Panel: Expert systems: How far can they go?

Terry Winograd, Randall Davis, Stuart Dreyfus, Brian Smith

We arc in the midst of a great wave of enthusiasm about the potential for expert systems in every area of human life and work. There is no agreement, however, as to just how much they can do, and where they will run into fundamental limits. The intent of this panel is to present and discuss some basic questions as to what expert systems can really be expected to do:

What is the nature of the problem domains in which expert systems are likely to succeed and those in which they will not? Are there domains in which their use might be dangerous?

How will their performance compare with that of human experts in the domain? Are there different facets of expertise that are not amenable to programming? How can human and machine expertise best be combined?

To what extent can we count on rule-based systems for 'flexibility' in dealing with unexpected situations? How reliable will such systems be in cases where the programmers (or knowledge engineers) did not anticipate significant possibilities?

How can a 'knowledge base* be subjected to standards of accountability? Who is responsible for what an expert system contains and what it does?

Expert systems:

What to do until the theory arrives

Randall Davis

MIT

In reading a newspaper recently I was struck by the profusion of expert advice available. There were three different expert opinions on the future course of the economy, several compelling (and contradictory) opinions about the likely course of events in the mideast and a number of suggestions about avoiding heart disease, as well as claims about long term weather patterns, advice from Ms. Manners on behavior and guidelines from a therapist on drinking and sex.

All of which made me begin to wonder:

Experts: How far can they go?

We are in the midst of a great wave of enthusiasm about the potential for experts in every area of human life and work. There is no agreement, however, as to just how much they can do, and where they will run into fundamental limits:

What is the nature of the problem domains in which experts are likely to succeed and those in which they will not? Are there domains in which their use might be dangerous?

To what extent can we count on carbon-based systems for 'flexibility' in dealing with unexpected situations? How reliable will such systems be in cases where their teachers did not anticipate significant possibilities?

How can a person's knowledge be subjected to standards of accountability? Who is responsible for what an expert contains and what that person does?

Hardly an original satire, but it does serve several purposes. First, it demonstrates that the questions are neither unfamiliar nor inherently mysterious. The answers for people may not be well established, but we do know something of how to proceed and we do believe there is no magic here: *some* form of knowledge accounts for an expert's competence. The nature and source of it may be far from understood, but that doesn't make the question unanswerable.

Next, it sets the argument out on what I believe to be an important direction: the questions we are asking are first about knowledge and only then about technology. That is, the first question should not be *What can expert systems do?* but rather *What do we know?* Only then we can address the technology issue and ask *And how easily can wc encode that knowledge?* Both matter but the order is important

Finally, the comparison is valid and provides an interesting way to proceed. Asking the same questions about people provides a useful, non-thrcatening way of examining the topics. Our answers may differ for people

and programs, but even those disparities will prove interesting. We explore peoples understanding of a subject with a test that examines only a limited sample of their knowledge, and then extrapolate, saying that people who pass 'understand' the material, meaning by that something more than that they can do exactly the problems chosen for the exam. If a program passed the same exam, would we be willing to do the same extrapolation? If not, why not? Jf we can determine what it is that makes us hesitate in the case of the program, we have the beginnings of an intriguing research agenda.

In developing these themes I will argue that there are two attributes of expert systems (and much of AI) that are central to this discussion:

It is a weak technology

It is a technology for dealing with incompletely understood ideas.

I will suggest that the first of these is a temporary vice that will be remedied in time (though not soon) and that the second is a permanent virtue.

Both of these have interesting implications for the use of expert systems (and indeed much of AI). Perhaps the most important implication is that most traditional rule-based expert systems will never have all the knowledge they need, and as a consequence they are guaranteed to fail occasionally (though perhaps infrequently) during all of their operational lives.

I will suggest ways of proceeding that take these issues into account, allowing us to employ the technology while reducing the potential difficulties that can arise.

The nature of expertise

Stuart E Dreyfus

University of California, Berkeley

All AI work with the exception of a few 'connectionist' theories assumes that knowledge must be represented in the mind as symbolic descriptions. Expert-system builders further assume that the expert possesses a particular kind of symbolic description: a knowledge base of facts, beliefs and 'if-then' rules that allow the drawing of inferences.

1 will argue that expert-system builders fail to recognize the real character of *expert* human understanding. Expertise is acquired in a five-stage process. The *beginner* applies rules to context-free features as would an expert system deprived of situational knowledge appropriate to the particular case. The *advanced beginner* learns from experience to recognize aspects of a situation without requiring a definition of them in terms of context-free features.

Aspects arc recognized after seeing several examples, apparently because of their similarity to already experienced prototypical cases. An interactive expert system could use aspects if they were identified for it by a human user. At the next stage, the *competent* performer organizes behavior by selecting plans, goals, or perspectives which determine hierarchically what facts to consider and what rules to apply. Expert systems can do likewise and, if they accept human situational assessments, could appropriately be called 'competent systems.' This, however, is the best they can do. The fourth stage, *proficiency,* is achieved when the performer no longer uses his knowledge to select a perspective or goal, but simply recognizes the appropriate one based on prior experience in similar situations in which goals were chosen and events either confirmed the wisdom of the choice or showed it to be mistaken. As with the recognition of situational aspects by the advanced beginner, the involved, intuitive recognition of similarity of whole situations is apparently not produced by rules operating on features but seems to be effortless and holistic. While the proficient performer still analytically figures out what to do once the situation is intuitively understood, at the highest level of skill the *expert* has experienced so many situations that he associates with each prototypical situation in his memory the decision, action or strategy that he has found to work. He reasons out neither strategy nor action. Intuitively responding to situational patterns as experience has shown appropriate, his skill depends neither on problem solving nor planning. Expert systems can neither recognize situations holistically without analysis into components nor know what to do without applying rules to decomposed knowledge, so they can be neither proficient nor expert.

If time permits, an expert will deliberate about his intuitive understanding. To fine-tune his responses, he will attempt to take account of subtle differences between his current situation and similar prior ones, he will ask himself whether there might be another quite different way of intuitively viewing his circumstance, and he will consider whether he has had enough experience in the particular kind of situation to trust his intuition. But he will rarely regress to competent detached problem solving.

While 'competent systems' have their useful place, there is no reason to expect them to perform as well as experts who have passed beyond the use of facts, beliefs, and rules of inference and who rely on memories of thousands of concrete experiences and what has worked in each. An examination of the performance of various expert systems supports the above analysis. In domains where human beings pass from reasoning to recognition as they become experts, expert systems, even when experts participate in their development, never perform as well as experts.

Models in expert systems

Brian Smith
Xerox PARC

All expert systems are based on models. The 'knowledge' embodied in expert systems, in particular, is usually encoded in a set of 'rules' that describe the problem and specify the behaviour that the system should manifest. These rules are always formulated with respect to a model of the underlying domain. This model must be determined in advance by the programmer, who may in turn have derived it from an analysis of the experts' performance on which the system is based.

Indeed, one of the prime tasks in building an expert system is to develop an appropriate model. There are various ways to do this: by analysing the desired behaviour, by building on underlying scientific theories, or by codifying the models apparently used by expert human practitioners. What phenomena are dealt with, what phenomena are ignored, and what patterns or regularities connect the phenomena that are dealt with -- all these decisions are made at the level of the model.

For example, a medical expert system designed to administer drugs might model drug absorption in terms of a scalar quantity proportional to the square of a patient's height, or proportional to the weight (neither model, of course, would be expected to be entirely accurate). An expert system for the office might model a secretary as a customer, producer, and processor of information, with a complex internal state. A defense warning system might model incoming missiles as point masses on parabolic flight paths, and model the atmosphere as a linear retarding force. And so on and so forth: the use of models permeates formal systems of all sorts.

When expert systems are actually deployed, however, they interact with the world itself, not with models. For example, when drugs are actually administered, or when offices are actually equipped with expert systems intended to work alongside people, we have full, thick situations to deal with, of at least potentially arbitrary complexity. Furthermore, the success of expert systems ultimately depends on their ability to deal with these rich, embedded situations. Their success, in other words, isn't exhausted by their ability to deal appropriately with the model used in their construction, or encoded in their knowledge bases.

In fact the only ultimate point of the models in expert systems is to help them succeed in the 'real world'. RCA, for example, is primarily interested in whether their satellites will actually get into orbit and stay there; they have only a derivative interest in whether the programs guiding them are proved correct with respect to a particular orbital model.

It is clear, therefore, that in order to analyse expert systems we need to understand the appropriateness of the models on which they are based. Analysing expert systems, in other words, comes in two parts: understanding the behaviour of a system in terms of the model on which it is based, and understanding the relationship between that model and the embedding world. At the present state of the art, we have a variety of techniques that enable us to study the former relationship, between system and model: formal semantics, model theory (hence its name), program verification. We have virtually no techniques, on the other hand, with which to study the latter relationship, between model and world. We are largely unable, therefore, to assess the appropriateness of models, or to predict when models will fail. All that we do when we prove a program 'correct' is to prove that it will behave as specified with respect to a model. It would be something quite else -- something we don't know how to do - to prove that a system will in fact do the 'correct' thing once embedded into a real situation.

Two conclusions. First, we should develop and use expert systems only in those domains where we have confidence in the accuracy and appropriateness of our models. Second, we should develop a 'theory of models' with which to understand better how models work, and how they make sense of the infinite complexities of the worlds they represent. Although such a theory may strain at the edges of what can be formalized, or even pass beyond strict formal limits, we can still develop rigorous tools, and use clear-headed thinking, in analysing this important relationship.

The trivialization of expertise

Terry Winograd
Stanford University

Professor Moto-Oka of the Japanese Fifth Generation project [3] predicts that: *"Fifth generation computers are expected to function extremely effectively in all fields of society. ...totally new applied fields will be developed, social productivity will be increased and distortions in values will be eliminated "* Feigenbaum and McCorduck [2] proclaim that: *"We are now at the dawn of a new computer revolution... [leading] to computers that reason and inform ...the engine that will produce the new wealth of nations.... Perhaps equally important to all the economic advantages the Fifth Generation promises is that intangible thing called quality of life. A society where knowledge is quickly and easily available to anybody who wants it will... be an alluring place."*

We might dismiss these statements as merely naive self-serving propaganda, but in doing so would we fail to

recognize the background in which they could be made sincerely and taken seriously by intelligent and educated readers. The computer is a powerful embodiment of a 'rationalistic' tradition that equates certain limited modes of rational description and inference with the full scope of how people think and what they do with language. This tradition is both demonstrated and further promoted by inflated claims for the potential benefits of 'expert systems.'

Rationalistic understanding has been extremely powerful in the creation and expansion of the physical sciences and in increasing our mastery over the physical world. At the same time, its power in dealing with these domains of reality has blinded us to its weakness in dealing with those more related to human life and society. The rationalistic orientation often promotes a wrong understanding of what constitutes a 'problem' in a human domain, and what computers can do to 'solve problems.' The failure of the rationalistic approach has led to a crisis in the practical areas where it has been applied seriously. The recognition of this crisis has emerged in the past few years in works on management theory, military systems, medicine, energy policy, and many other fields.

The problem is illustrated by the illusion promoted by the label 'expert system.' A human 'expert' is someone whose depth of understanding serves not only to solve specific well-formulated problems, but also to put them into a larger context. We distinguish between experts and idiot savants. What, in contrast, do computer systems do?

Expert systems are built on the basis of a relatively small, precisely defined set of object types and properties, together with a (possibly large and disorganized) collection of rules relating them. In pre-selecting the relevant elements out of which to build rules, one must cut out the role of context and background (which are at the heart of AI's theoretical difficulties). This process by its very nature creates blindness - a limit is set by the way the world has been articulated. There is always the potential for breakdowns that call for moving beyond this limit -- for returning to the context and reformulating the problem.

One might argue justifiably that this blindness is a problem for people as well. But (with the exception of idiot savants), people are not programmed for a particular task. A normally intelligent person can always step back and recontextualize. Expert systems, even those with 'meta-rules' do not have this openness. Further, the blindness inherent in representation leads to difficulty in understanding the range and limitations of a particular program. A human expert can enter into a dialog about his or her own range of knowledge and limitations, moving outside the putative domain and using ordinary language and ordinary common sense. Failures of AI programs will in part reflect the inability of people to understand what the program is actually doing, as opposed to what it might appear to be doing if the metaphor of 'thinking' is accepted.

Although the assumptions of the rationalistic orientation may seem self-evident (to those within our modern Western society), they are indeed only assumptions and can be challenged (see [4] for a more comprehensive discussion of this and the other issues raised in this paper). Instead of treating 'data* as the objective representation of reality, we can recognize symbols (on paper, or in a computer) as a medium for language, which is based on human commitment and is always relative to an unarticulated shared background. Instead of trying to get the machine to have 'understanding' through a collection of 'rules,' we can recognize that its manipulation of symbols is grounded in the background and interpretation of people who interact through it. We can identify and articulate 'systematic domains' of symbolic manipulation for which appropriate rules can be generated, and we can integrate these into a broader appreciation of human knowledge and expertise.

References

[1] Dreyfus, Hubert L., and Stuart E Dreyfus, *Mind Over Machine,* New York: MacMillan/The Free Press, 1985.

[2] Feigenbaum, Edward, and Pamela McCorduck, *The Fifth Generation: Artificial Intelligence and Japan!s Computer Challenge to the World,* Reading, MA: Addison Wesley, 1983.

[3] Moto-oka, T., Keynote speech: Challenge for knowledge information processing systems, in Moto-oka, T. (Ed.), *Fifth Generation Computer Systems: Proceedings of International Conference on Fifth Generation Computer Systems,* Amsterdam: North Holland, 1982.

[4] Winograd, Terry, and Fernando Rores, *Understanding Computers and Cognition: A New Foundation for Design,* Norwood, NJ: Ablex, 1985.