PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A
MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

The Role of Experimentation in Artificial Intelligence [and Discussion]

Bruce G. Buchanan, H. Hendriks-Jansen and T. Addis

Phil. Trans. R. Soc. Lond. A 1994 **349**, 153-166
doi: 10.1098/rsta.1994.0120

The role of experimentation in artificial intelligence

By Bruce G. Buchanan

Department of Computer Science, University of Pittsburgh,
Pittsburgh, PA 15260, U.S.A.

Intelligence is a complex, natural phenomenon exhibited by humans and many other living things, without sharply defined boundaries between intelligent and unintelligent behaviour. Artificial inteliigence focuses on the phenomenon of intelligent behaviour, in humans or machines. Experimentation with computer programs allows us to manipulate their design and intervene in the environmental conditions in ways that are not possible with humans. Thus, experimentation can help us to understand what principles govern intelligent action and what mechanisms are sufficient for computers to replicate intelligent behaviours.

1. Introduction

Artificial intelligence (AI) has a long-range scientific goal of understanding the nature of intelligence, merging the concepts of intelligence and mechanization in intelligent machines. Although AI has intellectual roots in antiquity and in rationalist and empiricist philosophies of the 17–19th centuries, it was not until the last half of this century that AI could be studied experimentally. Turing's seminal paper of 1950 proposed an experimental test of intelligence in machines, (more precisely, his operational test was an alternative to arguing the question of whether machines can think) and working AI programs have been demonstrated since 1957 (Feigenbaum & Feldman 1963). Henceforth, experimentation with AI programs has added concreteness and precision to our reflection about the nature of intelligence.

Every scientific discipline focuses on one or more natural phenomena; physics focuses on the nature of matter, astronomy on the origins and composition of the universe, biology on life, and so on. AI focuses on the phenomenon of intelligent behaviour in humans and machines. Because computers are artefacts, Simon refers to AI as a science of the artificial (Simon 1969), but he nevertheless views AI as an empirical science (Newell & Simon 1976).

The term 'artificial intelligence' has been said to be a contradiction in terms (see Boden (1977) for a discussion of this point). However, if one entertains the possibility that computers might be the kinds of things that can be intelligent, then it makes sense to ask how that might be brought about. One of the premises of AI is that artefacts as well as living organisms can exhibit intelligent behaviour. To deny the possibility of intelligence in machines is a definitional prejudice akin to other chauvinistic biases that have limited our vision.

Phil. Trans. R. Soc. Lond. A (1994) **349**, 153–166
Printed in Great Britain

TEX Paper

Table 1. *Characterization of two classes of problems (from Sternberg & Wagner 1993)*

academic problems	practical problems
formulated by someone else	require problem recognition and formulation
well-defined	ill-defined
complete information	require information seeking
single correct answer	multiple acceptable solutions
single method for obtaining answer	multiple paths to solution
little or no intrinsic interest	require motivation and personal involvement
disembedded from ordinary experience	embedded in and require prior everyday experience

2. Intelligence, the subject matter of AI

Intelligence is a complex, natural phenomenon that is manifest without sharply defined boundaries in humans and many other living things. Because the phenomenon of intelligence is not well-defined, Turing sidestepped the issue of defining intelligence *a priori*. Psychologists, on the other hand, have proposed many necessary conditions in the 150 or so different standardized tests of intelligence, and use them to measure and rank people's abilities. For this reason, the first Ph.D. thesis in AI involving a running program (Evans 1968) focused on analogy problems from a standardized college admissions test. Nearly all of the questions on these standardized tests involve narrowly constrained problems carefully insulated from complex interactions with the real world. These are referred to as 'academic' problems as opposed to 'practical' problems (Sternberg & Wagner 1993). The two classes are characterized in table 1.

AI researchers have selected problems that most people would say require some intelligence, usually those more characteristic of academic problems than practical ones. Some examples are analogy finding problems on intelligence tests (Evans 1968), algebra word problems (Bobrow 1968), cryptarithmetic puzzles (Newell & Simon 1972), checkers (Samuel 1959), chess (Berliner 1978), memorizing nonsense syllables (Feigenbaum 1961), block stacking (Winograd 1972; Sussman 1975), logic puzzles like the missionaries and cannibals puzzle (Ernst & Newell 1969), and default reasoning about zoo animals (Winston 1992).

Some of the more practical problems that have been used as vehicles for research are manufacturing assembly by robots (Kak 1990), recognition of spoken requests for information in a database (Woods & Kaplan 1971), scheduling manufacturing operations (Fox & Smith 1984), organic chemical structure elucidation (Lindsay *et al.* 1980), and medical diagnosis (Buchanan & Shortliffe, 1984). (Many commercial applications of AI use the technology but do not contribute directly to AI research. For some of these, see Scott & Klahr 1992, and previous volumes.) However, even these problems do not have the degree of involvement, or 'situatedness', as problems that children and adults solve daily when unexpected events force us to think.

In an empirical study on adults' conceptions of intelligence (Berg & Sternberg

Table 2. *Some of the highest-ranked behaviours in each of three groups of intelligent behaviours (from Berg & Sternberg 1992)*

interest in and ability to deal with novelty	everyday competence	verbal competence
analyses topics in new and original ways	displays good commonsense	displays the knowledge to speak intelligently
interested in learning new things	acts in a mature manner	displays good vocabulary
open-minded to new ideas	acts responsibly	can draw conclusions from information given
can learn and reason with new kinds	interested in family and home life	is verbally fluent
displays curiosity	adjusts to life situations	displays clarity of speech
discovers new ideas	makes rational decisions	

1992), 150 adults in New Haven were asked to list as many behaviours as they could that characterized very intelligent or very unintelligent behaviour. Some additional behaviours and characteristics were added from the psychology literature. Then a separate group was asked to rank these behaviours according to how much intelligence each required, and the resultant rankings were clustered into three groups. Some of the highest-ranked behaviours for each group are shown in table 2.

Most of AI's demonstrable progress is in the category of verbal competence, since that is where symbolic reasoning is placed (under 'drawing conclusions'). However, as the characteristics in this category have become common foci of AI experiments and have become better understood, AI researchers have directed their attention to the other two categories as well. (This is more true of the numbers of researchers than of the chronology of research. For example, McCarthy was writing about common sense reasoning in 1958.) Interestingly, these characteristically intelligent behaviours, especially those in the category of everyday competence, apply more to practical problems than to academic ones. This is not to say that early AI work on academic problems was at all irrelevant, as some have claimed (Dreyfus & Dreyfus 1986). Rather it can be seen as many experiments in controlled situations that were largely decoupled from the open-endedness, randomness, and complexity of the real world. In the same sense, Galileo's measuring the times it took balls to roll down an inclined plane is simple but still relevant to unlocking the secrets of gravitational attraction. With both phenomena, gravity and intelligence, small experiments lay the groundwork for greater understanding. AI, quite clearly, is still in a preliminary stage.

Some of the behavioural characteristics listed in the New Haven survey as unintelligent are shown in table 3. From these lists, we can conclude that computers exhibit, at best, very limited intelligence, in the 20th century. We can also see right away that we do not want our intelligent machines to simulate human reasoning faithfully, if that includes the unintelligent things we all do.

Table 3. *Unintelligent behaviour (from Berg & Sternberg 1992)*

ignorant about current controversial issues
lacks depth of understanding
unable to carry on intelligent conversation
unable to comprehend simple routine tasks
displays illogical thought
does not analyse information
is not interested in gaining knowledge
does not take steps to grow intellectually and learn
does not want to work

3. Experimental methods in AI

Phenomena that occur in the natural world, such as electricity, gravity, combustion, photosynthesis, genetic inheritance, disease, memory, aggression, and many others, have been well studied through scientific observation and experiment. A few of these interesting phenomena can only be studied through their effects because they are not directly observable – as with black holes, geologic change, and sub-atomic particles. Sometimes the phenomena are observable but too large, too remote, or too complex to bring into the laboratory – as with organized crime, solar storms, or global warming. In these cases, the possibilities of active experimentation are small, but passive observation and measurement are still possible.

Sometimes the phenomenon can only be brought into the laboratory in a natural host, without an artefact; as colonies of mice or bacteria are used to study disease in controlled environments. In these cases, experiments are limited to altering the environment and inputs to the system, holding everything else as constant as possible, and observing changes in the outputs. Sometimes, however, it is possible to reproduce them with an artefact in the laboratory and perform controlled experiments, as with electricity. Active experimentation can take place in these cases through altering and controlling many aspects of the experimental apparatus as well as its environment and the input to it.

These features of scientific phenomena are relative, however, and new technologies change our abilities to view phenomena directly and reproduce them. Telescopes and microscopes, for example, enhanced scientists' abilities to observe; electrical generators and the polymerase chain reaction, for example, gave scientists new abilities to experiment. These are summarized in table 4.

An important question for AI is where the phenomenon of intelligence is properly placed in table 4. Before experimental psychologists brought subjects into the laboratory, observation of intelligent behaviour was passive. Later, well-controlled experiments were designed to relate changes in subjects' responses to changes in input conditions.

With computers, we can perform experiments that are not possible with people. Starting with a problem that requires some intelligence by nearly everyone's account (e.g. diagnosing a patient's disease) we design a mechanism that seems plausible. Then we build a device (i.e. a program) that embodies that mechanism,

Table 4. *Two important dimensions of the phenomena studied by empirical science*

	intervention	
manipulation	passive observation (in Nature or laboratory)	active experimentation (usually in the laboratory)
naturally occurring	observe events as they occur e.g. solar storms, organized crime	change environment or input e.g. inherited disease
reproducible in artefacts	create conditions for events to occur, measure them when they do e.g. high energy physics, macro economics	manipulate fine structure, measure responses e.g. electromagnetism

and we run the device under many conditions. The program is the experimental apparatus. It can be configured in any way we please; once configured it can be lobotomized with no ethical misgivings. We can ask it to solve problems all day and all night, we can see how well it performs on impossible tasks. We can add individual facts or methods, and we can cause the subject to forget any of them completely by simply removing them.

It is a fiction, of course, to cast experimental science into a simple procedure. Nevertheless, most or all of the following steps are present in experimental science and can be seen to be present in AI research as well.

1. Start with a question.
2. Formulate a hypothesis.
3. Build a device (in AI, a program).
4. Design and run experiments to test the hypothesis.
5. Redesign the device (or the experiments or the hypothesis).
6. Continue experimenting.
7. Evaluate results of experiments.
8. Generalize the results and generate new questions.

In each experiment, two of the main issues to be addressed are: (i) How well does the program work? (ii) Why does it work as well as it does, i.e., what elements are responsible for its good and bad performance?

Part of the controversy surrounding AI seems to hinge on the extent to which intelligence can be studied in the same ways as other natural phenomena that are reproducible in experimental devices. AI researchers believe it can; critics move the phenomena closer to disease or organized crime.

For example, there is a recent trend in philosophy to treat mind as embedded in a larger environment than the body, to view intelligence as distributed among a brain, a body, and the larger world these interact with. Intelligence, by this view, is in the interaction more than in the brain or in an individual. (For an excellent review of this model, see (Berg & Sternberg 1993), (Vera & Simon 1993), and other articles in the same journal.)

John Haugeland (1993) uses an example of traveling from Berkeley to San Jose to illustrate. Suppose there were a stable of horses in Berkeley in which each horse knew the way to one of the neighbouring cities. Then to get to San Jose all the intelligence Haugland needs is to pick the horse that knows the way to San Jose.

He gets on, rides the horse to San Jose, and he gets off. Much of the intelligence in knowing and following the route lies in the horse. Now substitute freeways for horses. All Haugland has to do is pick the right freeway, get on, drive, and then get off when the exit sign tells him to. Much of the intelligence is in the freeway, he says.

However, this is not incompatible with AI at all. Just as people or insects may derive much of their seeming intelligence from interacting with a complex environment, so may computer systems (Simon 1969). How can it be otherwise in a complex world in which every agent cannot survive without interacting with other agents, with artefacts that other agents build, and with a natural world that no one person can understand fully?

The claim that experimenting with software can reveal anything new about intelligence has been questioned (Kukla 1991). The argument against it, in simplified terms, is that the output of AI programs is completely derivable from the program itself. Since all details of the program are known, the data points from observation and experimentation add nothing to the derivations. AI programs, in this argument, are closer to a set of logical axioms where the only truths to be learned are necessary, but not empirical, truths. We cannot learn new empirical truths by observing a machine whose operations follow necessarily from its program. Therefore, we cannot learn anything new about the empirical phenomenon of intelligence by studying software.

Without taking the whole argument into account here, the best succinct response is with an analogy to experimentation in physics. Under a deterministic view of the physical world, all observed data points of physics would also be derivable from applying the laws of the universe to the initial conditions. Physicists still construct devices to test hypotheses, however. When the devices work, the resulting data confirm the hypotheses. When data from the devices contradict the hypothesis, modifications are made; either to the hypothesis or to the device. The same is true with software devices.

Another way to avoid this dilemma is to see that, even though the output of a program may be completely determined by the input and the complete description of the program, the interesting question still remains of whether that output is an intelligent response in that situation. By collecting data on many responses we are in a position to say whether the program generally responds intelligently. For example, when MYCIN reached a correct diagnosis of streptococcal meningitis for a patient, we claimed it was behaving intelligently. When it made an incorrect diagnosis, we asked whether it was behaving stupidly or whether it was still a plausible, intelligent response, even though wrong. From many data points we confirmed that the program could diagnose causes of bacterial meningitis and construct therapy plans as well as experienced infectious disease experts (Buchanan & Shortliffe 1984).

However, we wanted to generalize further and explain why it performed as well as it did and what general architectural features others could use to design similarly intelligent programs. The big question for an AI program is not so much what its output will be. It is whether that output constitutes an intelligent response and, if it does, which parts and which interactions are responsible for the intelligence in the response. As might be expected, with MYCIN there was not a simple answer. A few of the components of our answer, though, can be stated simply.

(*a*) The detailed knowledge of meningitis, contained in about 500 conditional rules in MYCIN's knowledge base, was essential.

(*b*) An important ingredient of success was the fact that the number of diagnostic hypotheses was fixed and small.

(*c*) Another important ingredient was MYCIN's evidence gathering method. For example, MYCIN declined to offer a diagnosis when there was insufficient evidence for any alternative, rather than suggest the best of several bad choices.

(*d*) The details of the calculus for propagating uncertainty through reasoning chains were less important than the fact that uncertainty was managed explicitly.

(*e*) Little more logical machinery than modus ponens was necessary for good performance.

(*f*) Some very particular rules covered only rare exceptions but were necessary to avoid stupid mistakes.

(*g*) Explicit strategies made MYCIN's reasoning more understandable but could be compiled into the rule set without loss of accuracy.

Principles like these were synthesized from an analysis of many variations of MYCIN. They were then embedded in a high-level 'shell' system that could be instantiated with knowledge of other problem areas and would behave in the same general way as MYCIN. This system, called EMYCIN (VanMelle 1980), became the basis for many commercial expert systems which further confirmed the general principles.

Each program is a device that can be used for multiple experiments. By varying the inputs systematically we can observe and analyse changes to the outputs in order to understand the strengths and limits of the architectural principles governing each program's behaviour. When the internal details of the program are also modified, however, the experiments become finer-grained (Buchanan 1988). Insofar as a program exhibits any intelligence at all, both the large-scale and the fine-grained experiments can help us discover its causal and inhibitive factors.

Although there are relatively few sets of systematic experiments described in the AI literature, there are numerous examples of rather unsystematic variation and experimentation with methods and architectural principles resulting in general lessons of interest to the research community. Some of the results have also been successfully transferred to the worlds of commerce, manufacturing, military, and health (see, for example, Scott & Klahr 1992, and other volumes in this series). The set of experiments with the MYCIN program (Buchanan & Shortliffe 1984), guided the design and implementation of many rule-based expert systems and, we believe, elucidated some of the strengths and limits of rule-based architectures as computational models of intelligent decision making.

AI needs more systematic studies, data collection, and analysis leading to refinements in our hypotheses about the power, generality, and scope of applicability of our methods.

4. An example from machine learning

Machine learning has been an active research area for over 40 years and is often said to be an essential ingredient of machine intelligence (Minsky 1963). Its origins can be found in adaptive control theory (Wiener 1948). It has been,

Table 5. *Characteristics of the data in most practical problems*

characteristic	example
incomplete	not all relevant subclasses are represented; it is not even known what features are relevant
redundant	several features are manifestations of the same underlying phenomenon
noisy	laboratory measurements may be perturbed by systematic or random errors during data collection
erroneous	laboratory tests are not fully reproducible

and continues to be, addressed both theoretically and experimentally (Langley 1988). The work in our laboratory is experimental in two senses. First, we are testing the hypothesis that heuristic search is a powerful enough architecture for an inductive learning program that new and interesting generalizations can be found using it. Second, we are using an existing inductive learning program to experiment with datasets provided by different collaborators to help them find interesting classification rules. We usually start with a problem posed by a collaborator of the form: ‘Are there general rules that define membership in class X?’

For example, we† are currently investigating the question, ‘Can we learn a set of classification rules (a concept definition) for carcinogenic chemicals?’ (We are first looking at carcinogenic activity in rodents because there are more data available than for humans.) Not only is this an important public health question, but it challenges existing machine learning programs because there are so few chemicals with a full set of test values available, the features overlap and are not independent, and there are errors in the data (summarized in table 5).

Our working hypothesis is that the RL learning program (Provost *et al.* 1993) developed in our laboratory, is sufficient for learning with these data. A little more generally, the hypothesis is that the heuristic search architecture of RL is sufficient for this and many such problems of inductive generalization. RL is an induction program in that it uses the features of particular objects – individual chemicals in this case – to find general relations that define a target concept class, in this case a definition of ‘carcinogen in rodents’ or ‘non-carcinogen in rodents’. It may be briefly described as a heuristic search program that searches a space of rules to find a definition of the concept class, where its search is guided by prior knowledge as well as by the data. The overall data flow is shown in figure 1.

Using a generator of syntactically possible rules that define a total rule space, RL explores the space heuristically. It starts from rules with a single feature and adds one feature at a time to partial rules that look most promising. Each

† ‘We’ includes Mr Yongwon Lee, and Dr Foster Provost, Dr Rich Ambrosino, and Dr John Aronis of the Intelligent Systems Laboratory, in collaboration with Dr Herb Rosenkranz and members of his laboratory in the Department of Environmental and Occupational Health. The preliminary results shown here were generated by Yongwon Lee.

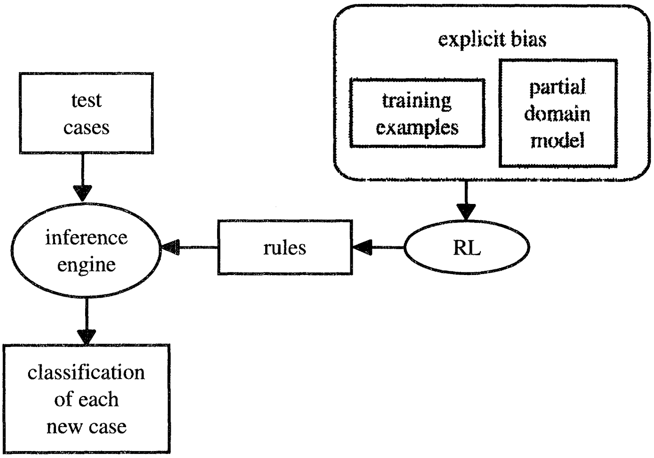

Figure 1. The overall flow of information for RL. RL uses a copy of the inference engine to make predictions for training examples and compares the prediction against the known classifications. The term ‘bias’ in an induction system refers to the choices made in designing, implementing and initializing an inductive learning system that lead the system to learn one generalization instead of another. The partial domain model is knowledge about the domain that RL can take as given which will help guide its search through the rule space.

partial rule is matched against the training cases to see how many of the positive cases are correctly predicted and how many of the negative cases are incorrectly predicted to be members of the target class, thus providing statistical guidance to the search. In addition, semantic relations among features that are specified in the partial domain theory are taken into account in exploring the rule space.

RL is one of several AI machine learning programs that learn inductively. (For an overview of recent work, see Michalski & Tecuci 1994.) Although it shares several characteristics with one or more other programs, it is distinguished by its flexibility, as summarized in table 6.

Working in collaboration with a toxicologist, we designed a set of experiments to test our hypothesis that RL was sufficient to learn rules in this domain. Note that we were also testing a hypothesis that the descriptive features from relatively inexpensive tests we were using to characterize the chemicals were a sufficient set to predict carcinogenic activity as well as the very expensive tests now used. One of the reasons this problem is important is that no such set is known at present. Thus any learning program used to learn rules must be flexible enough to accept different sets of descriptive features easily. One of the hypotheses we are testing is that the RL framework is flexible enough to allow many changes to the feature set, parameters, and assumptions that drive induction.

One question we are investigating, for example, is how an inference engine can most effectively use multiple predictions from a set of rules. If a new chemical is described as having features A1–A10, then what is the correct interpretation of the following rule set,

R1 A1 & A2 → carcinogenic (0.5),
R2 A2 & A3 → carcinogenic (0.8),
R3 A9 & A10 → not carcinogenic (0.8),

Table 6. *Characteristics of the RL induction program*

a space of possible rules is defined by a syntactic generator of rules
the search for plausible rules in this space is guided by prior knowledge (in a partial domain theory) as well as by statistics
objects may be described using symbolic features as well as numeric features
data are not assumed to be 100% reliable
the definition of the target class may be nonlinear
the system is modular and flexible

where the numbers associated with the rules are measures of evidential support determined by RL during learning.† Many sets of rules were learned under different conditions. Each set was applied to test data using different methods of combining evidence. Preliminary results are shown in table 7. From this small set of experiments we concluded that a simple voting scheme would be most effective in this domain. We have run many such experiments and have modified the assumptions and parameters that drive RL in this domain. We have also redesigned RL to broaden its scope. It was originally designed as a simple induction system to be run once (or a small number of times) to find classification rules. In the domain of chemical toxicity, as well as others, we found that one of the primary values of an induction tool is to assist scientists in exploring their own assumptions and their hypotheses about appropriate features to use in characterizing the data. Thus RL is becoming a general tool for exploration of data that are not yet well understood.

These explorations are not finished by any means. However, preliminary results (Lee *et al.* 1994) are encouraging for both toxicology and machine learning. We have shown, for example, that it is possible to improve the predictive power of the short-term assay that has been proposed as best, by including two other short-term assays, two toxicity dose measurements, and four physical properties, all of which are obtainable without great cost. One of the 25 rules learned is shown in table 8.

From the perspective of machine learning, we are demonstrating that RL can find (and represent) nonlinear relationships in data. We are also demonstrating (although it does not show here) that RL is flexible enough to undertake dozens of investigations around the same topic but with very different assumptions.

5. Conclusion

In AI we are looking at many different aspects of intelligence and building experimental programs that exhibit those aspects. We are still confined to small domains and to small tasks within those domains, but we believe we are moving toward greater understanding of how programs can behave intelligently in

† Evidential support is calculated as the simple ratio of the number of positive predictions divided by the total number of predictions. This, too, is open to experimentation.

Table 7. *Predictive accuracy of learned rules with different evidence gathering methods*
(Abbreviated results from experiments reported in Lee *et al.* (1994).)

test set	simple voting	weighted voting	single best rule
1	65.2	63.5	62.2
2	64.6	63.9	62.3
3	64.4	63.7	62.2
4	62.5	61.2	57.1
5	61.5	61.5	58.7
6	58.5	59.3	55.7
7	61.9	62.7	53.2
8	62.4	60.4	55.2
9	60.6	62.8	56.5
10	63.1	60.5	55.2
average	62.5	61.9	57.8

Table 8. *A sample rule from RL's examination of data on the classification of chemicals as carcinogenic or non-carcinogenic in rodents, using the physical properties and results of short-term assays as features*

(The best rule proposed in the literature – that a chemical is carcinogenic in rodents if and only if the Salmonella genotoxicity assay is positive – results in 55.7% predictive accuracy. On a test set of 15 chemicals for which the Salmonella assay is negative, the 25 learned rules have an overall predictive accuracy of 80%. Taken from Lee *et al.* (1994).)

IF: the Salmonella genotoxicity assay is negative
AND the chromosomal aberrations assay is positive
AND the log of water-octanol partition coefficient is greater than 3.3
THEN: the chemical is carcinogenic in rodents

True positive rate	=	29.0% (20/69)
False positive rate	=	3.0% (2/66)
Positive predictive value	=	90.9% (20/22)

increasingly more complex situations. Critics argue that intelligence lies only in the whole context of living successfully in the world; that it is nonsense to separate intelligence on academic problems from practical intelligence (e.g. (Dreyfus & Dreyfus 1986)). Most of us in AI, on the other hand, take it as an empirical question whether the mechanisms that are designed and analysed one at a time can be combined into agents that are more than the sum of their parts (e.g. (Newell & Simon 1976)).

The enterprise of understanding the nature of intelligence can seem overwhelmingly complicated unless we continue to work incrementally, and have conceptually simple models guiding us. In the early days of wireless radio, Einstein was asked to explain how it worked. He said we should first think about sending telegrams over wires; that's like having a large cat stretching over a large area so that when you pull its tail in New York its head meows in Los Angeles. 'And radio operates exactly the same way', he said, 'The only difference is that there is no cat.' (Quoted in Laquey 1993.) AI is still in its early days. However, we are past the very early stage where the best analogy about AI was that AI in computers is just like human intelligence, but without the human. Progress is the result of better methods and better understanding; these result from experimentation.

I am particularly grateful to Yongwon Lee, Foster Provost and John Aronis for carrying out the experiments with RL reported here. I also thank Alan Bundy (University of Edinburgh), and Diana Forsythe (University of Pittsburgh) for pointing out particularly opaque parts of earlier drafts, and E.A. Feigenbaum (Stanford University) and members of the Intelligent Systems Laboratory (University of Pittsburgh) for many interesting and provocative discussions on the topics discussed here. This work was supported in part by the National Library of Medicine and by the W. M. Keck Foundation.

References

Berg, C. A. & Sternberg, R. J. 1992 Adults' conception of intelligence across the adult life span. *Psychol. Aging* **7**, 221–231.

Berg, C. A. & Sternberg, R. J. 1993 Cognition in the head and in the world: An introduction to the Special Issue on situated action. *Cognitive Science* **17**, 1–6.

Berliner, H. J. 1978 A chronology of computer chess and its literature. *Artificial Intelligence* **10**, 201–214.

Bobrow, D. G. 1968 Natural language input for a computer problem-solving system. In *Semantic information processing* (ed. M. Minsky), pp. 146–226. Cambridge, MA: The MIT Press.

Boden, M. 1977 *Artificial intelligence and natural man.* New York: Basic Books.

Buchanan, B. G. 1988 Artificial intelligence as an experimental science. In *Aspects of artificial intelligence* (ed. J. H. Fetzer), pp. 209–250. Boston: Kluwer Academic Publishers.

Buchanan, B. G. & Shortliffe, E. H. 1984 *Rule-based expert systems: the* MYCIN *experiments of the Stanford Heuristic Programming Project.* Reading, MA: Addison-Wesley.

Dreyfus, H. & Dreyfus, S. 1986 Why expert systems do not exhibit expertise. *IEEE Expert* **1**, 86–90.

Ernst, G. W. & Newell, A. 1969 *GPS: a case study in generality and problem solving.* New York: Academic Press.

Evans, T. G. 1968 A program for the solution of geometric-analogy intelligence test questions. In *Semantic information processing* (ed. M. Minsky), pp. 271–353. Cambridge, MA: The MIT Press.

Feigenbaum, E. A. 1961 The simulation of verbal learning behavior. In *Proc. Western Joint Computer Conf.*, pp. 121–132. (Reprinted in Feigenbaum & Feldman 1963.)

Feigenbaum, E. A. & Feldman, J. (eds.) 1963 *Computers and thought.* New York: McGraw-Hill.

Fox, M. S. & Smith, S. F. 1984 ISIS: A knowledge-based system for factory scheduling. *Expert Systems* **1**, 25–49.

Haugeland, J. 1993 Mind embedded in the world. In *Colloquium at the Center for Philosophy of Science.* University of Pittsburgh.

Kak, A. 1990 Robotic assembly and task planning. *AI Mag.* **11**, Guest Editorial for Special Issue.

Kukla, A. 1991 *Medium AI and experimental science.* Computing and Philosophy.

Langley, P. 1988 Machine learning as an experimental science. *Machine Learning* **3**, 5–8.

Laquey, T. 1993 *The Internet companion.* Reading, MA: Addison-Wesley.

Lee, Y., Rosenkranz, H. S., Buchanan, B. G. & Mattison, D. M. 1994 Learning rules to predict chemical carcinogenesis in rodents. Tech. rept. ISL-94-25. Intelligent Systems Laboratory, University of Pittsburgh.

Lindsay, R. K., Buchanan, B. G., Feigenbaum, E. A. & Lederberg, J. 1980 *Applications of artificial intelligence for organic chemistry: The DENDRAL project*. New York: McGraw-Hill.

McCarthy, J. 1958 *Programs with common sense*, pp. 77–84. London: Her Majesty's Stationery Office. (Reprinted in Minsky 1968.)

Michalski, R. S. & Tecuci, G. (eds.) 1994 *Machine learning: a multistrategy approach, vol. IV*. San Francisco: Morgan Kaufmann.

Minsky, M. 1963 Steps toward artificial intelligence. In *Computers and thought*, pp. 406–450. New York: McGraw-Hill.

Minsky, M. (ed) 1968 *Semantic information processing*. Cambridge, MA: The MIT Press.

Newell, A. & Simon, H. A. 1972 *Human problem solving*. Englewood Cliffs, N.J.: Prentice-Hall.

Newell, A. & Simon, H. A. 1976 Computer science as empirical inquiry: Symbols and search. *Commun. ACM* **19**, 113–126.

Provost, F. J., Buchanan, B. G., Clearwater, S. H. & Lee, Y. 1993 Machine learning in the service of exploratory science and engineering: a case study of the RL induction program. Tech. rept. ISL-93-6. Intelligent Systems Laboratory, University of Pittsburgh.

Samuel, A. L. 1959 Some studies in machine learning using the game of checkers. *IBM J. Res. Development* **3**, 211–229. (Reprinted in Feigenbaum & Feldman 1963.)

Scott, A. C. & Klahr, P. 1992 *Innovative applications of artificial intelligence 4*. Menlo Park: AAAI Press.

Simon, H. A. 1969 *Sciences of the artificial*. Cambridge, MA: MIT Press.

Sternberg, R. J. & Wagner, R. K. 1993 The g-ocentric view of intelligence and job performance is wrong. *Current Directions Psychol. Science* **2**, 1–5.

Sussman, G. J. 1975 *A computer model of skill acquisition*. New York: American Elsevier.

Turing, A. M. 1950 Computing machinery and intelligence. *Mind*, October, 433–450.

VanMelle, W. 1980 A domain-independent system that aids in constructing knowledge-based consultation programs. Ph.D. thesis, Computer Science Department, Stanford University, U.S.A.

Vera, A. H. & Simon, H. A. 1993 Situated action: a symbolic interpretation. *Cognitive Science* **17**, 7–48.

Wiener, N. 1948 *Cybernetics*. New York: Wiley.

Winograd, T. 1972 *Understanding natural language*. New York: Academic Press.

Winston, P. H. 1992 *Artificial intelligence*, 3rd edn. Reading, MA: Addison-Wesley.

Woods, W. A. & Kaplan. R. M. 1971 The lunar sciences natural language information system. Tech. rept. 2265. Bolt, Beranek and Newman, Cambridge, MA.

Discussion

H. Hendriks-Jansen (*University of Sussex, U.K.*). Isn't Professor Buchanan worried that the choice of representations is imposed by the programmer, not developed by the demands of the problem? If AI systems were embodied, the demands of sensor-motor coordination would deeply affect how the system represented things.

B. G. Buchanan. I agree that the representations might be different in that case. But I think that's irrelevant to the applications I mentioned.

T. Addis (*University of Reading, U.K.*). A key problem in classical AI is how to define the distinctions used to describe the world. Robots interacting with the world discover their own distinctions. Classical AI works with pre-defined descriptions.

B. G. Buchanan. The AI systems I've described are purely cognitive. They interact with a sub-set of the world, but in a rich way. I disagree that robotics is the only way we can go. Despite the limitation you mention, AI-methods can be useful. But there is also considerable work within the classical paradigm on automatically extending the set of features given to a program, much of it under the name "constructive induction".