Human Window on the World (1985)

At the meeting in Toronto in 1977 of the International Federation for Information Processing, Kenneth Thompson of Bell Telephone Laboratories presented a computer program for playing the chess end-game of King and Queen against King and Rook. He had done this by the ultimate in 'hammer and tongs' methods: in the absence of a complete set of rules for playing the end-game, he had previously programmed the machine to work out what to do in every single possible position—and there were four million of them. This was done backwards, by taking every position and working out what the best-move predecessor would have been. All these moves were then loaded into a gigantic 'look-up' table in the machine's memory, each entry in the table simply saying, 'If the pieces are in these positions, move this piece there.'

It is known from the theory of chess that given best play, this end-game is an inevitable win for the Queen's side, except for a few special starting positions. Chess masters can ordinarily guarantee to win against any opponent. So when playing with the Rook, Thompson's program merely made whatever move would stave off defeat for longest. Present at the conference were two International Masters, Hans Berliner, former World Correspondence Chess Champion, and Canadian Champion Lawrence Day. Thompson invited them to demonstrate winning play for the Queen's side against the machine. To their embarrassment they found they could not win, even after many attempts. Yet every position they were confronted with in the entire course of play was a winning one for their side.

The machine repeatedly conducted the defence in ways which to them were so bizarre and counter-intuitive that they were left grasping air, time and again

missing the best continuation. For example, the cardinal rule which chess players learn about this end-game is, 'Never separate King and Rook'. The assumption is that the Rook needs the King to help protect it from the Queen. Yet the super-table separated the King and the Rook again and again, having found some path, however narrow and convoluted, through the problem space that maximally postponed its supposedly inevitable doom.

Naturally Berliner and Day found the experience upsetting. They wanted to ask the program to explain its strategy, but this of course neither it nor its author could do. The answer in every case was, 'It's in the table.' Its knowledge was comprehensive but there was no representation of the knowledge in terms of goals, opportunities, risks, themes, tactical ideas, and the rest of the rich conceptual structure in terms of which chess masters frame questions and receive answers. The machine was in no position to give answers like: 'At this stage White must drive the enemy King onto the edge of the board.' What it was lacking was a conceptual interface whereby the machine and the human could share knowledge in forms which humans could grasp, namely, concepts. It is the task of knowledge engineering to design and construct such conceptual interfaces to allow people (who are still much more intelligent than machines) and machines (which are already much cleverer than people) to understand each other.

HAZARDS OF THE SUPER-TABLE

It may be said that chess is just a game. But let the reader generalize a little. Thompson's super-table is not an unrealistic example. While the search for solutions to difficult problems struggles slowly ahead, electronic technology is galloping. This has been bringing the price and physical size of computer memory down at an unheard-of pace.

Trillion-bit memories are already in existence, and Lawrence Livermore Radiation Laboratories have issued specifications which call for this capacity to be pushed up by a factor of several thousand. Optical storage promises to exceed even these scales of capacity. Such changes will inevitably tempt people to set up in such memories huge databases of questions and answers in a very wide

range of subject areas, wherever problems need to be solved. While these might appear a boon to man, they actually pose a major social hazard.

At first sight the ability to hold in a crude fashion trillions of questions paired with their answers might seem not very useful, but in fact most practical knowledge can be expressed in this form:

'What is the square of 961?' '31.'
'What is the right thing to do when lost?' 'Ask a policeman.'
'What is the freezing point of the seas?' '−2°C.'
'What is the truth-value of Fermat's Last Theorem?' 'Unknown.'*

Computer technology seeks today to move into tackling difficult problems of the sort computers now cannot solve, problems for which there is no straightforward procedure which in a feasible number of steps can find the answer directly from the question by calculation. But it often happens that although a problem is difficult, its inverse is not. For instance, calculating a square root is quite involved, but finding a square is easy. So a schoolchild might consider it more economical to work out the squares of every number he or she could conceivably be asked for and fill a huge table with the answers (listing the answers, not the questions, in numerical order, perhaps with some interpolation to fill in gaps). Then, whenever a square root is needed, it is looked up in the table. This is the 'inverse-function method', by which Ken Thompson's chess-playing program was built. But as we saw, this technique has one major drawback: the result is inscrutable to human users.

SOCRATES AGREES

One might say that a race of blind question-answerers such as this which so debases—by dispensing with—human understanding and judgement would be better uninvented. Interestingly enough, this argument was first raised over 2,300 years ago by Plato. In the *Phaedrus* he has Socrates tell a story about the

* This article was written in 1985.

Egyptian god Thoth, who goes to the god-king Thamus and says: 'My Lord, I have invented this ingenious thing called writing, and it will improve both the wisdom and the memory of the Egyptians.'

Thamus replies that, on the contrary, writing is an inferior substitute for memory and understanding. 'Those who acquire it will cease to exercise their memory and become forgetful; they will rely on writing to bring things to their remembrance by external signs instead of on their own internal resources.'

Socrates cites Ammon against the fallacious view that 'one can transmit or acquire clear and certain knowledge of an art through the medium of writing, or that written words can do more than remind the reader of what he already knows on any given subject.' In other words, men will be led to think that wisdom resides in writings, whereas wisdom must be in the mind. 'You might suppose,' Socrates adds, 'that written words understand what they are saying; but if you ask them what they mean by anything they simply return the same answer over and over again.'

In short, Socrates' complaint is that writing fails to pass Alan Turing's famous test (by which a machine can prove it is really intelligent if it can fool a questioner, over a teleprinter link, into thinking he is conversing with a human being). And so it does fail. If it could explain what it contained, we could say in a sense it 'understood' and so was showing intelligence. As writing fails the Turing Test, so too will the trillion-bit question-answerers of the future. But like writing, they will assuredly survive and help to change our world. Will this be good or bad? Unless the substance of Socrates' complaint is seriously investigated in the new context, these giant question-answer systems will be a mixed blessing and could on occasion get their users into trouble. Such databases, remember, store only the basic elemental unvarnished facts of the given case, and contain nothing corresponding to understanding, inference, judgement, classificatory concepts, and the like. Truly, '... if you ask them what they mean by anything they simply return the same answer over and over again'.

So long as the contents of the electronic super-table remain purely factual in the ordinary sense, then nothing worse is likely to result than exasperation.

Infallible answers obtainable on tap, over unimaginably vast domains of discourse, will be readily accepted. But the absence of any explanations to accompany the answers will be taken by the users in bad part. 'Why,' a chemist user will say, 'does this pattern from the mass spectrometer indicate that the unknown compound is some particular poly-keto-androstane?' Answer: 'Because the trillion-bit dictionary says so!' The chemist then asks, 'How does it know? How did that answer get there in the first place'?' If the super-table has been constructed by the inverse function method, even telling him exactly how it got there will not make him much the wiser. He and his colleagues may be goaded into building new explanatory theories of what they find in their super-tables. If so, then this is to the good, and presages new pathways of scientific advance.

THE LUNATIC BLACK BOX

On the other hand, a table of question–answer pairs is not restricted to encoding factual information of this kind. The format lends itself equally well to expressing strategies, with the table consisting of situation–action pairs. This is exactly what Ken Thompson's chess program consisted of, and we have seen the problems that led to. But what if the system were doing something of social importance, such as managing a complex control function in factory automation, transport or defence? Two supervisors, let us imagine, are responsible for intervening manually in the event of malfunction. The system now does the equivalent in industrial or military terms of 'separating its King and Rook'. 'Is this a system malfunction?' the supervisors ask each other. They turn to the system for enlightenment. But it simply 'returns the same answer over and over again'.

The problem becomes of global importance when the system being operated is in air traffic control, air defence, or nuclear power. It is not too difficult to decide that a human decision-taker, say, a policeman directing the traffic at a crossroads, is drunk or mad. But US plans for air traffic control envisage ultra-powerful database and scheduling computations encapsulated in giant 'black boxes'. What will the human supervisors do on the presumably rare occasions when East Coast flights are mysteriously re-routed to Dallas, or inexplicable

groundings of harmless carriers raise doubts as to the system's sanity? As control devices and their programs proliferate, their computations may more and more resemble magical mystery tours. Most critical of all, if an air defence warning system suddenly says, 'There are twenty Russian missiles heading this way,' before the officer in charge pushes the Doomsday button he must be able to ask, 'What makes you think that?'

Any socially responsible design for a system must make sure that its decisions are not only scrutable but refutable. That way the tyranny of machines can be avoided.

There is of course a method of solving difficult problems that is totally different to the use of super-tables, namely, exhaustive searching through branching trees of possibilities: 'look-ahead', as when working out the outcomes of possible chess moves and choosing the best. Tables—we could call them 'look-up systems'—require vast amounts of data storage but little processing. In contrast, in order for a look-ahead search to be completed in a tolerable length of time, a great deal of processing power is needed but little memory. These two extremes are shown in Figure 1.

What happens when you get a pronouncement from a look-ahead system and you ask it 'Why?' Can it tell you anything? Most certainly! It can detail all the calculations it did in sequence. It can even disgorge the entire analysis tree. Could anyone wish for a more profound response?

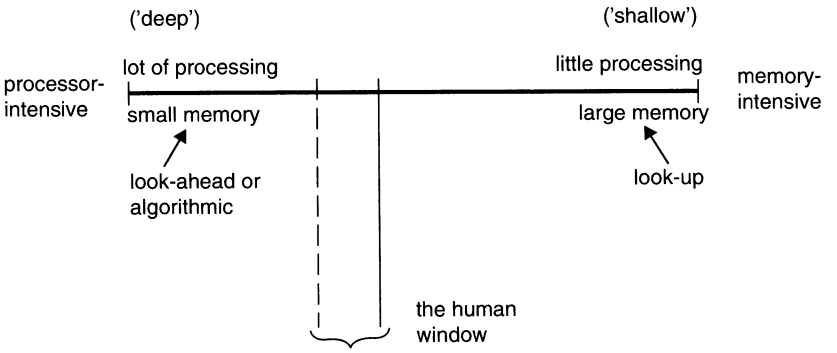

Figure 1. The spectrum of processing versus memory.

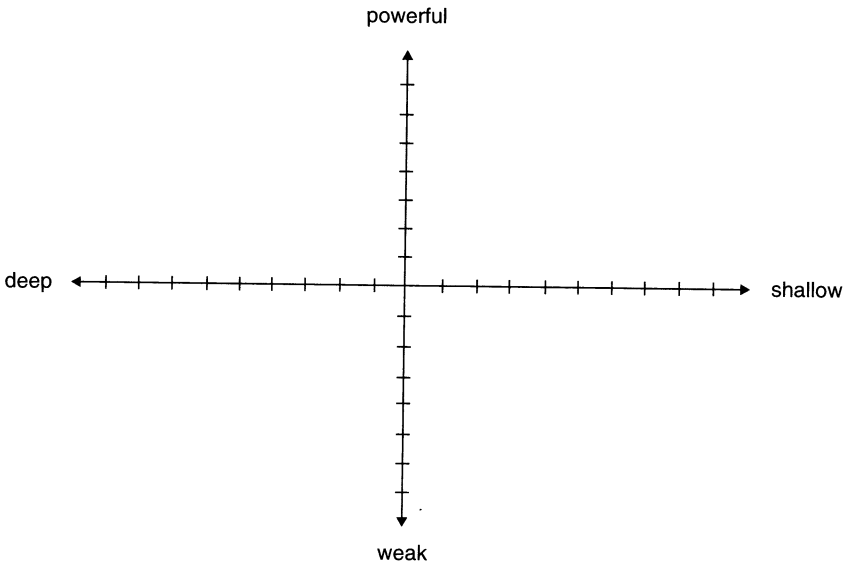

Figure 2. Two dimensions of a computer embodying an intellectual skill.

On the contrary, no mortal mind could possibly digest so much information. The tree could contain a million nodes, or a hundred million! The Three Mile Island fiasco is to the point—the operators made more mistakes, not fewer, because they were deluged with alarm signals, meter readings, and computer printouts. While a look-up system is too shallow in that it gives too little information, a look-ahead system tends to be too deep by giving too much. This is a separate issue from the power of the system—how much it is capable of doing. This distinction is shown in Figure 2.

THE HUMAN WINDOW

On the scale shown in Figure 1, 'deep' systems are at the processor-intensive end while 'shallow' ones are at the memory-intensive end. Somewhere in between is a narrow band where both the processing capability and the scales of memory are equivalent to those possessed by humans. We call this the 'human window', and it is here that computers must operate in order to be comprehensible to us whom they are intended to serve. Both the reasoning power required and the

way in which information is held must be on a human scale—elsewhere lies inscrutability.

A view which we shall call 'technomorphic' goes as follows: 'The machine's way of going about chess, or weather prediction, or plant control, or route scheduling, is bound to be different and ought to be different. The relative costs and constraints associated with the various aspects of the problem-solving process are quite disparate for machines and brains. Strategies which optimize performance with respect to two such contrasted profiles are doomed to diverge. Whatever way is most efficient for the machine to do the problem is the way we want to go. If Karpov has not got the calculating speed and working memory to grow a mental look-ahead tree of a million board states, or if our top meteorologists are not smart enough to be able to do partial differential equations in their heads, that is just too bad. Why should the programmer seek to copy their defects?'

From the point of view of optimizing the use of the machine the technomorph is right. But in the light of the brain's woeful disabilities as regards storage and processing speeds, efficient machine programs are not workable as representations for people. Where the technomorph goes wrong is in supposing that there is no criterion involved but machine efficiency.

Futurologists, in particular I. J. Good and Ed Fredkin, director of MIT's celebrated Project MAC, have speculated about the development of an 'ultra-intelligent machine' which would be able to 'reprogram itself within hours, constantly improve itself and rapidly become hundreds of times smarter than human intelligence'. Some people are worried about this. But the real social danger, certainly the first we shall see becoming manifest, is not the ultra-intelligent machine but the ultra-clever machine. The dangerous system is the one tuned by economic pressures to perform its task with machine-efficient inscrutability. These machine-oriented criteria can be shown to be irreconcilable with easy communication of concepts between man and machine. So performance must be sacrificed for the sake of transparency. Is that an economically acceptable sacrifice? Surely it is. Machines continue to become cheaper; human beings on the other hand do not. Adding artificial intelligence to the machine can offer the needed humanizing bridge. But if machine-optimality rather than human-optimality remains the design criterion, we are ultimately headed towards a technological black hole.

SYNTACTIC SUGAR IS NOT ENOUGH

So how should we design our machines to fit the 'human window'? The answer is not as straightforward as it may seem. Interactive diagnostics and trace routines, even when sprinkled with the very best syntactic sugar, do not necessarily suffice. Such things resemble orthopaedic shoes built to correct a patient's rolling gait: they may help, but if his trouble stems from a congenital abnormality at the hip joints, then the patient also needs reconstructive surgery. Just as there are walkable and non-walkable skeletal structures in human anatomy, so there are explainable and non-explainable computations, and the differences can be traced to the respective program structures.

Putting it another way, the addition of a simple 'user-friendly front end' when the subject area is very complex is like distributing powerful telescopes to inhabitants of Dover anxious to gaze upon the Eiffel Tower. To people ignorant of the curvature of the earth it could seem like a good idea.

In order for any beings, human or machine, to talk to each other, they must share the same mental structures. People's mental structures cannot be changed, so we must change the machines'. We need to restructure the entire way problem-solving programs do their jobs, not just how they interact with the user. The way the program holds information—its problem representation—must be recognizable to a human as a concept with which he is familiar. Both Ken Thompson's table and the weather-forecasting differential equations are non-starters in this respect. Rule-based expert systems on the other hand are specifically designed to operate with human concepts, both accepting them from the domain specialist and displaying them to the user as explanations. These provide a start, but much research still needs to be done on the technology of the conceptual interface.

SOFTLY, SOFTLY AUTOMATION

We call the application of these ideas to factory equipment and other control systems 'soft automation'. This is increasingly needed for cleaning up the complexity pollution which hard automation tends to generate. The greatest

social urgency attaches not to extending automatic processes but to humanizing them. Of course, for tasks of low-to-middling complexity, opacity is not really a problem. We have lived with it for a long time without any ill effects. Suppose that a resource allocation program schedules a job better than a human project director. How much desire does he feel to pry into its detailed workings or to argue with it, so long as it is doing what he wants? It can be as much of a 'black box' as it chooses.

However, there are other applications for which an 'open box' mode is essential. As yet, there are few of these, since information processing has yet to penetrate far into the more complex and responsible levels of human affairs. 'Complex' and 'responsible' are separate reasons for insisting that a program operate within the human window. Some problems are so difficult that a man–machine intellectual partnership is needed. Others involve life and death, or the manageability of the economy.

One computer program for diagnosing acute abdominal pain, entirely lacking in 'explain' facilities, continues to be used by the doctors involved only through pressure from higher authority. Despite its potentially life-saving power, clinicians cannot feel confident using a black box. True expert systems such as Mycin, however, are capable of giving answers to the question, 'How did you work that out?'

With soft automation, systems are forced at the design stage into the human mental mould. Looking to the future when teams of cooperating robots are at work in our factories, we should ask, 'How should signals be passed between robots? Along wires, by infra-red beams, radio, or some other humanly inaccessible channel?' Synthesized voice would be better, so that human supervisors can keep an ear open for what is going on, as has been shown to be feasible by work at Edinburgh.