Landgarten H B 1980 *Clinical Art Therapy*. Brunner/Mazel, New York
Richman J 1977 Art therapy and group process. *Art Psychother*. 4: 5–9
Robbins A, Sibley L B 1976 *Creative Art Therapy*. Brunner/Mazel, New York

R. M. Thomas

Artificial Intelligence

Before defining what artificial intelligence is, a brief discussion on intelligence itself is needed. The concept of intelligence has captivated and confounded educators, philosophers, psychologists, theologians, and neurophysiologists alike throughout the ages. Despite the profusion of words generated on the topic, researchers have only succeeded in concluding that, "intelligence is what intelligence tests measure." Although a succinct definition is unavailable, there are a number of behaviors that can be classified as "intelligent." For instance, understanding language, employing inductive, deductive, and commonsense reasoning, solving mathematics problems and puzzles, and even planning shopping trips all demand intelligence. Traditionally, these actions have been restricted to humans. However, since the early 1950s, electronic computer systems have also succeeded in performing these same tasks. Systems such as these, then, possess "artificial intelligence."

1. The History of Artificial Intelligence

Throughout history, humans have attempted to imitate themselves, producing objects that are modeled after humans, yet which possess characteristics above and beyond human capabilities. This preoccupation may be traced back as far as early cave drawings where people were depicted in heroic battles with wild beasts. The Greeks also had a propensity to invent "superhumans," or artificial intelligences, that served as the gods in their mythology. As such, these artificial intelligences behaved as humans in some contexts, but ultimately were capable of feats beyond mere human powers.

Automata, or self-locomoting contrivances, appeared around 200 BC in Hellenic Egypt (McCorduck 1979). Some of these mechanisms prophesied and gestured to awestruck audiences by means of quicksilver, hydraulics, or pulleys and strings. Technically these devices cannot be classified as true artificial intelligences since the source of their intelligent behaviors resided in sources extrinsic to their structure, that is, in the hands of priests or actors.

The sixteenth century saw a profusion of automata. These mechanisms were not "thinking machines" but clever combinations of gears, fabric, and imagination. One such smart automaton, the chess-playing Turk in the early 1800s, travelled the world and amazed audiences with its chess expertise. To the dismay of the many who wanted to believe, the Turk was subsequently proven fraudulent since it was operated by a small person who sat in the box that allegedly housed the chess-playing machinery. However, the quest persisted for a truly intelligent, self-contained machine.

The conception of the first multipurpose computer may be tentatively attributed to Charles Babbage, a mathematics student at Cambridge in the early part of the nineteenth century. His first dream was to develop an automatic means of calculating logarithm tables. He succeeded in 1822 with the "difference engine," a small model of the original conception. Meanwhile, his dreams embraced the notion of creating an all-purpose machine which he called the "analytic engine." Due to the constraints of the technology at the time, he was never able to make this dream come true.

With the advent of general-purpose digital computers, the stage was set for artificial intelligence. The originator of digital computers, or at least of the initial specifications for them, was Norbert Weiner (1894–1964). Essentially, his specifications included "a numerical central processor whose mechanism would be electronic and not mechanical, based on a binary rather than a decimal system; a machine with built in abilities to make logical decisions, and an apparatus for easy storage and manipulation of data" (McCorduck 1979 p. 44). John von Neumann played a key role in the development of the digital computer.

Alan Turing (1912–1954) provided the first major link between modern computing systems and thinking. He was a British mathematician/logician who envisioned a computing system capable of not only "number crunching" but symbolic manipulation as well. What he proposed was the possibility of an abstract, universal computing device that is today called a Turing machine. In addition, he developed the "imitation test," now known as the "Turing test," which is a means of determining a machine's "intelligence." The Turing test consists of "an interrogator communicating via teleprinters with a human and a computer. The interrogator can attempt in any way possible to determine which is which through conversation over the communication links" (Roberts 1981). This test stands today as the ultimate challenge in artificial intelligence.

Dartmouth College, New Hampshire, in 1956 was the site of an embryonic formalization of artificial intelligence. The occasion was a conference bringing together a small group of people from diverse backgrounds and places. It was here that the term "artificial intelligence" was coined by John McCarthy. The common thread linking these people was their belief in the digital computer's potential to "think." John McCarthy, Marvin Minsky, Nathaniel Rochester, and Claude Shannon were the organizational members of this group. Some of the others attending this momentous conference included: Trenchard More, Arthur Samuel, Oliver Selfridge, Ray Solomonoff, Allen Newell, and Herbert A. Simon. This meeting began charting the course for the field of artificial intelligence in the future.

In this New Hampshire setting, the group met and exchanged ideas and research for two months. Probably the research that most adequately demonstrated machine intelligence at this conference was the work done by Newell, Simon, and their colleague J. C. Shaw. They had implemented a list-processing language at the Rand Corporation and had succeeded in creating a program, the Logic Theorist, which could prove theorems in Whitehead and Russell's *Principa Mathematica*. Moreover, they were developing their General Problem Solver (GPS) program which, by way of forward reasoning, could solve general problems.

The GPS employed a "means–end analysis" where the goal was to make the present state of the problem and the desired state the same. This was achieved by generating subgoals that progressively reduced the difference between the goal and the present state. A hierarchy of goals and subgoals can be used to guide the problem solver to a point where a legal transformation may be applied to achieve the final subgoal. For example, consider the problem of transferring A to B. The process is a depth-first search, and the goals stack hierarchically as: (a) transfer A to B, (b) reduce the difference between A and B by modifying A, (c) apply operator Q (legal transformation) to object A.

The goal addresses changing the current situation to the desired state and the rules define which transformations or operations are legal. This heuristic may be applied to the solution of a relatively simple problem like "Cannibals and Missionaries," where three cannibals and three missionaries have to cross a river in one boat. The cannibals cannot outnumber the missionaries, though, on either side of the river, or the missionaries will be eaten. The heuristic may also be applied to a more complex domain like chess with the goal being the capture of the opponent's king. The means–end analysis is a powerful tool when the domain has limited knowledge, but problems arise with more complex problem spaces. The GPS represented the first successful, artificial intelligence system.

The research developments subsequent to the Dartmouth conference lay beyond the imagination of those present in 1956. Artificial intelligence has since infiltrated many scientific fields and branched out into many subspecialty areas with seemingly limitless possibilities of exploration. Before some of the diverse applications of artificial intelligence today are discussed, some basic concepts must first be delineated.

2. *Basic Concepts and Theories*

The original purpose of the computer was to perform mathematical operations very quickly, eliminating the drudgery of calculations and possible human error. Everything in a computer is represented as a string of binary digits, zeros and ones, called "bits." Typically, the bits are interpreted as a code for decimal digits; however, with a slight extension of this principle, they can also be grouped and interpreted as characters. Once this is done, word and sentence interpretation is a fait accompli, and more complex groupings of words become possible. This process is called "symbol manipulation" and is essential to developing programs that show intelligent behavior.

Of the many programming languages available today, two are currently favored by artificial intelligence researchers: LISP (for list processing) and PROLOG (for programming logic). Both languages are designed for symbolic manipulation and each has its adherents. Since they are higher level languages, programming is less cryptic than with older machine languages. Furthermore, the newer languages represent an attempt to achieve some global modularity within the programming context and several things intrinsic to these systems (e.g., pattern matching and general search heuristics) make them more understandable, modifiable, and hence, more "user friendly."

With the tools now available for working with symbols (words), some conceptual issues of how people think must be addressed, particularly issues of types of knowledge and production rules. These are of crucial importance if the purpose of artificial intelligence is, in fact, to model human thinking, or in the case of "expert systems," to surpass it.

2.1 Types of Knowledge

Knowledge that a person has about the world can be divided into two major categories. First, "declarative knowledge" corresponds to factual knowledge about the world, similar to textbook knowledge. Such a declarative representation would be: "Canberra is the capital of Australia." Knowledge of this type has been included in computer systems for years. The other knowledge type posited is of a "procedural" nature; that is, knowledge of how to do something. Heuristics, judgment rules, general inference procedures, and so on illustrate components of procedural knowledge. Typifying this type of knowledge would be knowing how to multiply two numbers together. Programming this knowledge is somewhat more difficult than for declarative knowledge since much of what people know how to do has become so automated that explication—step by step—is sometimes nearly impossible. For instance, imagine trying to teach someone how to ride a bicycle with only verbal instructions. Since the media of communication with a computer is by way of words, the procedural decomposition into discrete units with all units contributing to a coherent and complete representation is required. Task analysis (Resnick 1976) in cognitive psychology does just that, systematically breaking down a cognitive task (e.g., solving a geometry problem) into its component parts.

2.2 Production Rules

"If–then" statements, or condition–action rules, are called "production rules" and allow the system to execute the appropriate actions given the satisfaction (matching) of a particular condition or set of conditions. Once an area has been decomposed, procedural knowledge may be represented, allowing for the representation of individual chunks of knowledge. Also, this type of rule representation is more convenient for programming inductive learning where, by progressive extension and hierarchical structuring, incremental building up of the knowledge base occurs. Variables may replace constants at the higher levels, allowing for generalizations to be made as well as organization of data to be effected.

As more complex domains are encountered and represented, it becomes increasingly important to deduce the effects of actions from axioms about the world rather than representing each effect explicitly. Deductive rules can be programmed via "if–then" relations; and if the precondition of a deductive operator holds, the effects may be added to the database. This is an example of an adaptive learning system which is one that can modify its own production rules (Waterman 1970). Basically, there are three ways of achieving this: adding new rules, deleting old rules, or changing existing rules. This corresponds to the human experience of learning.

Learning, then, is much more than simply adding new facts to a database. It involves relating something new to what is already known in a complex, coherent manner. Therefore, both types of knowledge, declarative and procedural, must be represented and utilized for a machine to possess a measure of intelligence. Ideally, the system should also include "executive functions," or some form of self-awareness. Brown (1978) has proposed six components of an executive control system. These include the ability to: (a) predict the system's capacity limitations, (b) be aware of its repertoire of heuristic routines and their appropriate domain of utility, (c) identify and characterize the problem at hand, (d) plan and schedule appropriate problem-solving strategies, (e) monitor and supervise the effectiveness of those routines it calls into service, and (f) dynamically evaluate these operations in the face of success or failure so that termination of strategic activities can be better timed.

Implementing all of these functions into an operating computer system is not a far-off goal of the future but a reality of today in many artificial intelligence programs.

3. Applications of Artificial Intelligence

Humans differ from computers in a variety of ways. One distinction is in the use of logic in problem solving or decision making. Whereas computers engage in strict logical reasoning, humans often display "semilogical" reasoning (Duda and Gaschnig 1981) which incorporates intuitions, prior experiences, pet theories, and other things into a problem-solving task. One branch of artificial intelligence specifically attends to the modeling of human behavior. This is called "computer simulation" and serves the function of confirming psychological theories or models. Another important difference is that humans have capacity limitations while computers have storage capacities well beyond human abilities. Any occupation that requires large, interconnected knowledge bases with difficult decision-making tasks (e.g., medical diagnosis) could be done also by computers. The first artificial intelligence application discussed is that known as "expert systems."

3.1 Knowledge-based Expert Systems

The first thing to be done in creating a computer expert system is to engage in intensive/extensive interviews with human "experts" in a particular field. The purpose of this task is to amass the knowledge necessary for the knowledge base as well as to capture some of the human, "semilogical" elements for rendering judgments. For example, one system called MYCIN (Shortliffe 1976) is an "expert" in diagnosing bacterial infections. The way it works is based on the "if–then" rules discussed earlier. In addition, weights or "certainty factors" are attached, reflecting the degree to which the system believes in the correctness of an hypothesis, given the evidence presented. These values range from +1 to −1 where greater values indicate hypothesis validity and values closer to −1 indicate that the hypothesis is probably false. A zero value may be interpreted as insufficient evidence. An illustration of a MYCIN rule would be:

> If: (a) the stain of the organism is grampos, and
> (b) the morphology of the organism is coccus, and
> (c) the growth conformation of the organism is chains,
> Then: There is suggestive evidence (0.7) that the identity of the organism is streptococcus.

MYCIN requires specific information about the patient. Therefore, it asks questions (displayed on the computer terminal) that will aid in diagnosing the problem. On the basis of the answers supplied by the user (usually the attending physician), the task now involves a four-stage decision problem. First, the computer must decide which organisms, if any, are causing significant disease. Second, it must determine the likely identity of the significant organisms. Next, the system must decide which drugs are potentially useful. Finally, a selection must be made of the best drugs for the patient based on the diagnosis.

The organization of the hundreds of rules (almost 500 in the original system) is in the form of rule networks. Furthermore, a separation exists between (a) rules forming the knowledge base from (b) the information about the current problem, and (c) the methods for relating the general knowledge to the problem at hand by way of the rule interpreter. This separation of functions allows for modifying the knowledge base without disturbing the other programs (current database or rule interpreter). Also this type of system maintains an ongoing record of its reasoning path, capable of describing the rules it selected and why. Therefore, it may be employed as a tutorial/consultant device.

Another expert system currently in use is PROSPECTOR, a system for mineral exploration. This rule-based model effects a relation between the geologist's field evidence and relevant geological hypotheses (Duda and Gaschnig 1981). PROSPECTOR can evaluate the likelihood of an area's containing a particular ore, as well as provide an explanation as to why the site should or should not be selected for drilling. DENDRAL, developed at Stanford University (Feigenbaum et al. 1971), engages in a heuristic search of chemicals to determine reasonable structural representations of organic molecules from mass-spectrogram data, nuclear–magnetic–resonance data, and additional information provided by the user. This system has currently succeeded in surpassing chemical experts in speed and accuracy of judging chemical compounds for the molecular families covered by its rules.

Expert systems exist that can prove theorems in mathematics, plan the construction of robots, or the configuration of computers (DEC's VAX systems), tutor students in a given domain (via drill and practice, learning games, or discovery learning), diagnose heart diseases, and much more. Any domain which can be described by rules is a potential candidate for expert system representation. However, there must be at least one human expert in the field both to input knowledge and to corroborate the computer's output for any domain. This forces people to formalize their thinking by making concrete and explicit what may have formerly been based on subjective interpretations or "feelings."

3.2 Natural Language Processing

When two people communicate with each other using language, they effortlessly use complex and, as yet, little understood processes. What transpires is a communication in much more than just the lexical meanings of exchanged words. Languages that are used as the principal means of communicating in daily affairs of humans are called "natural languages." These contrast with other, more formal languages which have been invented by people for particular kinds of communication and are called "artificial languages." Examples of artificial languages include predicate calculus, LISP, and even musical notation.

It has been very difficult to develop computer systems capable of generating and "understanding" even parts of a natural language. This is due in part to the difficulty in programming all of the necessary contexts and experiences two humans have in common when they converse. Much of what becomes communicated is left implicit; that is, the particular context the persons are sharing allows them to relate on a mutually shared, more abstract level (streamlined messages), leaving many things as "understood."

In order for a computer to understand natural language, a complete delineation of the present environment (i.e., the context) would need to be input, as well as a means for the computer to make inferences from the statements. To further confound issues, individuals communicate by use of analogies, humor, and even "body language" which makes the programming task seemingly impossible.

Despite these problems, systems have been developed that understand spoken and written fragments of language. Some of the earlier programs, like ELIZA (Weizenbaum 1976), used pattern matching of keywords to elicit a preprogrammed response. The semantic content of the sentence was ignored, with only the structuring elements (keywords) utilized to effect the response. This system was based on Rogerian, nondirected psychotherapy in which a "client" would type into the computer a problem, such as, "I've been having problems with my mother," and the computer, focusing on the keywords "problems" and "mother" would respond, "Please tell me more about your mother," and so on.

These early systems were comparatively unsophisticated programs with preset semantic knowledge. In the mid-1960s and early 1970s real-world knowledge was being incorporated into programs with inference rules and semantics, allowing for a fuller "understanding" of the text by the computer.

One such system that dealt with natural language processing was Bobrow's STUDENT program (1962) which solved elementary algebra word problems directly as they were stated in English in high-school mathematics books. The system's database contained some general knowledge about the world, such as: three feet equals one yard, and distance equals velocity × time. An example of a problem that STUDENT could solve would be: "If the number of customers Tom gets is twice the square of 20 percent of the number of advertisements he runs, and the number of advertisements he runs is 45, what is the number of customers Tom gets?" STUDENT's solution to this problem is not based on any conventional linguistic analysis method. Rather, the focus is on its known goal: to translate the input problem into a set of simultaneous algebraic equations, which could then be given to a subroutine for equation solving. Words or phrases were "translated" into arithmetic operators, constants, or variables and then solved mathematically.

The STUDENT program represents a special purpose system and, as such, is restricted to its particular domain of algebra word problems. Other systems have been designed from the start as experiments in the analysis and representation of general knowledge.

Quillian (1968) at Carnegie-Mellon University built one of the first computer representations of general knowledge based on semantic classifications. Each word in the system was defined in the computer's memory by a network of labelled links to other words. This "semantic memory" included such link relations as class inclusion, size, color, logical relations to other words, and so forth. In order to compare two words, the computer would first need to access the words, then describe all paths existing between the two words.

Present systems have evolved to the point where question answering is possible, accepting simple English queries which specify what information the user needs, then generating fairly complex programs specifying "how" the computer should retrieve the information. LADDER, developed at SRI International, has these capabilities with a large database that includes over 100 fields (Naval issues) in 14 files, as well as records on over 40,000 ships.

Two problems had to be overcome for successful operation of this system. First, the system required a translation from English into a formal language and, second, the system had to convert that statement of what was requested into a statement of how to get it. This second problem addresses the issue of "automatic programming" and may be understood as a "supercompiler" within the system that generates a program in response to interpreted demands.

LADDER accepts input in the form of specific questions on ships, and it supplies appropriate answers. This seems simple enough on the surface, but the interpretive processes that go on inside the machine are complex. There is a dictionary for making spelling corrections, interpretations of partial sentences ("ellipses") based on preceding statements (context), a process for understanding pronoun usage and some colloquialisms, and more. A sample of LADDER's communication follows (Hendrix and Sacerdoti 1981 p.314):

1. Give me the length of the Kenedy.
   Spelling → KENNEDY
   PARSED!
   Give the length of the ship KENNEDY JF
   LEN = 1072 (feet)
2. Width and draft
   PARSED!
   Trying Ellipsis: GIVE ME WIDTH AND DRAFT OF THE KENNEDY
   Give the beam and draft of the ship KENNEDY JF
   (Beam = 130 Draft = 36)
3. Who is her commander?
   PARSED!
   Give the rank and name of the commander of the ship KENNEDY JF
   (Rank CAPT Name MOFFETT P)

While LADDER is very helpful for providing information from English input, it represents expertise in but a limited domain. Further work in this area must include knowledge bases that are independent of the language processing system itself. This is important for generalizing the application of the system and for allowing flexibility in dealing with the changing contexts that occur in real-world communications.

Other related work in the area of natural language processing includes speech understanding systems (e.g., the HEARSAY II model by Lesser and Erman 1979), machine translations of documents into another natural language, document understanding which could assimilate information and output a summarization, and document generation which would translate stored information in the computer memory into natural language. In addition, robots that communicate in a natural language will bring into reality what was previously only the content of science fiction.

Once machines have the ability to understand natural language, then almost anyone will be able to interact successfully with a computer. First, however, more research in the field of psycholinguistics is required for a fuller understanding of the components of communication such as the representation and utilization of real-world knowledge, the role of planning and reasoning in communication, and so on. Once this is accomplished, such a system will surely pass the Turing test of machine intelligence.

3.3 Robotics

No overview of artificial intelligence would be complete without some discussion of robotics. It is an exciting field and exists today due to the integration of a number of subspecialty fields in artificial intelligence, such as machine vision, planning and scheduling, automatic programming, problem solving, and so on, into a single system. This research has led to several techniques for modeling states of the world (environmental representations) and for describing changes from one world state to another. Resulting from this modeling is a better understanding of how a computer can: (a) represent a given world; (b) generate plans for actions; (c) execute a particular task; and (d) monitor the effectiveness of the executed plans. The solutions to these modeling problems have had various degrees of success in laboratories working with robots, with the work progressing in conjunction with the development of computers, sensors, and effectors (i.e., devices for effecting a particular change, such as a mechanical arm).

In a most general sense, "robot" refers to a mechanical device that displays humanlike abilities to perform physical tasks (Raphael 1976). One of the first modern robots was developed in the early 1960s at Johns Hopkins University. It was a mobile unit, completely self-contained with no cable or radio link to any computer, power supply, or human operated terminal. Its power source was a battery and its decision-making abilities were minimal so that the final system design was of a machine whose sole purpose in life was to charge its own battery.

The way this mission was executed was based on a number of related elements. First, as the robot traversed a hallway, sonar measurements kept it relatively centered. By combining photocells, lenses, and circuits, the robot could find electric outlet cover-plates into which it would put its plug-shaped "hand" in order to "feed" itself.

Following this creation, laboratories around the world began actively engaging in robot research and development. The basic components in a robot system are sensors, effectors, and computers. Sensors serve to detect light intensity, color, touch, pressure, heat, sound, distance to obstacles, and so forth. Incorporating such human senses as taste and smell has, as yet, been deemed unnecessary. Some commonly used sensors are television cameras and photoelectric cells for "seeing," contact switches for "touch," pressure and force sensors, and mechanisms that allow the robot to know the position and status of all its robot parts. Presently, there is research going on for developing means of distance perception. One example is a two-camera system at the Jet Propulsion Laboratory (Pasadena, California) whereby a robot can compute distances by comparing two pictures.

Effectors can assume the form of mechanical arms or hands, elaborate or specific, simple or general. Also, wheels and other means of robot mobility are classified as effectors. Finally, computers used in robots are becoming more compact yet with larger memory stores, and are faster and less expensive as well.

Although there are impressive practical uses of robotics throughout the world, the first-generation robots were constructed mainly out of pure research curiosity. After the initial flurry of excitement in the 1960s and early 1970s, emphasis shifted from building single system robots to more emphasis on the component parts. This trade-off is due to the problems of designing a system that has general-purpose problem-solving skills versus one with more focused expertise. The next generation of robots, which will arise from the work being done today on the individual parts, may possibly resolve this conflict by their being an expert in a given domain (e.g., housekeeping) but also possess a wide repertoire of general problem-solving skills.

As can be seen from the preceding applications of artificial intelligence, there is a diverse group of problems currently under investigation. In addition, many important applications have been necessarily omitted from discussion, such as the work being done with computer-assisted instruction (CAI) (see *Computer-assisted Learning*), computer perception, problem-solving methods, knowledge acquisition, programming languages, metaknowledge (i.e., awareness of one's own cognition), and combinatorial and scheduling problems, to name a few. New ways of exploiting the computer's potential are being devised all of the time. While the Turing test has yet to be passed, the time is not far off when a machine will be demonstrating humanlike, general intelligence. As Turing wrote in the 1950's:

> I believe that at the end of the century the use of words and general educated opinion will have altered so much that one will be able to speak of machines thinking without expecting to be contradicted. (Roberts 1981)

4. Future Research

As computers are becoming less expensive and more widely available, their applications are becoming more diverse as they assume more roles in society. Furthermore, the potentials for application are only bound by imagination.

The future applications of artificial intelligence will proceed from the development of science and technology. For instance, knowledge-based expert systems will have more extensive and better organized knowledge resources with predictive abilities combined with their tutorial and consultative skills. Such systems may be able to predict natural catastrophes (e.g., earthquakes) or financial catastrophes (e.g., stock market plunges). Expert systems will become more widely available for personal use, in homes and offices, giving advice or instructions on a broad range of topics. Presently the creation of expert systems requires much time for the interviewing of experts. In the future, this process may be automated. This will follow from simplifying user/computer interactions as well as from incorporating adaptive production systems in which a computer can "learn by doing." Recognition of analogies between present and past problems will allow for greater computer flexibility and generalizability of skills.

Development of computer software will result in new programming languages, further simplifying the interaction between user and computer. Additionally, developments in the field of natural-language processing will one day produce machines capable of carrying out instructions supplied by written or spoken commands. These computers will be fluent and skilled in the use of many natural languages.

Hardware advances in the form of more reliable and less expensive sensors, effectors, and computers will surely be utilized in creating the next generation of robots. These robots will not only be invaluable assistants, doing the work people tend to put off until tomorrow, but will also assume responsible positions in jobs that are dangerous (e.g., steel mill furnace operator) or just tedious (e.g., assembler in a factory).

Additional hardware advances will make it possible for the parallel processing of several rules simultaneously in a rule-based system, rather than by serial application. This will speed up processing time and simulate some human processing abilities. The coordination of a large community of somewhat independent systems will lead to more general machine intelligence, with the systems communicating with each other to solve problems cooperatively.

Cognitive psychologists, by answering more questions on the representation and organization of knowledge types in memory, can provide more detailed structural specifications for implementation in the domain of computer simulation. Likewise, determinations of how humans deal with knowledge that is uncertain or indefinite will profit computer simulations. For a machine to function intelligently, it must also have some commonsense knowledge of cause and effect.

Other areas in the future of artificial intelligence that will be explored include the incorporation of metaknowledge into the system. This is invaluable in all intelligent behaviors, from communication to problem solving. Systems with this "executive" or introspective function will know when and how to apply other knowledge.

As implied in the forgoing article, artificial intelligence has important applications in education. Computers will be appearing in greater numbers in the schools, providing a medium for children's expression and experimentation. This may be accomplished through programmed instruction in which course material is organized and tailored to the individual (see *Individualized Instruction*). By the computer's presenting material on a television screen and recording learners' responses, students' progress and mistakes can be recorded and summarized for teachers for evaluation and possible remediation. Instruction in spelling, arithmetic, and language learning may be given to pupils by the computer, with private tutorials built into the system in order to address individual needs. Finally, by turning the tables and allowing children to "teach" computers via programming, pupils learn problem-solving skills by being forced to specify their ideas in the designing of algorithms and in organizing the problem-solving task into chunks or subroutines.

Finally, educators are concerned that the artificial intelligence of machines be directed entirely to humane uses. To this end, Asimov created the "Three Laws of Robotics" which may typify the conscious concern of moral and ethical issues by those working with "smart" machines:

(a) a robot may not injure a human being through inaction, allow a human being to come to harm;
(b) a robot must obey all commands given by a human being except in the event that such orders might conflict with the First Law; and
(c) a robot must protect its own existence as long as such protection does not conflict with either the First or the Second Law. (Asimov 1970)

See also: Computers in Education; Computer Technology and Telecommunications; Programmed Learning

Bibliography

Asimov I 1970 *I, Robot.* Fawcett, Greenwich, Connecticut

Bobrow D G 1962 *A Question-answerer for Algebra Word Problems.* Artificial Intelligence Project, Computation Center and Research Laboratory of Electronics, MIT Press, Cambridge, Massachusetts

Brown A L 1978 Knowing when, where, and how to remember: A problem of metacognition. In: Glaser R (ed.) 1978 *Advances in Instructional Psychology*, Vol. 1. Erlbaum, Hillsdale, New Jersey, pp. 77–165

Davis R, King J 1977 An overview of production systems. *Machine Intelligence* 8: 300–32

Duda R O, Gaschnig J G 1981 Knowledge-based expert systems come of age. *BYTE* 6(9): 238–81

Feigenbaum E, Buchanan B, Lederberg J 1971 Generality and problem solving: A case study using the DENDRAL program. *Machine Intelligence* 6: 165–90

Hendrix G G, Sacerdoti E D 1981 Natural-language processing: The field in perspective. *BYTE* 6(9): 304–52

Lesser V R, Erman L D 1979 An experiment in distributed interpretation. University of Southern California Information Sciences Institute Report No. ISI/RR-79-76, May

McCorduck P 1979 *Machines Who Think: A Personal Inquiry into the History and Prospects of Artificial Intelligence.* Freeman, San Francisco, California

McDermott J 1980 R1: An expert in the computer systems domain. Proceedings of the First Annual National Conference on Artificial Intelligence, pp. 269–71

Newell A, Simon H A 1972 *Human Problem Solving.* Prentice-Hall, Englewood Cliffs, New Jersey

Nilsson N J 1980 *Principles of Artificial Intelligence.* Tioga, Palo Alto, California

Quillian M R 1968 Semantic memory. In: Minsky M (ed.) 1968 *Semantic Information Processing*, MIT Press, Cambridge, Massachusetts

Raphael B 1976 *The Thinking Computer: Mind Inside Matter.* Freeman, San Francisco, California

Resnick L B 1976 Task analysis in instruction design: Some cases from mathematics. In: Klahr D (ed.) 1976 *Cognition and Instruction.* Erlbaum, Hillsdale, New Jersey, pp. 51–80

Roberts S K 1981 Artificial intelligence. *BYTE* 6(9): 164–78
Shortliffe E H 1976 *Computer-based Medical Consultations: MYCIN*. Elsevier, New York
Waterman D A 1970 Generalization learning techniques for automating the learning of heuristics. *Artificial Intelligence* 1(1-2): 121–70
Waterman D A, Hayes-Roth F (eds.) 1978 *Pattern-directed Inference Systems*. Academic Press, New York
Weizenbaum J 1976 *Computer Power and Human Reason: From Judgement to Calculation*. Freeman, San Francisco, California
Whitehead A N, Russell B 1968 *Principa Mathematica*, 2nd edn. Cambridge University Press, Cambridge
Winograd T 1972 *Understanding Natural Language*. Academic Press, New York
Winston P H, Horn B K P 1981 *LISP*. Addison-Wesley, Reading, Massachusetts

V. Shute

Assertiveness Training

Assertiveness training is a systematic approach developed by behavior therapists (Wolpe 1958) to help people learn effective ways of interacting with others in a variety of interpersonal situations. Specifically, assertiveness training has been employed to teach people how to: (a) stand up for their legitimate rights; (b) initiate and refuse requests; (c) give and receive compliments; (d) initiate, maintain, and terminate conversations; (e) express love and affection; (f) express personal opinions; and (g) express justified anger and annoyance to and/or with a variety of people, including friends, intimate others, parents, family, authority figures, and strangers (Galassi and Galassi 1978).

Effectiveness, from an assertiveness training perspective, is defined in terms of two components. First, an effective response is one that has a high probability of achieving a person's goals in an interaction (e.g., refusing an unreasonable request). Second, it is one that considers the rights and feelings of the other person(s) involved in an interaction and has a low probability of threatening, humiliating, or otherwise hurting them. Behaviors that meet both of these criteria have been traditionally called assertive and have been distinguished from aggressive and nonassertive responses. Aggressive behaviors are those that may achieve a person's goals, but have a high probability of degrading, threatening, humiliating, or hurting recipients of the response (i.e., usually offensive expressions that do not consider the rights and feelings of others). Nonassertive responses are those that often fail to achieve the person's goals through nonexpression (e.g., agreeing to an unreasonable request), avoidance (e.g., avoiding situations where a conversation may have to be initiated with a stranger), or weak expression (e.g., asking for a favor in a hesitating, overly apologetic manner).

Thus, assertiveness training attempts, in a systematic way, to teach people specific verbal, nonverbal, and cognitive skills to express their feelings, needs, preferences, and opinions in a potentially nonoffensive, nonthreatening, and nonaversive manner in specific interpersonal situations. Assertiveness training has been conducted in both individual and group counseling and educational contexts and typically employs such training procedures as role playing, modeling, instructions, feedback, prompting, and specific homework assignments. More recently, imaginal rehearsal, cognitive restructuring, and decision-making strategies have been employed with some success. Although specific exercises devoted to helping people become aware of their rights as human beings have been advocated (e.g., Lange and Jakubowski 1976), no reports of research testing their efficacy have yet appeared.

An additional important goal of assertiveness training, when practiced by a knowledgeable trainer, is that of helping people generate a variety of responses to particular situations and of aiding them to learn to decide for themselves when and under what conditions a direct, but nonoffensive expression of opinions, needs, or feelings is appropriate. Stated simply, a second goal of assertiveness training is to provide participants with a choice of responses that may be used in their everyday interactions. A person who is continually expressing his or her opinions, wants, and feelings in all situations, at all times, and with all people is likely to be no more effective than the person who is never able to make his or her wants and desires known to others. This important goal is one that is overlooked by many poorly trained practitioners. The potential consumer of assertiveness training would be well-advised to terminate participation in any program that denies his or her right to choose when to be assertive, and that promotes the view that one must always be assertive to be interpersonally effective and psychologically healthy.

Readers interested in learning more about teaching assertiveness should acquire knowledge on: (a) the history and theory of assertiveness training, (b) research on assertiveness training outcomes and techniques, and (c) practical guidelines for conducting assertiveness training. A starting curriculum would include the seminal works of Salter (1949) and Wolpe (1958, 1968); research reviews by Bellack (1979); Brown and Brown (1980), Galassi and Galassi (1978), McFall (1982), and Rich and Schroeder (1976); and the professional manuals of Eisler and Frederickson (1980), Lange and Jakubowski (1976), Shelton and Ackerman (1974), and Trower et al. (1978). Two excellent articles (Ralph 1982, Shelton 1977) pertaining to ethical and professional issues in assertiveness training would also be included in an introductory curriculum.