SHOULD GET THOSE ITALIAN CROOKS.

I: Are you Italian?

P: MY BACKGROUND IS BRITISH, BUT I WAS BORN IN THIS COUNTRY.

There is an interesting paradox about this system, designed by Colby to model a paranoid patient in a mental hospital: it is almsot certainly the most robust general dialogue system in existence; the only one that could even be considered as a candidate for Turing's famous indistinguishability test for machine intelligence. However, the prestige of this system is low at the moment among research workers in artificial intelligence (AI for short) and natural langauge, and many do not consider it AI work at all. Why is this? Because, they say, it knows nothing : there is nothing on which it is an expert except a recitation of "facts" about its own imagined personal history. It appears to understand language, they continue, only by covering up its continual misunderstanding with an ad hoc "bag of tricks".

Let me now list a number of key features of the PARRY system:

(1) It goes for the underlying "meaning structure" of the input rather than the surface syntax. If you type to it "would you be so kind as to give me a hamburger", it would consider only the last four words when constructing a reply.

(2) PARRY has "conversational/communicative strategies" for winning in conversations with the human questioner, and winning for PARRY is telling you what he wants you to hear (as in the introduction of Italian mafiosi in the quotation above).

(3) PARRY has an elementary model of his interlocutor, which changes with the values of FEAR and ANGER variables.

(4) PARRY can refer back in the dialogue : for instance, to tell you you have said the same thing before and to ask you why you are repeating yourself.

(5) Words and sentences are interpreted not context-free, but dependent on where one is in the dialogue. This feature was called a "script" in the early, more primitive, system ELIZA by Weizenbaum (1967).

I list these features first because these happen to be, and this has been largely unnoticed, the dominant themes of recent research on natural language analysis within the AI paradigm. Many of those researching on the above themes now do not realize their provenance. And, as I hinted earlier, I am going to concentrate in the body of this paper on systems that embody, not only non-linguistic expertise, but interest from the point of view of research on natural language itself. Hence the themes above will recur in what follows.

I brought PARRY into this discussion, secondarily, because it sharpens the earlier point on expertise : PARRY is an expert on nothing objective and non-linguistic (i.e. only on his own history), but he most certainly is a linguistic-communication expert, in the sense of a performer, though not of course a reflective one. He cannot tell us about linguistic communication.

In what follows I shall concentrate on two expert systems with serious language-analysis interest, built by Grosz et al. and Perrault et al. respectively. Others, for whom there is no space for detailed treatment, will be mentioned at the end. In these systems, the level of expertise is low and commonsensical, albeit complex and of commercial importance. In that sense, they do not conform to Feigenbaum's very high level demand that expert systems should be of Ph.D, or a physician's, competence, as is his own MYCIN (Shortliffe 1976). There simply are no expert systems at that level that have any interesting language aspect whatever.

The SRI assembly system

The following is a typical pair of dialogue fragments from Grosz (1979).

SI: The lid is attached to the container with four ½" bolts.

RI: Where are the bolts?

S2: Attach the lid to the container.

R2: Where are the bolts?

The task setting here is an expert system, giving the responses above, that can advise on the construction of bolted structure from parts (and, of course, can actually carry out the task if coupled to a suitable robot). The language of understanding expert module we are concerned with is one that can give and receive advice in natural language about that task, above all because it has a complex representation of the task and the components required for it.

This work fits clearly within the so called frame-theory tradition in AI, a term given to its present meaning by Minsky (1975), and which denotes any system that is sufficiently complex to express the stereotypical sequence of sub-activities that constitute a higher-level activity. Typical examples of frame-systems were formal expression of the structure of sub-actions constituting shopping in a supermarket, or eating in a restaurant.

The important idea behind the word frame is that of "top down understanding" of language, situations and tasks : that is to say, that we must be able to access unmentioned items and sub-activities in order to understand. Grosz refers to the phrase <u>the bolts</u> as being in <u>explicit focus</u> in response RI and in <u>implicit focus</u> in R2.

It seems quite clear that in both responses the phrase <u>the bolts</u> is the focus, or the centre of attention or discussion at that point in the dialogue. The key difference is that in R2 it has been drawn, perfectly appropriately, from knowledge structure of the sort Minsky called a frame. It is just this ability to bring in unmentioned, or implicit, items that constitutes so much of intelligent dialogue.

The formalism in use at SRI for expresseing such structural knowledge is a sophisticated version of semantic net formalism due to Hendrix (1975). Semantic networks have traditionally been a static representation for knowledge consisting of nodes (representing entities, or classes of entities) and arcs (representing the relationship between the entities).

Thus:

The lower nodes represent a single dollar $1 and a specific set of bolts (B1) respectively, and the upper nodes the set of all dollars and the set of all bolts respectively. The labels on the arcs, e and s, therefore denote the relationships of set and subset membership respectively.

The key idea in Hendrix's approach to semantic networks is that of network partitioning. We can see the need for that if we consider a version of one of the classic problems of intensional logic :
Mr Smith may be both my neighbour and the Mayor of San Diego, but when discussing him with another I do not know whether my interlocutor knows Smith has all the roles he has. It will confuse the issue if I refer to Smith as "the Mayor", even though he is, if my conversational partner knows him only as my neighbour. In other words, I need some formal representation of Smith, one might say, so that only relevant sub-sets of his attributes are in focus at any one time. It is this that Hendrix's partitioning of networks aims to provide

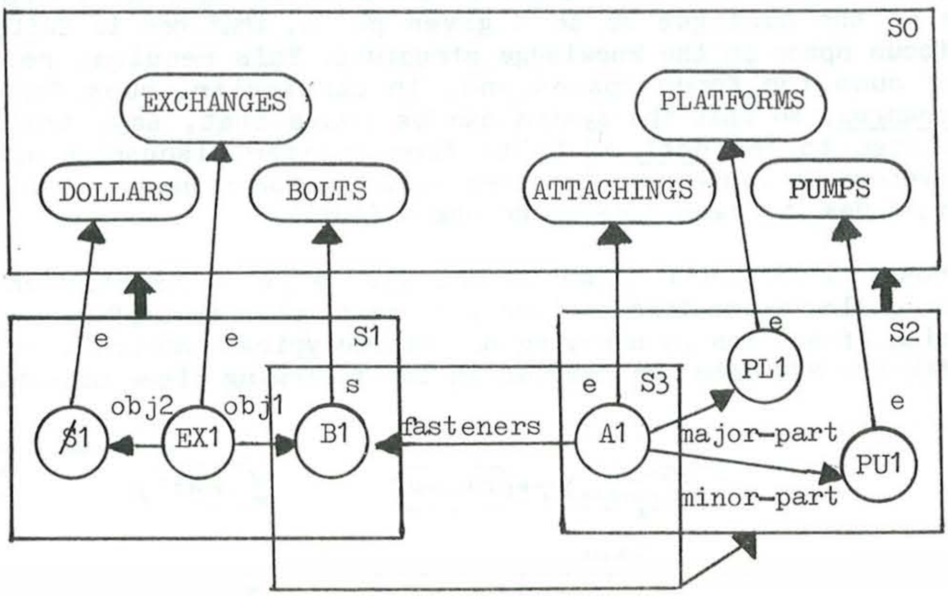


Figure 1 A Sample Partitioned Semantic Network

In this diagram we have four focus spaces S0, S1, S2 and S3. They are the overlapping partitions of this small sub-network of items, whose names appear in full in S0. The two key features of this form of organisation of knowledge are (a) that items within a focus space can be associated together in a dialogue : so that, for example, whenever an item is in focus, all other items that can be found in its vista of focus spaces are also in focus; where (b) a vista is a sequence of focus spaces such as (S3 S2 S1).

It is the vista sequences of focus spaces that exploit the fact that an item can appear in many focus spaces. Thus the set of bolts B1 can be seen either as an object of exchange (i.e. it can be bought for money) in the vista (S1, S0), or it can be seen as a necessity for fastening platforms to pumps, in the vista (S3, S2, S1). The broad arrows from space to space show the orthodox vista available from a given focus space.

The notion of implicit focus, required by the initial example, is given by the vista : when any item is in focus, then all the items in the vista that can be seen from the initial focus space are in implicit focus. So, if the bolt set B1 was in focus and we know we are in focus space S3 then everything in spaces S3, S2 and S0 is in implicit focus and can be drawn in to the conversation as appropriate.

Most importantly, this means that, in such a dialogue, the bolts are seen only as parts for attaching one component to another, and not at all as items to be bought for money. The difficult issue, of course, is focus space matching : of knowing for certain, from the course of the dialogue up to a given point, that one is "within" a given focus space in the knowledge structure. This requires reliable surface cues for focus spaces and, in particular, cues for focus space changes, so that the system can be aware that, say, the topic has shifted to the cost of bolts from earlier discussion of their effectiveness as fasteners. Detailed work on focus space changes in such dialogues has been done by Reichman (1978).

These partitioned networks are also able to express what we earlier called frame-like notions : in particular that of sequence or succession of actions in carrying out stereotypical actions. This is done with the SUC label on arcs as in the following piece of network:

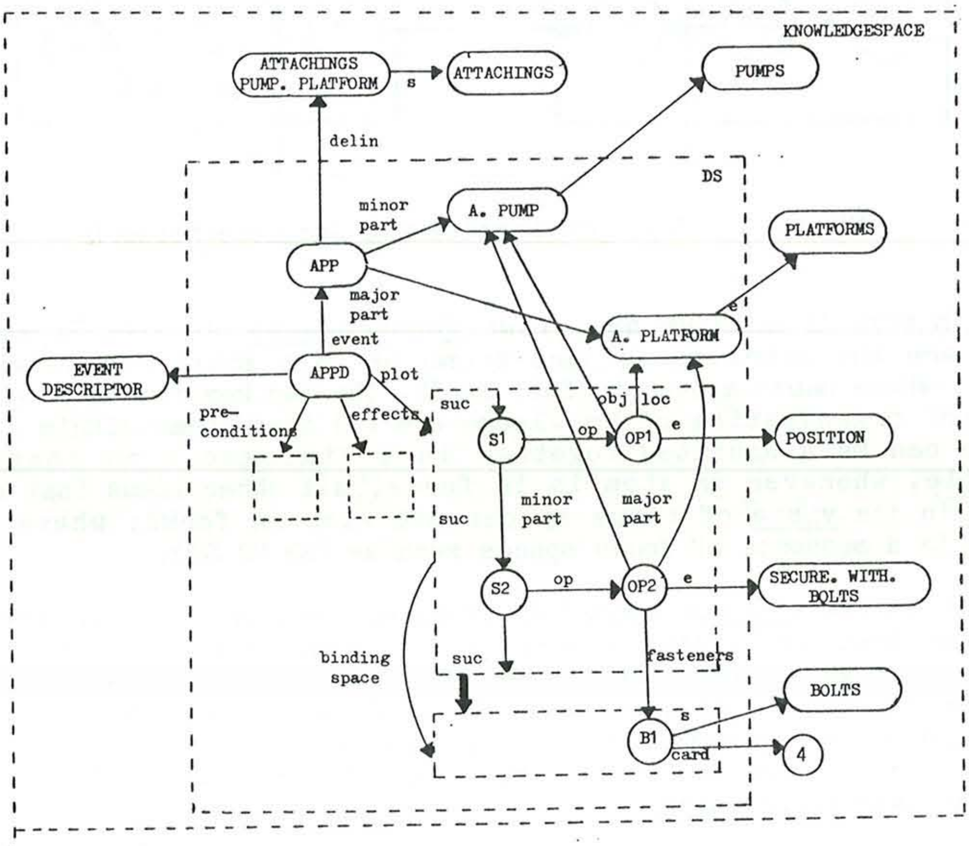


Figure 2 Event Encoding Showing Implicit Focus

The key node is APPD the event oof attaching a typical pump to a typical platform. The arc label PLOT from APPD gives the inner dotted sub-networks whose inner structure is the plot, or frame, of the attachment event. It has two stages S1 and S2, to which correspond bolting operations OP1 and OP2, where OP2 points directly to the bolt set B1 with the label FASTENERS. The label SUC links the two stages S1 and S2 and this arc is the dynamic element, that expresses the succession of assembly stages through time.

When one of the assembly/attachment operations is in explicit focus, as in the dialogues above, all nodes in the vista from that focus space (containing the operation nodes OP1, OP2) are in implicit focus; hence the introduction of bolts in implicit focus in response R2 above.

As a dialogue with the system proceeds, focus spaces open and close, while the dialogue in effect causes a path to be traced through the network. This is a far more flexible approach to focus than other systems explicating the notion within a frame-like paradigm (cf. Lehnert 1975), where what is in focus is taken to be whatever is mentioned in the dialogue, but is not represented in the knowledge structure (i.e. what is interesting in what you say is what I don't expect). This notion is too simple-minded, even for stereotyped situations like pump assembly, for the real pay-off from a system like Grosz's is the way in which she can locate the referents of pronouns in dialogues.

In Grosz's system only one focus space is active at any time, though others are open, and can be returned to. One of the most impressive features of the system is the way in which pronouns can be correctly referred to entities mentioned much earlier by only, and here is the point, if the entity's node is in a focus space still open.

There is also an interesting economy of effort assumption in the system : new sub-nets constructed during processing are deleted when the focus space to which they were added finally (in that particular dialogue) closes.

The Toronto station system

A group in Toronto (see, for example, Cohen and Perrault 1979) have developed a system designed to aid a traveller at a main station with dialogues like:

Passenger : The 3.15 train to Windsor?

System : Gate 10.

Here the passenger has aked a question, in common-sense terms, even though the grammatical form of his utterance is merely a noun phrase; the questionmark indicates the underlying purpose of the utterance, and not at all its syntax.

A well-known variant of this phenomenon of communication can be seen in a question from the passenger like :

"Do you know when the Windsor train leaves?"

This is a question, a request for information, as its grammatical form declares it to be, but it does not make the request it apperars to, if taken literally. For the system to answer "yes", on the grounds that it did know when the train left, would make the system useless.

Under the name "speech acts", much effort has been devoted in recent philosophy and linguistics to phenomena of this general sort. The problem has been isolated as recognizing the <u>type</u> of speech act being made by a given utterance (a <u>request</u> in the cases above, regardless of the accompanying surface grammar), and isolating the instance of that type that is present (recognising <u>what</u> exactly the request is in the last example).

The Toronto dialogue system is not just an expert in train services, but on the explication of speech acts : it is able to engage in real co-operative dialogues with passengers precisely because it is able to detect what the user actually wants to know. And that cannot be inferred in any simple way from the surface syntax and semantics of the utterance. Much theoretical work has assumed that such inferences can be made reliably, and has generated rules for the "deep grammar of questions", for example : the use of surface cues to determine that sentences of certain classes are <u>really</u> questions even though they have the surface form of, say, statements, as in "You have a light".

The Toronto approach is quite different, and based on <u>plan detection</u>. Users, the system assumes, will have one of a small number of plans in view, and to understand what is said to it, and reply appropriately, all it has to do is to determine <u>which</u> plan any given user has. In fact, the number of plans is only two : a user is either meeting a train (if the cues indicate the train referred to is coming <u>to</u> Toronto), or is catching one (if the cues indicate the train is leaving Toronto for somewhere else).

"Understanding an utterance consists in deducing the passenger's plan by seeing what expected plans could include the observed speech act...the constructed plan (i.e. by the hearer, system, <u>for</u> the speaker, passenger - YW) will be part of what it believes the passenger wants" (ibid. p 179).

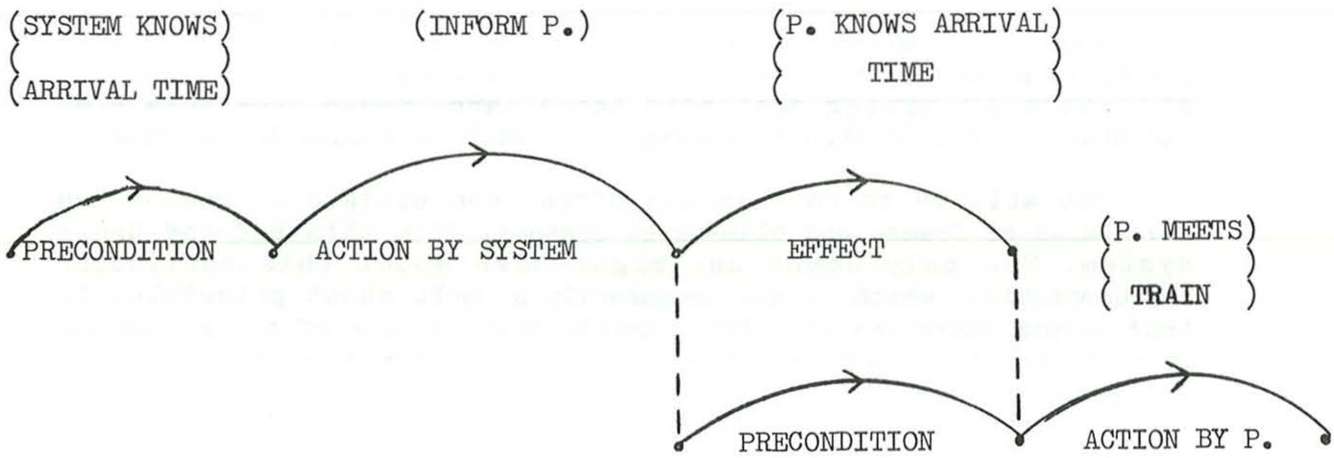
The system itself, of course, contains an overall plan to help passengers fulfil their plans. The actual form of the planning theory used is that of STRIPS (see, for example, Files and Nilsson 1971) and will not concern us here. What follows is an example of plan-detection adapted from (Cohen and Perrault 1979). Suppose the passenger asks :

Passenger : "Do you know when the Windsor train arrives?"

This is deemed to be what they call a REQUEST-INFORM (roughly a question, rather than a REQUEST, which is closer to an imperative) since the literal reading is rejected - the one in which such a questioner genuinely wants to know whether the hearer knows that - on the grounds that the passenger knows that the system knows the arrival time, and therefore cannot be asking it. Here one can contrast the question :

"Do you know how late the Windsor train is yet?"

where there is no such assumption. Thus, if we assume a simple network representation of states (nodes) and preconditions for them and effects of them (arcs to and from nodes) we can express this situation as :



One must think of the system inferring in two different ways : firstly, what we may call bottom up, in which the system interprets the question literally, but infers that the passenger already knows the system knows the arrival time (since it knows, and is known to know, all scheduled times). But, knowing that is a precondition for informing the passenger of the time, and that in turn, as an action by the system, will cause the passenger to know the arrival time. Now, and secondly, the system is also inferring, top down, from its guess that since an arrival is in question the user intends to meet it. An inference, backwards as it were (lower line in figure), from that user action to its precondition, is that the passenger knows the arrival time.

The overlap of effect and precondition as the top down (lower line) and bottom up (upper line) inferences is taken to confirm both the detected plan (of the passenger) and the appropriate informing action of the system. It then does inform the passenger of the arrival time.

This sort of inference chain is often called an indirect speech act (Searle 1969), but one of the most implausible features of the standard philosophical and linguistic analyses is that every time one asks "Can you pass the salt" intending the request "Pass the salt, please", there is supposed to be such a chain of reasoning from the "failure" of the literal interpretation, on the grounds that it is obvious that the hearer can pass the salt etc. One of the most interesting aspects of the Toronto approach is their proposal that such chains be contracted to become just macroprocesses in the system, to be called as wholes. This seems, not only a computationally efficient, but a psychologically insightful, view of such indirect reasoning.

Another interest of the system is that it is able, in a limited sense, to parse the input top-down; to interpret what the passenger says only by having available these large-scale "possible plan structures", rather than by working only with word-cues in the input.

The ability to do this has often been claimed as one of the advantages of frame- and plan-based systems, like this and the Grosz system. The only doubt one might have about this particular implementation, which is not necessarily a doubt about principle, is that since there are only two possible plans a user of the system can have (to leave or to meet), there is a slight air here of forging a sledgehammer to crack a nut.

There is also a possible problem with the system's data base in that all possible beliefs are already stored : the passenger's belief about what the system believes etc. There are good reasons for thinking that not all such computational "points of view" can be computed in advance (see Wilks and Bien 1979).

Some other systems

Concentration on the two above systems, within the narrow confines of this paper, has been purely a matter of personal preference. However, a number of other expert systems that have independent natural language interest should be at least mentioned.

GUS (1977) was a system designed to simulate a travel agent, one which not only knew flights, routed, etc., but which could schedule passengers optimally. Its natural language interest was that it was founded on the KRL (Knowledge Representation Language) concept. This was an attempt to construct a high-level programming language, based loosely on Minsky's frame concept, but which could express, within a single formalism, not only the relevant knowledge of the real world (flights, passengers etc.) but also syntactic and semantic knowledge of the surface input language, English, sufficient to parse it. Again, the claim was that the parsing was top-down, in that the knowledge of the real-world of flights to some degree determined the interpretation of the English input.

There have been a number of frame-based text understanders from the group at Yale University. The one closest to an expert system is that of deJong (1979); and it is, in some ways, the most interesting from the point of view of text analysis. Like most of the Yale systems it uses frame systems called scripts : in the present case the topic of the scripts is a concept like earthquakes. DeJong's system is able to search newspaper stories, coming in on the AP wire service, and to (a) determine their topic by their closeness to the script of a stereotypical earthquake (Richter-scale force, location, number of casualties etc.) and (b) to "parse" the story, when located, by filling in the variables, or slots, in the script so as to produce an acceptable paraphrase of the original story.

The key features of deJong's approach are, first, that the script contains only "sketchy" expertise about the topic and, secondly, that the script structure is itself used to parse the input.

At Bolt, Beranek and Newman, Woods (1970) has constructed a number of Augmented Transition Network (ATN) parsers over the years. These are essentially node and directed arc systems with registers attached to arcs. In a very simple ATN grammar, an arc would normally bear a label like NP (Noun Phrase) to indicate that to pass across that arc a noun phrase must be located in the sentence under examination. That label would be a pointer to another network that indicated in its structure the ways in which an English noun phrase could be parsed : that the noun was essential, but the article and adjectives were optional, for example.

Such networks normally had as labels such syntactic categories. However, Burton (1976) designed an ATN parser, as part of the SOPHIE system, that had as labels actual words from the vocabulary of electronic repair ("voltage" etc.). SOPHIE, although originally a computer-aided instruction system was also an expert on the repair of electronic circuits. The claim behind Burton's shift from standard syntactic ATN's to what he called "semantic grammars" was very like that behind all the systems discussed in this paper : that the expert knowledge can be expressed in a form such that it can itself direct the parsing, or understanding, of the English input.

But in Burton's case this knowledge was not in fairly abstract frame or plan structures, not even in "sketchy scripts" like deJong's, but in the networks that themselves expressed the permitted forms of English sentences. This was an interesting idea (though one considered and rejected right back at the beginning of work on the generative grammar of English, i.e. Chomsky (1957)) but is probably only applicable within very limited subject areas since the parsing is so "fragile" : the actual objects of every verb in the system, for example, must be explicitly specified. However, it may well be that, for the limited purposes of expert systems, that degree of explications is possible.

Conclusion

In selecting a few expert systems which also have interest from the point of view of natural language understanding I ignored Feigenbaum's very high standards for expertise : travel agents and electrical repairmen may reasonably be considered experts, but do not require MDs and PhDs.

Again, the features to be found in the most interesting systems discussed : points of view, communication strategies, models of the speaker, understanding via dialogue context etc., were not in any obvious way derived from the notion of expertise, and were all to be found in Colby's system, at least in rudimentary form, which is as far from an expert system as one can get!

So then, we may ask whether "interesting" natural language understanders or indeed any language understander at all, is necessary to an expert system at all and, if it is, whether any natural relation between the two emerges in this paper.

As we noted at the beginning, natural language understanding cannot be essential to an expert system. On the contrary, a number of considerations tell against the inclusion of a natural language front-end in an expert system : in particular, that an expert will get tired of using long forms and will begin to want to shift to an abbreviated semi-formal front-end as soon as possible. At the very least he will insist on a robust spelling and grammar corrector as part of the front-end if he is to continue using it (see Ball and Hayes, 1980, on this point).

Again, the tradition of natural language understanding within artificial intelligence can be seen as opposed to expert systems : many in that field have been engaged, so they believe, in a form of psychological modelling of human beings, with all the scope for ambiguity, vagueness and even misunderstanding that that implies.

On the other hand, one can argue that the distinctive feature of the artificial approach to natural language understanding has been the claim that knowledge structures are essential to language understanding, and what are expert knowledge structures but these?

I believe one can set up a spectrum of approaches, ranging perhaps from a linguistic view like Chomsky's at one extreme (on which no real-world knowledge need be involved in a natural language specification system), to systems like Burton's at the other end in which no distinction can be made between the expertise and the natural language parser. These last systems tend to be very fragile and of limited value, except withing the narrowest domains.

I believe one can make out a case for a middle position, to which all the systems described here (except Burton's) conform : one in which there is a parsing module, containing real world knowledge, but not all the expert knowledge the system has. This module does however have the knowledge of language the system has, but is detachable from the general expertise module. The operation of the system then consists in interaction between these modules, so that expert knowledge can impose an interpretation on what is located by the parser : as with the passenger's plans determining the interpretation of the input in the Toronto system. The crucial issue on this view would be whether there was any separation at all between the parsing and the expertise.

This is not yet a settled question, and much of general interest hangs on it, such as whether humans have any general knowledge of their language, over and above their areas of particular expertise. The degree to which such expert systems as those described here are modular, will therefore be of great theoretical as well as practical interest.

Discussion

Following the first lecture, **Dr. Woods** expressed doubt as to the possibility of just ignoring non-function words. It might work on some examples, but would certainly fail in others. **Professor Wilks** pointed out that as people mis-understand sentences too, perhaps some such errors could be tolerated. There was something of a puzzle in the fact that though a top-down approach did not work - one could not ignore the low level detail-nevertheless bottom-up approaches did not perform much better. **Professor Seegmuller** enquired about the timescale of likely practical applications. **Wilks** was not over-hopeful, and doubted if they would appear from Universities, but pointed out the enormous possible impact in areas such as welfare payments. **Dr. Barrow** then led a short discussion on whether the notion of 'script' could be useful at the syntactic level. 'John ran a mile' might violate the rules for 'run' but be usefully analysable by some form of template matching.

Referring to the Toronto station system, **Dr. Betteridge** and others were unhappy at the elaborate treatment suggested for information-requests framed as questions, as in 'Can you tell me when the Windsor train arrives'. They consider 'yes' an adequate reply, thus forcing the questioner to be more precise. **Dr. Woods** disagreed strongly. If the questioner's pre-suppositions are false, it is extremely misleading for a system not to point it out, but simply answer 'no'. **Miss Barraclough** felt one could combine literal question answering with the learning ability to notice common sequences of questions, and then answer successor questions before they were asked, but **Professor Wilks** was adamant that he was neither concerned with modelling learning, nor even wanted systems that behaved like that.

References

Ball, E. and Hayes, P. (1980). "Representation of task-specific knowledge in a gracefully interacting user interface", Proc. First Annual National Conference on Artificial Intelligence, Stanford, California, pp. 116-120.

Bobrow, D., Kaplan, R., Kay, M., Norman, D., Thompson, H. and Winograd, T. (1977). "Gus, a frame driven dialog system", Artificial Intelligence, pp. 46-72.

Burton, R. (1976). "Semantic Grammar", BBN Report No. 3453, Cambridge, Mass.

Colby, K.M. (1975). "Artificial Paranoia", Pergamon Press, New York.

Chomsky, N. (1957). "Syntactic Structures", Mouton, The Hague.

Cohen, P. and Perrault, R. (1979). "Elements of a plan-based theory of speech acts", Cognitive Science, pp. 273-303.

DeJong, G. (1979). "Prediction and substantiation", Cognitive Science, pp. 251-273.

Fikes, R. and Nilsson, N. (1971). "STRIPS : a new approach to the application of theorem proving to problem solving", Artificial Intelligence, pp. 189-208.

Grosz, B. (1979). "The representation and use of focus in a system for understanding dialogues", Proceedings of the International Conference on Artificial Intelligence, Tokyo, pp. 67-76.

Hendrix, G.G. (1975). "Partitioned networks for the mathematical modelling of natural language semantics", Tech. Report NL-28, Dept. of Computer Science, University of Texas.

Lehnert, W. (1975). "Question answering in a story understanding system", Yale University Computer Science Dept., Research Report No. 57.

Minsky, M. (1975). "A framework for representing knowledge", In The Psychology of Computer Vision (ed. P. Winston), McGraw Hill, New York pp. 211-277.

Reichman, R. (1978). "Conversational Coherency", Memo No. TR-17-78. Aiken Computation Lab. Harvard University.

Searle, J. (1969). "Speech Acts", Cambridge University Press, Cambridge.

Shortliffe, E.H. (1976). "Computer based medical consultations", Academic Press. New York.

Steels, L. (1979). "The XPRT Description System", Working Paper 178. Massachusetts Institute of Technology, Artificial Intelligence Lab.

Weizenbaum, J. (1976). "ELIZA", Communications of the Association for Computing Machinery. Vol. 10, pp. 474-80.

Wilks, Y. and Bien, J. (1979). "Speech acts and multiple environments", Proc. Internat. Joint Conference on Artificial Intelligence, Tokyo, pp. 471-475.

Woods, W. (1970). "Transition network grammars for natural language analysis", Communications of the ACM Vol. 13 (10), pp. 591-606.