DICTIONARY OF

ARTIFICIAL INTELLIGENCE

DICTIONARY OF

ARTIFICIAL INTELLIGENCE

Dennis Mercadal

VAN NOSTRAND REINHOLD
New York

Library of Congress Catalog Number 90-38646
ISBN 0-442-00451-6

Printed in the United States of America

Van Nostrand Reinhold
115 Fifth Avenue
New York, New York 10003

Chapman and Hall
2-6 Boundary Row
London, SE1 8HN, England

Thomas Nelson Australia
102 Dodds Street
South Melbourne 3205 Victoria, Australia

Nelson Canada
1120 Birchmount Road
Scarborough, Ontario M1K 5G4, Canada

16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1

Library of Congress Cataloging in Publication Data

Mercadal, Dennis, 1943–
Dictionary of artificial intelligence / Dennis Mercadal.
p. cm.
Includes bibliographical references.
ISBN 0-442-00451-6 (pbk.)
1. Artificial intelligence—Dictionaries. I. Title.
Q334.2.M47 1990
006.3'03—dc20 90-38646
CIP

Designer: M.P ° Design.

Preface

The main goal I had in writing this book was for you, the reader, to find the large majority of terms you are looking for in this book. I use the words "large majority" because I know this dictionary cannot include all the terms and concepts in this dynamic field. Once you find the term, my hope is that you will find the information accurate and useful.

I wrote this dictionary for the novice and for the professional in artificial intelligence. I placed emphasis not just on defining terms but in giving numerous concrete examples. For the more experienced professional, I included contrasts with other terms and spent time pointing out the implications of the concept.

The languages of artificial intelligence have received special attention. Trying to understand artificial intelligence without knowing LISP, PROLOG, or Smalltalk is like visiting a foreign country without knowing the language. You cannot obtain a true appreciation of the country without knowing the language.

Terms from related fields, such as logic, are included because concepts from other fields have much to offer to the field of AI.

Some terms not specific to artificial intelligence are included because they are frequently used in the AI literature. Examples include data types and pointers.

Many terms have multiple definitions. A given term may have definitions that are related, unrelated, or even contradictory. Rather than simply giving you the "correct" definition of a term, I have often provided a full range of definitions.

Expert systems are highlighted because most people reading this text will be looking for information on expert systems and related areas. An appendix of expert systems is provided.

Many new areas of AI are included. You will find information on genetic algorithms, neural networks, model-based reasoning, case-based reasoning, object-oriented programming, intelligent data bases, and more.

The many references included can steer you onto further information about a subject.

My final goal is to make the text interesting so that readers will use it as more than a reference book. I hope this book helps open insights into this exciting field.

Acknowledgments

All of the professionals cited in the references have contributed greatly to this work and their efforts are greatly appreciated.

Contents

Preface ix
Dictionary of Terms and Concepts 1
Signs, Symbols, and Numbers 308
Expert System Shells/Tools 309
References 318

A

A* algorithm A heuristic search that is an enhanced version of branch-and-bound search. The A* algorithm is an optimal type of heuristic search, in that it finds the optimal goal and the optimal path to the goal rather than just any goal, or just any path. It does not, however, guarantee that the effort put forth in the search is minimized. It is a search algorithm in which the next node chosen is based on the cost thus far from the beginning and on heuristics that estimate the future cost to the final goal. Estimates are made of the current accumulated cost and the estimated cost to finish. These two costs are combined in a final result called A'. The different A's for each alternative are then compared and the most promising one is chosen. This approach prevents the exploration of paths that are too time consuming. The other alternatives are kept in case the most promising path does not turn out to be the best path after all. If the most promising path turns out not to be the best path, the A* algorithm searches the next most promising path. The A* method works well as long as the future cost is not overestimated. As long as the A* procedure does not overestimate the correct future cost, it is guaranteed to discover the best route in comparison with other search methods that have the same information. The A* algorithm is much more efficient than breadth-first search. When the heuristic for estimating distance to the goal is trivial, the A* algorithm reduces to branch-and-bound/uniform cost search. Like all best first-search techniques, it is memory hungry. *See also* Beam search for a type of search that cuts down on the memory required (Forsyth and Naylor 1985, 155–159; Flamig 1987, 18–26; Wolfgram 1987, 66–67; Barr and Feigenbaum 1981 vol. 1, 64–73; Chabris 1989, 159–162, 285; Luger and Stubblefield 1989, 166–167; Smith 1989, 1; Winston 1984, 113–114).

AAAI *See* American Association for Artificial Intelligence.

AAL *See* Adventure Authoring Language.

Aaron An artificial intelligence program devoted to drawing. The three concepts the program uses are the differences between inside and outside, closed and open, and figure and ground. The program consists of different experts capable of carrying out different tasks. The architecture is reminiscent of Hearsay II (Johnson 1986, 211–223).

a-list An association list (Winston and Horn 1988, 574; Townsend and Feucht 1986, 243).

A^q algorithm An algorithm used in propositional logic for selecting training instances that are considerable distances from one another in the instance space (Smith 1989, 12).

Abduction An illegal inference that generates explanations for a phenomenon. The explanations generated may not be correct. Example: If X implies Y and Y is found to be true then by deduction X is true. Deduction is a much sounder computation. Medical diagnosis is a form of abduction. Another example: For all X, X is having a heart attack if X is having chest pains. Abduction may be considered the first part of the generate-and-test process. *Compare with* Deduction. *See also* Plausible reasoning (Charniak and McDermott 1985, 21; Chabris 1989, 285; Smith 1989, 1; Olsen et al 1987, 118).

Abductive inference The generation of hypotheses to explain a phenomenon. *Contrast with* Deduction (Jackson 1986, 117).

Absolute rule A rule that evaluates a circumstance. For example, in a problem-solving process an absolute rule may detect the velocity of a particle that has been computed and the absolute rule then calculates the acceleration (Hunt 1986, 41).

Abstract class A class that holds the methods of all its subclasses. A class that spawns subclasses, not instances. A class whose members consist of other classes. Example: The animal class is an abstract class because it consists of other classes—mammals, birds, carnivores, and so on (Miller 1989, 64; Thomas 1989, 238).

Abstract data type A data type and the operations relevant to the objects in the domain of the data type. It describes the representation of objects. It hides the details of the operations of the data type. Examples of abstract data types are classes in Smalltalk. Each class in Smalltalk has operations called methods. Another example would be an abstract data type called animal. Methods for communication are associated with each animal. A dog in the animal class barks when a message requesting it to bark is passed to it. A cat meows when a message to talk is passed to it (Olan 1988, 37; Miller May 1988, 45; Pountain 1986, 227; Swaine June 1989, 114–116).

Abstraction 1. The technique of using simplified assumptions to cope better with large solution spaces (for example, instead of analyzing all the subtle details of different types of interest rates, simply looking at the general classes of interest rates). When a general class of interest rate candidates is established, then the specific interest rate computations in that class can be examined. Human experts routinely use abstraction, and it is one reason why they can quickly solve problems without having to go through extended logic computations. In expert systems involved in planning, hierarchical planning is regularly used. That is, an abstract plan is first formulated. Then more detailed implementations of the plan are generated. Such planning forms a hierarchical structure of the plan from the more abstract to the more specific. All expert system factbases employ abstractions of the real world to some degree. 2. The process of hiding implementation details. Such abstraction can take place with data and with flow of control. *See also* Data abstraction (Hayes-Roth et al 1983, 70, 85; Walters and Nielsen 1988, 281–283, 326–327; Barr and Feigenbaum 1981 vol 3, 516–518, 528–530).

Abstraction barrier A protective layer concealing lower-level details, which can be created using either procedure or data abstraction (Winston and Horn 1988, 437).

Abstract object type An object/class used to spawn other objects/classes that are not instances. For example, the object animal may be used to spawn mammals and birds. Neither mammals nor birds are specific concrete objects. Abstract superclass (Duntemann 1989, 132).

Abstract operator An operator used to delineate abstract actions in planning systems. Once the abstract operations have been delineated the planning system works on filling in the details of the plan (Smith 1989, 2).

Abstract superclass *See* Abstract object type (Duntemann 1989, 132).

ABSTRIPS A successor program to STRIPS. It is a robot-planning program that uses hierarchical planning. In the more abstract plans only the essential conditions are considered. It then moves to more concrete plans, where the less important conditions are considered. All conditions are assigned criticality scores that are used to decide how important a given detail is. Its use of hierarchical planning results in less search and less backtracking than its predecessor, STRIPS. It carries out a length-first search which forms a complete overall plan first before carrying out actions (Hayes-Roth et al 1983, 104–106; Barr and Feigenbaum 1981 vol 3, 517–518, 523–530).

Access-oriented method A computational method that is triggered by the reading of data or changes in data (Hunt 1986, 41; Smith 1989, 2).

Access procedure A LISP procedure that includes constructors, readers, and writers. *See also* San Marco LISP Explorer.

Accidental property The property of an object that may be possessed by all objects in a class, but which is not crucial to the definition of the object. *Compare with* Essential properties and Typicality (Jackson 1986, 67).

Accuracy **1.** The measurement of expert systems achieved by comparing the actual correct predictions with the expected number of correct predictions. There are statistical techniques such as Chi Square that help in measuring accuracy. Other important measures of the value of an expert system are adequacy, validity, reliability, breadth, practicability, adaptability, credibility, and generality. **2.** The difference between the calculated location of a robot arm and the real location of the arm in robotics (Hunt 1986, 41; Marcot 1987, 44).

ACE An expert system that detects problems in telephone systems and makes recommendations for repairs. It is a rule-based forward-chaining system (Hunt 1986, 41–42).

ACES A program that places labels on maps and uses information about locations on the map to label the point with the correct label size and type font. It is implemented in LOOPS (Hunt 1986, 42).

ACRONYM A model-based vision program that uses geometric matching and algebraic consistency checking to interpret three-dimensional interpretations of images. The user tells the system what images to expect and the program attempts to find those images in the data it receives. It employs a powerful generate-and-test procedure. ACRONYM has been used in aerial photograph interpretation. It makes a clear distinction between image domain and scene domain (Mishkoff 1986, 5–13; Barr and Feigenbaum 1981 vol 3, 313–321; Townsend and Feucht 1986, 183; Waldrop 1987, 105–106; Winston 1984, 164–166).

ACT A computer model of human cognition; a successor program to HAM. ACT includes both short-term and long-term memory. Its long-term memory is made up of nodes and arcs. The nodes represent concepts and the arcs represent relations between concepts. The short-memory also consists of nodes, but the nodes represent more active concepts. There is a separate production rule component. The production rules examine short-term memory and based on the results initiate changes in long-term memory by activating nodes in long-term memory and placing the nodes in short-term memory. The placing of nodes in short-term memory is the process that is responsible for ACT's ability to shift attention readily. ACT can be programmed to demonstrate a variety of cognitive tasks (Barr and Feigenbaum, 1981 vol 3, 50–55).

Act One of the four cycles in the recognize act cycle of OPS5 expert systems. Act is responsible for firing the commands found in the right-hand side of the production rule after conflict resolution has taken place. The four cycles in the recognize act cycle are conflict resolution, act, match, and halt.

Action An action carried out on the right-hand side of a production rule in a production system. The actions vary depending on the production system. Some actions may include making, deleting, or modifying a working memory element, building a rule, writing to the screen, opening and closing a file, and binding a value to a variable (Brownston et al 1986, 51–60).

Activation A number representing the degree of attention an entity may receive in an artificial intelligence program. A rule with a high activation number is more likely to be used than one with a low activation number (Brownston et al 1986, 284–285; Hunt 1986, 42).

Activation accommodation The use of sensory feedback and control mechanism in robotics to manage the movement of a robot in a changing environment; analogous to the human nervous system, which allows us to sense and adjust to objects in our environment (Hunt 1986, 42).

Activation cycle The time it takes for activation to be spread to adjacent objects in an activation network. More than one activation cycle may take place during a single recognize-act cycle (Hunt 1986, 42).

Activation environment The environment in force when the procedure needing the free variable values is called (Winston, and Horn 1984, 54).

Activation filtering A type of data filtering in which the datum is assigned a number that determines the relevancy of the datum for a particular part of the problem-solving process. *See also* Certainty filtering for another type of data filtering (Brownston et al 1986, 310).

Activation function A function that computes the neural net output based on the input to the system and the system's current state (Obermeier and Barron 1989, 219).

Activation network A network of nodes and arcs. The arcs represent relationships between the nodes. The nodes represent objects and each arc may have a number that represents the strength of the relationship. When an object is processed its activation level can be altered and it can, in turn, alter the activation level of adjacent objects (Brownston et al 1986, 441).

Activation time constant A constant used in the instar activation equation which determines the rate of decay of learning over time (Caudill Nov 1988, 60).

Active data structure A data structure that both represents and processes data. Active data structures are found in the connection machine. This type of data structure receives instructions from an external source. It then uses its own built-in procedures to carry out the request. This structure differs from conventional passive data structures which are manipulated by external procedures. Active data structures include objects, frames, sets, trees, butterflies, strings, arrays, and graphs (Hillis 1985, 18).

Active illumination A means of increasing the amount of information from a scene by varying the lighting (Rosenberg 1986, 3).

Active image A demon that can be used to call an image which can then be used to choose from a set of options. Example: An image with five buttons can be called. Each button represents a choice. A mouse can be used to press one of the buttons to execute a procedure (Luger and Stubblefield 1989, 549).

Active instance selection The search of the instance space in learning programs to find active instances, which can be used to produce rules (Smith 1989, 4).

Active object An object that can take action without having to be called by another object. The object may, for example, take action in response to a change in data (Wegner 1989, 246).

Active value 1. A value in an artificial intelligence program which, when added, deleted, scanned, or modified, can trigger a procedure called a demon. The demon may carry out such procedures as requiring a password to access the value, modify another value, place constraints on any modification of the value, display a graph of the variable being reasoned about, or trigger-rule sets. Active values can be used in building truth maintenance systems. They can be used to implement nonmonotonic reasoning. 2. Procedural attachment and demons. That is, active value can refer to the procedure which is activated when a value is somehow accessed (Tello 1989, 28, 190; Shepard 1987, 76; Rauch-Hindin 1988, 58; Walters and Nielsen 1988 138–139; Harmon and King 1985, 58, 126; Rosenberg 1986, 3; Waterman 1986, 388).

Actor 1. An object that carries out actions in conceptual dependency theory. 2. The fundamental entity in Carl Hewitt's actor theory. Actors differs from the standard Smalltalk approach in that actors take an intermediate value as a message and continue processing. In Smalltalk the object always returns a value in response to a message (Chabris 1989, 286; Smith 1989, 4).

ACTOR An object-oriented language developed by the Whitewater group which supports inheritance and is similar to Smalltalk. ACTOR's code is closer to C than standard implementations of Smalltalk. One feature of ACTOR is that it uses windows (Tello 1989, 115).

Actor language An object-based language that is capable of concurrency, but does not support inheritance and classes. Its promise is in open systems that use parallel processing. Actor languages should not be confused with the ACTOR language from the Whitewater Group which does support inheritance (Tello 1989, 285; Wegner 1989, 249).

Actor theory A theory developed by Carl Hewitt consisting of subprograms called actors. Each actor is able to respond to messages, change its internal state, send messages, or create a new actor. Each actor gains control of an artificial intelligence program at different times depending on the circumstances. It is somewhat reminiscent of Freud's id, ego, and superego theory in which each of these actors gains control at different times. Hewitt is currently developing APIARY, an implementation of actor theory.

Actual parameter An actual value that replaces the formal parameter, for example, Max(*A*, *B*,*C*). The procedure decides which of the three numbers represented by *A*, *B*, and *C* is the maximum number. *A*, *B*, and *C* are formal parameters for the procedure Max. The actual parameters are the numbers *A*, *B*, and *C* represent (Hunt 1986, 192).

Actuator The mechanism that powers the movement of a robot; the three common types are pneumatic, hydraulic, and electric (Mishkoff 1986, 5–17).

Acyclic data base A data base in a PROLOG program obtained from the original data base through queries and deduction (Shmueli et al 1986, 248).

Acyclic graph A graph in which a descendant may have more than one parent.

ADA A general-purpose, strongly typed computer language which, while not usually used in artificial intelligence, has certain characteristics that make it attractive for AI applications: Its memory management is the type needed in real-time expert systems. ADA uses recursion and allows for the possibility of object-oriented programs. The language permits a compile time version of polymorphism and multitasking. It has the ability to carry out parallel processing. It is portable, readable, has good error-handling capabilities, and is able to implement real-time programming. ADA is an object-based language, but it is not a class-based language. ADA should be considered especially in very complex applications (Tello 1988, 397–398; Wegner 1989, 247; Roland 1987, 55).

Adaline A single-neuron system which can learn to recognize a single letter, even if the orientation of the letter is rotated; it uses the Delta learning rule (Caudill Feb 1988, 56; Tank and Hopfield 1987, 104; Caudill June 1988, 53).

Adaptability The ability of an expert system to deal with different hardware and software, and different types of information. Other important measures of the value of an expert system are adequacy, accuracy, reliability, breadth, practicability, validity, credibility, and generality (Marcot 1987, 45).

Adaptive Having the ability of a robot to alter its control system in response to a changing environment (Hunt 1986, 43).

Adaptive control theorist A researcher interested in obtaining a synthesis of learning theory and control theory (Jorgensen and Matheus 1986, 31).

Adaptive learning Learning employed in artificial intelligence for dealing with noisy environments. The program builds a model that correctly approximates the inputs and outputs; examples include automata learning, parameter learning, statistical algorithms, and structural learning (Smith 1989, 4–5).

Adaptive linear element A single-neuron system capable of learning to discriminate patterns even if the spatial orientation or size of the pattern is changed. It could, for example, recognize an "A" even though the "A" is upside down or half the original size; developed by Widrow.

Adaptive linear network *See* Adaline.

Adaptive pattern-recognition processing A pattern-recognition procedure used in the SAVVY front end that makes educated guesses about the user input. *See also* Adaptive query recognition (Tello 1988, 27).

Adaptive production system A production system that can modify its production rules with experience (Smith 1989, 5).

Adaptive query recognition A language-recognition process that is used in the natural language front-end Savvy; a holistic approach to pattern matching. Adaptive query recognition avoids analysis of the sentence. It matches the query with the statement in its data base that most closely resembles the query. It answers the question of which statement in the data base best approximates the query. *Contrast with* the logic-based approach found in Clout (Rubin 1985, 46).

Adaptive resonance theory (ART) A self-organizing neural network noted for its autonomy, complexity, and power; based on outstar learning. It has the capabilities of categorizing input patterns without outside feedback, learning new patterns, recategorization, and immediate recall (Caudill May 1989, 57; Caudill Aug 1989, 61–67).

Adaptive robot A robot that can make adjustments to changes in its environment. It possesses enough artificial intelligence programming to process sensory information and use that information to give motor commands (Rosenberg 1986, 4).

Adaptive systems researcher A researcher interested in the properties of living organisms, particularly properties that contribute to adaptation such as self-modification and organization. These researchers have given impetus to the study of neural networks (Jorgensen and Matheus 1986, 31).

Adder The process that keeps track of information from filters and demons in a perceptron (Reece 1987, 52).

Address-induced representation The connecting of cells of a data structure by the addresses of the cells (Hillis 1985, 91).

ADEPT A battlefield management expert system developed by TRW using the expert system language ROSIE (Hunt 1986, 4).

Adequacy A measure of the number of different conditions in the knowledge domain with which an expert system can cope. In a medical expert system, the number of symptoms the expert system can handle is a measure of adequacy. Other important measures are breadth, accuracy, reliability, validity, practicability, adaptability, credibility, and generality (Marcot 1987, 44).

Ad-hoc polymorphism The polymorphism found in Common Lisp Object Oriented systems. It diffe s from standard polymorphism in that the system is more sensitive to argument types than is standard polymorphism (Gabriel Sep/Oct 1989, 41).

Admissibility theorem A means of ascertaning whether or not A*, in conjunction with a given heurist..., will always find the optimal solution to a problem. To find the optimal path it must be guaranteed that calculations of the remaining distance to the goal state be underestimates of the correct distance (Bratko 1986, 284; Chabris 1989, 286).

Admissible A property of search. When a search procedure is admissible, it means the procedure always finds the best solution path, not just any solution path (Luger and Stubblefield 1989, 165–166; Smith 1989, 1; Chabris 1989, 286; Bratko 1986, 273, 284).

Adventure Authoring Language A language designed for writing text adventure games, which is a combination of object-oriented programming, logic programming, and LISP (Amsterdam 1988, 18–39).

Advice taking The ability of some artificial intelligence programs to learn from instructions (Hayes-Roth et al 1983, 153; Barr and Feigenbaum 1981 vol 3, 345–349).

ADVISOR An advisory system for users of MACSYMA. It analyzes what the user is attempting to do, diagnoses the user's mistakes and gives the user advice on how to achieve his goal (Hunt 1986, 44).

Advisory system An expert system that emphasizes the use of advice rather than commands; used to make recommendations, not to make final decisions. Advisory systems also s. s detailed .planation systems. "Advisory system" may be a more appropriate term than "expert system" for many applications (Hunt 1986, 44; Walters and Nielsen 1988, 165).

After inheritance The function in a frame-based system stored in an object's slot and applied after the inherited function is applied. In this arrangement the object function may be used to massage the data further after the more general inherited function has completed generating data. Th. t function may also be used simply to record the data produced by the inherited function. *Contrast with* Before inheritance (Walters and Nielsen 1988, 242–243).

After-when-modified demon A type of demon summoned after the value of a slot is altered (Matthews 1987, 80).

AGE An expert system tool developed at Stanford. AGE stands for attempt to generalize. Its development was influenced by Hearsay II. It is capable of using either a backward-chaining paradigm or a blackboard paradigm (Johnson 1986, 145; Hayes-Roth et al 1983, 196–198, 203–209; Harmon and King 1985, 140).

Agenda 1. A means of organizing goals in expert systems. A listing of prioritized activities waiting to be acted on. A means of control. A variation of a queue, with the exception that individual goals can be assigned priorities over other goals. This structure is useful in a situation in which goal priorities are subject to frequent change. The goal priorities are usually controlled by a rule set. An agenda can be reasoned about by an expert system. Other data structures used in organizing goals are tree structures, queues, and stacks. **2.** Agenda is also used to refer to conflict sets (Matthews 1987, 82; Hayes-Roth et al 1983, 16–17, 399; Brownston et al 1986, 281–282; Barr and Feigenbaum 1981 vol 1, 338–339; Tello 1988, 311; Harmon et al 1988, 173; Smith 1989, 7).

Agenda-based system A system that uses an agenda as a means of central control. The agenda can control which set of rules may be executed next. The agenda is a series of tasks that are not static, but may change throughout a program run. Tasks may be added or removed from the agenda because of the execution of other tasks, or as the result of the execution of a rule. There is an intense interaction between the agenda and the rest of the system as each influences the other. It is easier to follow the logic of an agenda-based system than a pure production system. Examples of agenda-based systems include CENTAUR, DENDRAL, and AM (Aikins 1990, 3–4).

Agenda element A decision element used to record the future sequence of knowledge-based rules which will be scrutinized in a blackboard system. One of three types of decision elements in a blackboard system; the other two are plan elements and solution elements.

Agenda manager A mechanism in an artificial intelligence application for scheduling events against a real or simulated clock.

AI/COAG A medical expert system that analyzes and interprets blood coagulation. It assists the physician in diagnosing diseases of hemostasis. AI/COAG was developed at the University of Missouri School of Medicine (Hunt 1986, 45).

AIMDS An expert system language developed at Rutgers University using LISP. A frame-based system that uses multiple inheritance, deductive and nondeductive reasoning, belief maintenance, analogical inference, and procedural attachment. AIMDS has a set of procedures to detect inconsistencies in the knowledge base and is capable of procedure oriented representation (Hunt 1986, 45).

AI/MM A medical expert system that makes interpretations of data in the area of renal physiology, which understands physiology, physics, and anatomy and uses this knowledge to interpret conditions. A rule-based system developed at Stanford University using MRS (Hunt 1986, 45).

AI representation language A language that must include the ability to deal with qualitative data, represent abstractions, and infer facts. It must be able to handle both general principles and details. Metalevel reasoning and complex meaning must be handled effectively (Luger and Stubblefield 1989, 30).

AI/RHEUM A medical expert system that does differential diagnoses of diseases of rheumatology. A rule-based system developed at the University of Missouri School of Medicine using EXPERT (Hunt 1986, 45).

AIRID An expert system used to identify aircraft. An amalgamation of a rule-based and semantic network representation. It was developed at the Los Alamos National Laboratory using KAS (Hunt 1986, 46).

AIRPLAN An expert system used to help carrier based aircraft avoid such problems as running out of fuel by warning the air operations officer of a potential problem and making recommendations for avoiding the problem. It was developed at Carnegie Mellon using OPS7 (Hunt 1986, 46).

AI system A program capable of knowledge acquisition, goal-directed behavior, and skill acquisition (Firdman 1986, 81).

AI workstation A computer designed specifically for artificial languages such as LISP, sometimes called a LISP machine (Mishkoff 1986, G-1).

AKO A Kind Of. A link used in object-oriented languages. A link between classes and superclasses. The values of slots are inherited through this link. *Contrast with* an Is-a (Amsterdam 1987, 15; Jackson 1986, 76).

AL A robotics control language developed at Stanford.

ALADIN An expert system that assists in finding currently existing alloys, or in designing alloys that will fit specific requirements (Hunt 1986, 46).

ALDOUS A human-simulation program that takes on a variety of personalities, including decisive ALDOUS, hesitant ALDOUS, radical ALDOUS, and saint ALDOUS. The program was designed by John Loehlin (Frude 1983, 170–171).

AlgebraLand A computer-aided instruction program that keeps an audit trail of the student's problem-solving process, which can be examined later (Waldrop 1987, 205–206).

Algorithm An abstract description of a procedure or a program. A step-by-step procedure that if followed will lead to a correct answer. Artificial intelligence programs rely less on algorithms than do conventional programs. Instead, they use heuristics, a procedure which does not guarantee a correct answer (Johnson 1986, 83–84; Brownston et al 1986, 441; Townsend, 1987b, 376; Winston and Horn 1984, 17; Townsend and Feucht 1986, 26).

Algorithmic solution A sequential fixed procedure for arriving at a solution, associated with procedural programming and guarantees a specified output. It is faster than a heuristic state space search and should be employed when possible. Heuristic state space search can be used for many problems that do not have algorithmic solutions.

Aliasing A distortion found in computer-produced images. It is produced by inadequate sampling of the signal (Tanimoto 1987, 392–394).

ALICE A logic-based experimental language developed at the Institut de Programmation in Paris (Hunt 1986, 47).

Allophone A unit of speech that stands for a unique sound in a word.

Alpha notation An extension of the LISP function FUNCALL, which is used in CmLISP and which allows parallel mapping (Hillis 1985, 37–41).

Alpha-beta algorithm *See* Alpha-beta pruning.

Alpha-beta pruning An algorithm used in game search, with two parameters, alpha and beta, which are used to prune a search tree. It prunes the search tree by cutting off branches that are known not to be useful. Example: In a game situation, assume an opponent has a devastating response to a move the home player is considering. Once this response is discovered, there is no need to waste time looking at other possible responses the opponent could make to that move. It is used to augment the minimaxing algorithm by eliminating those portions of the search space that cannot possibly give a good solution. Under certain conditions it has the capability of searching twice as deep as MINIMAX when both procedures are limited to the same number of static evaluations (Chabris 1989, 179–185; Forsyth and Naylor 1985, 183–188; Winston 1979, 115–122; Barr and Feigenbaum 1981 vol 1, 88–93; Townsend and Feucht 1986, 42; Winston 1984, 117–126; Knuth and Moore 1975, 293–297).

Alpha-testing Testing of an expert system after a successful prototype has been developed and before it is placed on site. In alpha-testing the adequacy, accuracy, and reliability of the system is tested. *Contrast with* Beta-testing (Marcot 1987, 44).

Alternate worlds reasoning A sophisticated what-if capability found in some expert systems.

Copies of the knowledge base are made with some changes to see how these changes would alter the solution (Schuler 1987, 101).

Alvey project An AI project sponsored by the United Kingdom which is a response to the Japanese fifth-generation project (Hunt 1986, 47).

AL/X A knowledge-engineering language using frames, rules, semantic nets, forward and backward chaining, truth maintenance, and certainty factors. It is built by Intelligent Terminals, Ltd., using PASCAL (Hunt 1986, 47).

AM Automated Mathematician—developed by Doug Lenat for discovering mathematical concepts without outside help. It has the capability of learning from experience. Lenat calls the type of learning involved "discovery learning." Frames and rules are used for representing knowledge. It uses frames and slot filling to represent and modify concepts. It includes about 250 rules of thumb, which work in conjunction with the knowledge in the frames. It utilizes a survival-of-the-fittest process. Old frames are mutated and scores are assigned to the new frames. The low-scoring frames are dropped while the high-scoring frames end up with greater attention. Eventually these frames are used to develop new concepts. Example: Starting with general heuristic rules, a few primitive set concepts like equality and the empty set, it is able to derive mathematical operations like addition and subtraction. It goes even further by being able to assert, for example, that any even number can be expressed as the sum of two prime numbers. The author of AM holds that the flexibility available in LISP is one of the key reasons AM is able to show such originality. EURISKO is the successor program to AM (Johnson 1986, 191–193; Barr and Feigenbaum 1981 vol 1, 195–197; Waldrop 1987, 57–59; Forsyth and Naylor 1985, 212–214).

Ambiguity The doublespeak found in human language, Example: "I hit the man with the ball." Did I hit the man with a ball, or did the man I hit have a ball in his hand? Dealing with ambiguity is a major challenge for language-understanding programs (Rubin 1985, 44; Barr and Feigenbaum 1981 vol 1, 208–211).

American Association for Artificial Intelligence (AAAI) The society for artificial intelligence. The society publishes the *AI Magazine*. Its address and phone number are 445 Burgess Drive, Menlo Park, CA 94025, (415) 328-3123 (Harmon and King 1985, 250).

AML A computer language used for programming robots. It was developed by IBM and is based on AL (Mishkoff 1986, 5–19; Rosenberg 1986, 7).

AMORD A rule-based knowledge-engineering language using a forward-chaining paradigm. It uses truth maintenance and discrimination networks. It was developed at MIT using MACLISP (Hunt 1986, 47).

AMUID A battlefield expert system. A real-time, rule-based system with certainty factors, it uses incoming information to identify enemy units (Hunt 1986, 48).

Analogical inference The process of inferring similarities between two objects and using information about the known object to solve a problem concerning the unknown object. Example: Tigers have fur. Lions are like tigers. Lions probably have fur. Patrick Winston designed two programs, Macbeth and a cup-learning program, which used analogical inference. Both programs use a semantic network representation (Johnson 1986, 161; Barr and Feigenbaum 1981 vol 1, 146; Rosenberg 1986, 7).

Analogical means-end analysis The process of using a previously solved problem to solve a new, somewhat similar problem. Operators are used to reduce differences between the two problems. *Refer to* Analogical problem-space (Rosenberg 1986, 7).

Analogical problem-space A representation of a problem space in which each node represents a problem solution. The different problem solutions possess a degree of similarity. An operator transforms one problem solution to another similar problem solution. *Refer to* Analogical means-end analysis (Rosenberg 1986, 7).

Analogical reasoning A method of reasoning employing two separate systems that have some type of resemblance. One system is well understood. A characteristic of the well-understood system which somehow resembles the second system is applied to the second system to enhance understanding of the less understood system. Comparing love to a rose is an example of analogical reasoning. Both are beautiful, but both eventually wither away. There has been little success in employing this type of pattern matching in artificial intelligence. Frames are a prime candidate for dealing with analogical reasoning. An analogy can be made with frames by filling empty slots in a frame with corresponding attribute values from another frame. Assume a system with many different concepts that are represented as frames in a hierarchy. Assume another frame which represents a current problem to be solved. Now assume a rule base that actively compares the problem frame with the concept frames, and derives a best-fit score for each higher-level concept frame. The higher-level concept frame with the best score can be investigated further to see if a specialization of that frame produces an even better fit. The best-fit concept frame is then used to fill the slots on the problem frame. It is proposed by Doug Lenat that analogical reasoning rests not only on having a good representational technique, but more important on having access to a great deal of knowledge which can be efficiently searched for analogies. *See also* Analogical inference, a synonym (Lenat 1988, 73–74; Fischler and Firschein 1986, 44).

Analogical representation Representation found in some artificial intelligence programs that is somehow analogous to the structure it is representing in the real world. The structure of the representation gives information about the structure itself. An example of an analogical representation is a map of a city. The geometry theorem prover uses analogical representation. Diagrams of angles and figures are used to direct the reasoning process. Different types of knowledge representation differ in the degree to which they are analogical representations. A frame is closer to an analogical representation than is a propositional representational scheme. A propositional representational scheme is more of an analogical representation than is a procedural program which uses arrays of numbers and computations to represent objects. Direct representation is used interchangeably with analogical representation (Hayes-Roth et al 1983, 120; Barr and Feigenbaum 1981 vol 1, 200–206).

Analogy approach Machine learning that relies on learning through analogies. Other approaches to machine learning include empirical induction and discovery systems. CYC is an example of the analogy approach (Tello 1988, 509, 517).

Analysis by synthesis An image-comprehension procedure. *See also* Bottom-up comprehension (Forsyth and Naylor 1985, 86–87).

Analysis expert system An expert system that breaks down a problem in order to study or analyze a given situation. Example: MYCIN. *Contrast with* Synthesis expert system (Floyd 1988, 64).

ANALYST A real-time battlefield expert system that uses both frames and rules to interpret battlefield situations, which was developed by Mitre Corporation using FRANZ LISP (Hunt 1986, 48).

Anaphoria Using replacements in language. Pronouns are examples. Anaphoria is a problem with which computer language processors must cope.

Anaphoric reference A reference to an earlier word or phrase (Smith 1989, 10).

Ancestor A class from which an object inherits its methods, slots or values. A parent is an immediate ancestor of an object.

Ancestor class A class that is related to and higher in the hierarchy than the class under consideration (Duntemann 1989, 132).